Omanic: Towards Step-wise Evaluation of Multi-hop Reasoning in Large Language Models

Xiaojie Gu[1], Sherry T. Tong[1], Aosong Feng[2], Sophia Simeng Han[3], Jinghui Lu[4,5], Yingjian Chen[1], Yusuke Iwasawa[1], Yutaka Matsuo[1], Chanjun Park[6], Rex Ying[2], Irene Li[1,5†]
[1]The University of Tokyo, [2]Yale University, [3]Stanford University,
[4]Xiaomi EV, [5]Smartor LLC, [6]Soongsil University
[†]**Corresponding author:** irene.li@weblab.t.u-tokyo.ac.jp

Abstract

Evaluating the reasoning abilities of large language models (LLMs) solely from final answers can obscure failures in intermediate steps, especially in multi-hop QA benchmarks without step-level annotations. To address this gap, we introduce Omanic, an open-domain 4-hop QA benchmark designed not only to measure final-answer accuracy but also to diagnose where reasoning breaks down. Omanic contains 10,296 machine-generated training examples (OmanicSynth) and 967 expert-reviewed human-annotated evaluation examples (OmanicBench), with each evaluation question decomposed into single-hop sub-questions, intermediate answers, and structured graph topologies. Experiments with proprietary and open-source LLMs show that Omanic is challenging, while step-wise analysis reveals a later-hop bottleneck, factual knowledge floor, and error propagation along reasoning chains. Fine-tuning on OmanicSynth transfers to six reasoning and mathematics benchmarks, yielding a 7.41-point average gain and validating its effectiveness as supervision for reasoning-capability transfer. We release the data at https://huggingface.co/datasets/li-lab/Omanic.[1]

1 Introduction

As LLMs mature (Google, 2025; OpenAI, 2025; Qwen, 2026), the research frontier has shifted from single-task proficiency to complex reasoning, with Chain-of-Thought (CoT) prompting (Wei et al., 2022) becoming central for eliciting intermediate steps. Yet growing evidence shows that LLMs often rely on ***reasoning shortcuts*** (Shojaee et al., 2025; Gao et al., 2026; Hammoud et al., 2025), reaching correct answers through heuristic pattern matching rather than rigorous deduction, while high end-to-end accuracy can hide failures in intermediate steps (Jacovi et al., 2024). This concern is acute in multi-hop reasoning: benchmarks such as MuSiQue (Trivedi et al., 2022) advance cross-document evidence synthesis, but typically evaluate only final answers and lack step-level annotations for diagnosing *where* and *why* models fail. Without such ground truth, it remains difficult to distinguish genuine compositional reasoning from shortcut exploitation (Gu et al., 2026).

To bridge this gap, we introduce **OmanicBench** (**O**pen-domain **M**ulti-hop questions with **AN**notated reason**I**ng **C**hain), a 4-hop open-domain QA benchmark centered on step-wise diagnosis rather than final-answer accuracy alone. OmanicBench consists of 967 multi-hop questions, each manually annotated and expert-reviewed (over 300 hours of annotation effort), providing high-quality ground truth for analyzing reasoning behavior. Crucially, every question is decomposed into four cross-domain single-hop sub-questions with intermediate answers, drawing on rich factual knowledge and connected through mathematical reasoning. These annotations make it possible to evaluate each hop, inspect how answers propagate, and identify where a model's reasoning chain breaks. Besides, we also release OmanicSynth, a machine-generated training set containing 10,296 instances for supervised training and transfer experiments.

Leveraging the step-wise annotations in Omanic, we analyze state-of-the-art LLMs and reveal two phenomena obscured by final-answer evaluation. CoT gains exhibit a *knowledge floor*, diminishing as required atomic facts become missing, while errors propagate along reasoning chains, with later hops consistently showing higher error rates. We further conduct an entropy-reduction analysis to probe these behaviors. Together, these findings show that OmanicBench functions as a diagnostic testbed for separating factual retrieval, compositional reasoning, and propagation failures.

In summary, we make three main contributions. First, we propose Omanic, an open-domain 4-hop QA resource with 10,296 training and 967 expert-

[1]Code at https://github.com/XiaojieGu/Omanic.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 36623–36647
October 24-29, 2026 ©2026 Association for Computational Linguistics

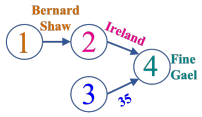
Reasoning Graph	Multi-hop Question	Single-hop Composition
	In the country of citizenship of the author of Candida, which political party was founded the same number of years before 1968 as the number of distinct 3-member committees that can be formed from a group of 7 candidates? Fine Gael *[A: Fine Gael B: Labour Party C: Fianna Fáil D: Sinn Féin]*	1. Who is the author of Candida? Bernard Shaw 2. What is the country of citizenship of Bernard Shaw? Ireland 3. How many distinct 3-member committees can be formed from a group of 7 candidates? 35 4. In Ireland, which political party was founded 35 years before 1968? Fine Gael

Table 1: An example illustrating the multi-hop reasoning graph and its step-by-step question decomposition.

reviewed testing instances, where state-of-the-art LLMs only achieve 73.11% accuracy. Second, fine-tuning on OmanicSynth yields a 7.41-point average gain across six external benchmarks, showing strong transferability and data quality. Third, using single-hop decomposition and intermediate answers, we empirically analyze the *knowledge floor* effect and *error propagation*, showing how OmanicBench localizes failures that final-answer evaluation would obscure.

2 Omanic Construction Pipeline

We describe the main dataset construction steps in this section. Figure 4 in Appendix B provides an overview of the full pipeline.

Triplets Retrieval To construct the Omanic dataset, we begin with the answers to the original 2-hop questions in MuSiQue (Trivedi et al., 2022), which are themselves composed of two single-hop questions. These answers serve as anchor subjects for retrieving corresponding $(subject, relation, object)$ triplets from Wikidata5M (Wang et al., 2021), a large-scale knowledge graph derived from Wikipedia. These retrieved triplets function as the foundational building blocks for our 4-hop expansion.

Constrained Synthesis To construct 4-hop queries, we merge original MuSiQue components with new single-hop questions synthesized from retrieved triplets, utilizing Claude-Sonnet-4.5. This process is governed by domain constraints and reasoning-graph topology. Each synthesized single-hop question is assigned to one of eight predefined domains (e.g., *History and Literature*, *Art and Architecture*), and the source Wikidata triplet is contextually refined to improve fluency and coherence. Each 4-hop instance is also required to contain at least one mathematically grounded hop. Numerical, temporal, or countable attributes are rewritten into sub-questions that require explicit quantitative reasoning, such as comparison, aggregation, counting, arithmetic composition, or temporal calculation. This mathematical hop is embedded into the chain rather than appended independently, so its inputs depend on earlier hops and its output can support later ones. For each query, we randomly choose one of three reasoning graph topologies (Trivedi et al., 2022) (Figure 7), which determines both question synthesis and final assembly. For example, under the Bridge pattern (Table 1), the second hop depends on the first, and the fourth depends on both the second and third, preventing shortcut solutions that bypass intermediate reasoning. Finally, each single-hop question is paired with three distractors, and the answer and options of the last hop are used as the ground truth and candidate set for the full 4-hop query.

Automated Filtering To maintain a great difficulty ceiling, we filter the synthesized dataset using an ensemble of four models[2]. Any question answered correctly by two or more models is deemed too simple and discarded. After pruning, 3,415 instances were removed, yielding a final training set of 10,296 examples.

Expert Review To ensure the high quality of the OmanicBench, 1,172 candidate instances undergo rigorous human audit conducted by 10 trained undergraduates and postgraduates over approximately 300 person-hours via the `Label Studio` platform[3]. For each instance, annotators first verify the factual correctness of every sub-question and its answer by consulting the Wikipedia articles linked to the underlying triplets; for questions involving mathematical reasoning, annotators are additionally required to provide detailed step-by-step computations. They then examine the logical coherence of the full 4-hop reasoning chain, checking whether it conforms to the designated graph topology, and then assign quality scores across multiple dimensions (factual accuracy, distractor plausibility, flu-

[2]Including Llama-3.1-8B-Instruct, Qwen3-8B, Mistral-7B-Instruct-v0.3, and Gemma-3-4b-it.

[3]`https://labelstud.io/`

	MCQ	Exact Match	F1-Score
Proprietary LLMs			
GPT-5.4	49.22	22.85	32.22
GPT-5.4$_{\text{CoT}}$	70.84	27.09	43.58
Claude-Sonnet-4.6	55.43	20.73	37.15
Claude-Sonnet-4.6$_{\text{CoT}}$	**73.11**	14.81†	32.27†
Gemini-3.1-flash-lite	44.88	23.47	32.31
Gemini-3.1-flash-lite$_{\text{CoT}}$	72.60	23.99	35.72
Qwen3-Max	49.02	17.79	26.31
Qwen3-Max$_{\text{CoT}}$	72.08	**35.99**	**45.51**
Open-source LLMs			
Qwen3-8B	25.65	9.26	13.77
Qwen3-8B$_{\text{SFT}}$	53.62	10.97	16.60
Qwen3-8B$_{\text{SFT+GRPO}}$	53.77	11.79	17.98
LLaMA-3.3-70B	40.04	11.77	20.47
LLaMA-3.3-70B$_{\text{SFT}}$	**57.55**	**19.42**	**29.04**

Table 2: Performance comparison of LLMs on OmanicBench. **Bold** marks the best result per metric. †: discuss in Appendix C.2.

	Step 1	Step 2	Step 3	Step 4
Direct				
GPT-5.4	85.63	86.97	84.80	66.08
Claude-Sonnet-4.6	90.59	88.31	87.49	68.46
Gemini-3.1-Flash-Lite	89.35	87.38	86.04	66.18
CoT				
GPT-5.4	93.80	95.66	92.66	79.63
Claude-Sonnet-4.6	93.59	95.86	93.38	80.14
Gemini-3.1-Flash-Lite	94.42	96.28	91.73	78.39

Table 3: Single-hop step-level MCQ accuracy of selected models on OmanicBench. Red values indicate the lowest result for each model.

ency, and reasoning integrity) following a standardized rubric (detailed in Appendix B). They also verify the correctness of the required numerical operations in mathematically grounded hops, ensuring that intermediate calculations and final derived values are arithmetically consistent. Where deficiencies are identified, annotators directly correct the instance or supplement supporting references and derivations. Instances failing to meet predefined quality thresholds were excluded, yielding a final set of 967 high-quality instances as the eval set. An example instance is shown in Table 1, and a comparison with previous benchmarks is provided in Table 6.

3 Experiments and Analysis

3.1 Models and Setup

To assess the quality and utility of Omanic, we evaluate a set of proprietary and open-source LLMs under multiple settings. For proprietary models (e.g., GPT-5 (Singh et al., 2025)), we evaluate both direct answering and CoT prompting. For open-source models (e.g., Qwen3-8B (Qwen, 2025)), we additionally fine-tune selected models on the Omanic training set via supervised fine-tuning (SFT) and GRPO-based (Shao et al., 2024) reinforcement learning. To examine whether training on OmanicSynth transfers to broader reasoning capabilities, we further evaluate the fine-tuned models across multiple reasoning benchmarks (e.g., MATH (Hendrycks et al., 2021)). For evaluation on OmanicBench, we report three complementary metrics across two evaluation paradigms. For the multiple-choice question setting in Table 2, we use Multiple-Choice Question(MCQ) accuracy (Xinjie et al., 2025), which measures selection accuracy over four candidate options. For the open-ended generation setting, we report Exact Match (EM), which requires strict string equivalence with the gold answer, and F1-Score, which captures partial credit at the token level (Rajpurkar et al., 2016). We further report accuracy on each decomposed single-hop step in Table 3, where each sub-question incorporates answers from preceding steps, enabling step-wise diagnosis beyond the final answer.

3.2 Main Results

Table 2 presents the overall performance of all evaluated models on OmanicBench. Proprietary LLMs consistently outperform open-source counterparts, while CoT prompting and OmanicSynth training both improve performance. For example, Qwen3-8B improves substantially after training on OmanicSynth (MCQ: 25.65 $\rightarrow$ 53.77), confirming that OmanicBench is challenging yet amenable to targeted training. While Table 2 reports end-to-end performance, Table 3 highlights the diagnostic value of OmanicBench by exposing substantial variation across reasoning hops. Across all selected models and prompting settings, Step 4 is consistently the most difficult, with accuracy dropping far below earlier steps even when CoT is used. This pattern suggests that later-stage reasoning is not merely harder because of weaker models, but reflects an inherent bottleneck in composing multiple intermediate facts. Therefore, OmanicBench enables evaluation beyond final-answer correctness by localizing failures along the reasoning chain. Notably, we also compare output length in Table 11:

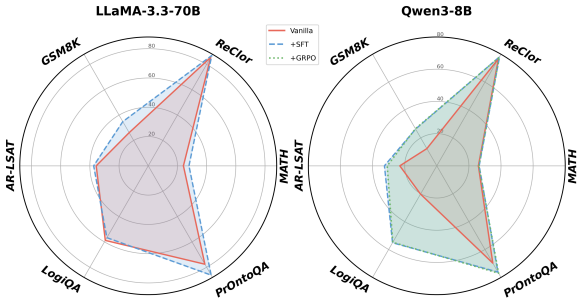


Figure 1: Comparison of accuracy across benchmarks between the vanilla model and its counterparts fine-tuned on OmanicSynth.

the gains from CoT for some models come at substantially higher inference costs. While Qwen3-Max$_{\text{CoT}}$ achieves the best open-ended performance, its output token length far exceeds that of all other models. In contrast, GPT-5.4$_{\text{CoT}}$ is considerably more efficient in practice. Full results including step-level accuracy in Appendix C.

To further assess the transferability of skills acquired from OmanicSynth, we evaluate the fine-tuned models on established reasoning (Yu et al., 2020; Liu et al., 2021; Saparov and He, 2023) and mathematics (Cobbe et al., 2021) benchmarks. As shown in Figure 1, fine-tuned models consistently outperform their vanilla counterparts, demonstrating that OmanicSynth comprises high-quality, non-trivial instances that cultivate genuine complex logical reasoning and mathematical reasoning capabilities rather than superficial pattern matching or factual knowledge retrieval alone. All implementation details are provided in the Appendix C.1.

3.3 Key Observations

Beyond aggregate performance, the step-level annotations in OmanicBench allow us to quantitatively examine three research questions about multi-hop reasoning behavior.

RQ1: To what extent does CoT rely on a sufficient knowledge foundation? (Wang et al., 2023) To investigate this question, we group multi-hop questions by the number of constituent single-hop questions answered incorrectly, and then measure multi-hop accuracy within each group. As shown in Figure 2 (left), even in the zero-error group, multi-hop accuracy under Direct prompting reaches only about 60%, well below ceiling, indicating that multi-hop reasoning and single-hop knowledge retrieval are not equivalent. Meanwhile, CoT gain decreases monotonically as the number of erroneous single-hop steps increases, dropping to near zero

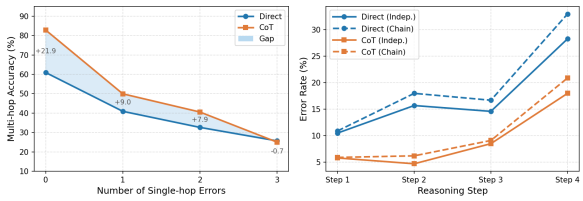


Figure 2: Results averaged across LLMs under Direct and CoT prompting. Left: Multi-hop accuracy by number of single-hop errors. Right: Step-wise error rates under independent and chain evaluation.

(-0.7) when three steps are incorrect, while the largest gain (+21.9) is observed in the zero-error group. These results quantify a clear knowledge floor on OmanicBench: effective CoT requires sufficient factual grounding, sharpening compositional inference but not substituting for missing facts.

RQ2: To what extent do errors amplify toward the end of a multi-hop reasoning chain? (Press et al., 2023) To quantify this effect, we compare two evaluation protocols: independent evaluation, where each step receives the gold answer to prior single-hop questions; and chain evaluation, where answers from previous steps propagate to subsequent steps. As shown in Figure 2 (right), even under independent evaluation, Step 4 already exhibits a substantially higher error rate than earlier steps, revealing an inherent difficulty gradient in OmanicBench beyond pure propagation effects. Under chain evaluation, errors compound: Step 4 reaches a 33.0% error rate under Direct prompting, 4.7 points higher than under independent evaluation. While CoT reduces absolute error rates at every step, the amplification pattern remains intact, suggesting that sequential multi-hop inference is intrinsically more fragile in later hops. Taken together, these analyses show that OmanicBench is not only a benchmark for end-to-end accuracy, but also a diagnostic testbed for quantifying where reasoning breaks down and how strongly errors compound across hops.

RQ3: Does fine-tuning on OmanicSynth mainly inject factual knowledge, or does it teach models to reason across hops? To understand why fine-tuning on OmanicSynth improves multi-hop performance, we compare the answer entropy of vanilla Qwen3-8B and its fine-tuned variant Qwen3-8B$_{\text{SFT}}$ on OmanicBench. The key question is whether SFT primarily injects factual knowledge (i.e., the model knows more) or teaches the model to leverage prior-hop information to progressively narrow the answer space (i.e., the model

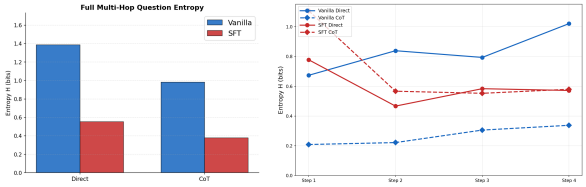


Figure 3: Answer entropy analysis for vanilla and SFT model. Left: Final answer entropy on full multi-hop questions. Right: Per-hop answer entropy under gold-context evaluation.

reasons better). Lower entropy indicates higher confidence in the answer. Higher entropy indicates greater uncertainty, with probability mass spread across multiple options.

When evaluated on full multi-hop questions, SFT substantially reduces final-answer entropy under both Direct and CoT prompting, as shown in Figure 3 (left). This indicates that the fine-tuned model produces more concentrated answer distributions regardless of whether step-by-step reasoning is explicitly elicited. Notably, the entropy of the vanilla model under CoT remains higher than that of the SFT model under Direct prompting, suggesting that fine-tuning on OmanicSynth provides greater uncertainty reduction than explicit chain-of-thought guidance alone in this setting.

To move beyond end-to-end multi-hop performance, we further analyze answer entropy at the level of decomposed single-hop questions. We analyze gold-context evaluation, where the model answers each hop with the full multi-hop question and the gold answers to all preceding hops. This setting tests whether the model can use correct prior-hop information to continue the reasoning chain and narrow the candidate answer space. As shown in Figure 3 (right), the key signal is not that gold context always lowers entropy relative to the single-hop setting; in fact, adding prior-hop answers can slightly increase absolute entropy, suggesting that additional context may introduce distractors as well as useful evidence. Instead, the diagnostic signal lies in the direction of entropy change as gold information accumulates along the chain. Under Direct prompting, the vanilla model's entropy rises, indicating that it becomes increasingly uncertain as more context is provided. In contrast, the SFT model exhibits the opposite trajectory: entropy decreases even without explicit step-by-step prompting. This divergent trend, with rising entropy for vanilla but falling entropy for SFT under the same gold-context condition, shows that SFT teaches the model to use prior-hop information to progressively narrow the answer space rather than merely recall more facts. Interestingly, vanilla Qwen3-8B under CoT exhibits even lower entropy than the SFT model. This does not contradict the effect of SFT; rather, it suggests that the relevant knowledge is largely present in the base model and can be elicited by an explicit reasoning scaffold. SFT therefore appears to internalize part of this scaffold, improving Direct prompting without requiring step-by-step instructions, while explicit CoT remains a strong external mechanism for organizing the model's latent knowledge. Taken together, these entropy patterns suggest that fine-tuning on OmanicSynth primarily teaches the model to organize and propagate information across hops, rather than simply injecting additional factual knowledge.

We then analyze single-hop evaluation, where each decomposed sub-question is answered separately without the full multi-hop context or any preceding-hop answers. This setting serves as a step-level baseline for measuring the intrinsic uncertainty of individual hops. As shown in Figure 3, under Direct prompting, the entropy gap between vanilla and SFT is small at Hop 1 but grows substantially by Hop 4, indicating that the benefit of SFT becomes larger on later, more compositionally demanding steps. If SFT mainly injected missing factual knowledge, this gap would be expected to remain more uniform across hops. The growing gap instead suggests that SFT improves the model's ability to identify the relevant evidence and resolve deeper sub-questions. Moreover, vanilla Qwen3-8B under CoT attains entropy levels comparable to, and in some hops lower than, SFT under Direct prompting. Since CoT changes only the prompt rather than the model parameters, this pattern further indicates that the relevant knowledge is largely present in the base model, while SFT primarily teaches a more effective reasoning framework for extracting and organizing that knowledge.

4 Conclusion

Omanic is an open-domain 4-hop QA benchmark combining mathematical and factual reasoning with step-level annotations for diagnosing model failures. Experiments show that CoT gains depend on factual completeness and errors accumulate across hops. Together with OmanicSynth, it supports fine-grained evaluation and targeted training of multi-hop reasoning.

Limitations

Omanic has several limitations that suggest directions for future work. First, the benchmark is restricted to English only, limiting its applicability to multilingual reasoning evaluation. Second, while 4-hop questions represent a meaningful increase in complexity over existing 2-hop benchmarks, extending to longer reasoning chains (e.g., 6-hop or 8-hop) would further test the limits of compositional reasoning. Third, although Omanic spans eight knowledge domains, certain specialized domains (e.g., legal, biomedical) remain underrepresented. Fourth, the dataset scale (10,296 training and 967 test instances) is moderate; scaling up through broader knowledge graph coverage could improve both training utility and evaluation robustness. The motivation statement on Omanic can be found in Appendix A.

Acknowledgments

Dr. Irene Li is supported by JST ACT-X (Grant JPMJAX24CU) and JSPS KAKENHI (Grant 24K20832). This work used supercomputers provided by the Research Institute for Information Technology, Kyushu University, through the HPCI System Research Project (Project ID: hp260013).

References

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Fan Gao, Sherry T Tong, Jiwoong Sohn, Jiahao Huang, Junfeng Jiang, Ding Xia, Piyalitt Ittichaiwong, Kanyakorn Veerakanjana, Hyunjae Kim, Qingyu Chen, and 1 others. 2026. Med-coreasoner: Reducing language disparities in medical reasoning via language-informed co-reasoning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 24868–24888.

Google. 2025. Gemini-3-pro. https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/3-pro. Retrieved December 12, 2025.

Ken Gu, Advait Bhat, Mike A Merrill, Robert West, Xin Liu, Daniel McDuff, and Tim Althoff. 2026. Synthworlds: Controlled parallel worlds for disentangling reasoning and knowledge in language models.

Hasan Abed Al Kader Hammoud, Hani Itani, and Bernard Ghanem. 2025. Beyond the last answer: Your reasoning trace uncovers more than you think. ArXiv preprint arXiv:2504.20708.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2023. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Alon Jacovi, Yonatan Bitton, Bernd Bohnet, Jonathan Herzig, Or Honovich, Michael Tseng, Michael Collins, Roee Aharoni, and Mor Geva. 2024. A chain-of-thought is as strong as its weakest link: A benchmark for verifiers of reasoning chains. In *Proc. of ACL*.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2021. Logiqa: a challenge dataset for machine reading comprehension with logical reasoning. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3622–3628.

OpenAI. 2025. Gpt5.1. https://openai.com/ja-JP/index/introducing-gpt-5/. Retrieved December 12, 2025.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2023. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5687–5711.

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

Qwen. 2026. Pushing qwen3-max-thinking beyond its limits.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proc. EMNLP*.

Abulhair Saparov and He He. 2023. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. In *The Eleventh International Conference on Learning Representations*.

Julian Schnitzler, Xanh Ho, Jiahao Huang, Florian Boudin, Saku Sugawara, and Akiko Aizawa. 2024. Morehopqa: More than multi-hop reasoning. *arXiv preprint arXiv:2406.13397*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical

reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

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. *arXiv preprint arXiv:2506.06941*.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, and 1 others. 2025. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*.

Yixuan Tang, Hwee Tou Ng, and Anthony Tung. 2021. Do multi-hop question answering systems know how to answer the single-hop sub-questions? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3244–3249.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. Musique: Multi-hop questions via single-hop question composition. *TACL*.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of ACL*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Proc. of NeurIPS*.

Jian Wu, Linyi Yang, Zhen Wang, Manabu Okumura, and Yue Zhang. 2025. Cofca: A step-wise counterfactual multi-hop qa benchmark. In *Proc. of ICLR*.

Zhao Xinjie, Fan Gao, Xingyu Song, Yingjian Chen, Rui Yang, Yanran Fu, Yuyang Wang, Yusuke Iwasawa, Yutaka Matsuo, and Irene Li. 2025. ReAgent: Reversible multi-agent reasoning for knowledge-enhanced multi-hop QA. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 4067–4089, Suzhou, China. Association for Computational Linguistics.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. In *International Conference on Learning Representations*.

Andrew Zhu, Alyssa Hwang, Liam Dugan, and Chris Callison-Burch. 2024. Fanoutqa: A multi-hop, multi-document question answering benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 18–37.

A Statement on OmanicBench Design and Scope

OmanicBench is designed as a controlled diagnostic benchmark for studying multi-step reasoning in LLMs, rather than a new reasoning evaluation framework Its fixed 4-hop structure allows us to compare models, localize reasoning failures, and quantify error propagation under different reasoning topologies. These questions test abilities that are broadly useful in complex reasoning settings, including factual retrieval, mathematical reasoning, and compositional dependency tracking.

B Human Annotation Guidance & Dataset Statistics

We recruit 10 undergraduate and graduate students to conduct the human correction and scoring process. To ensure ethical research practices, all annotators are compensated at a rate exceeding the local minimum wage. The entire project encompasses a cumulative total of 300 person-hours. The detailed human annotation guidance is listed in the following:
For single-hop questions:

- **Factuality and Accuracy**: This criterion assesses the truthfulness and precision of the question-answer pair within real-world contexts, specific subject areas, or the provided knowledge source. Annotators follow four steps:
  1. *Core Element Extraction.* Identify key entities (names, locations, events, dates) and logical relationships in the question.
  2. *Cross-Source Verification.* Verify each entity's attributes using reliable databases, encyclopedias, or authoritative references.
  3. *Terminology and Value Audit.* Check that specialized terms are spelled correctly and that any numerical computations are error-free.
  4. *Final Scoring.* Assign a score based on the number and severity of errors (critical vs. minor).

  - **5 (Excellent)**: All key terms, concepts, dates, and numerical values are entirely accurate. The content is supported by authoritative evidence, remains free of "hallucinations" or misleading information, and utilizes precise terminology consistent with field conventions.
  - **4 (Good)**: Most facts are accurate, with only minor flaws in non-core details that do not hinder overall understanding. Terminology is generally correct, though it may be slightly simplified while remaining acceptable within the domain.
  - **3 (Satisfactory)**: Core facts are fundamentally correct, but errors exist that could lead to partial misunderstanding. Some key terms may be poorly translated or expressed, requiring the reader to infer the true intent based on common sense.
  - **2 (Poor)**: Multiple critical facts or numerical values are incorrect, significantly impacting the understanding of the problem. Evident misuse of terminology or factual conflicts severely damages the professionalism of the content.
  - **1 (Very Poor)**: Contains severe factual errors, false statements, or entirely fabricated "hallucinations." The question and answer fail to correspond, or most content contradicts known facts.

- **Distractor Quality**: This criterion evaluates whether the three incorrect options (distractors) are sufficiently deceptive and belong to a plausible logical category. Annotators follow four steps:
  1. *Category Consistency Check.* Verify that all distractors belong to the same logical category as the correct answer (e.g., if the answer is a location, distractors must also be locations).
  2. *Domain Relevance Analysis.* Assess whether distractors share geographic, temporal, or thematic proximity with the correct answer within the same domain.
  3. *Trap Logic Identification.* Examine whether distractors exploit plausible intermediate-step errors or common over-simplifications (e.g., arithmetic near-misses or temporally adjacent entities).
  4. *Final Scoring.* Assign a score: if distractors are nearly indistinguishable without full reasoning, score 5; if they span unrelated categories, score 1–2.

  - **5 (Highly Deceptive)**: All distractors belong to the same domain. Numerical options represent logical "traps" (e.g., calculation near-misses), while semantic options consist of

easily confused synonyms, similar institutions, or entities derived from intermediate reasoning steps.

- **4 (Effective)**: Distractors are within a reasonable scope (e.g., for a question about Europe, all distractors are European). The format is highly consistent with the answer, making them impossible to exclude via simple word-class or logical loopholes.
- **3 (Moderate)**: Distractors share attributes with the answer (e.g., both are years or locations), but one or two options can be quickly excluded using general knowledge or obvious context clues.
- **2 (Weak)**: Distractors belong to the same broad category but possess obvious attribute differences (e.g., asking for a university name while providing a country name as a distractor) or inconsistent formatting.
- **1 (Non-existent)**: Distractors are completely irrelevant or cross-domain (e.g., a "year" question with "apple" as an option). These are illogical "fillers" that can be instantly identified.

• **Fluency and Completeness**: This criterion evaluates whether the language is natural, the logic is clear, the grammar is correct, and all necessary constraints for a unique answer are provided. Annotators follow four steps:

1. *Naturalness First-Read.* Read the question as a native speaker, checking for awkward phrasing, inverted word order, or logical discontinuities—paying special attention to subordinate clauses and pronoun references in longer sentences.
2. *Constraint Completeness Scan.* Verify that the question stem contains every constraint necessary to derive a unique correct answer (e.g., specific years, inclusive/exclusive conditions).
3. *Grammar and Spelling Check.* Inspect punctuation, capitalization of proper nouns, and tense consistency.
4. *Final Scoring.* Assign a score: flawless grammar with a self-contained information loop scores 5; translation artifacts or missing critical constraints lower the score accordingly.

- **5 (Excellent)**: Expression is extremely natural and smooth, fully adhering to native usage habits. The structure is rigorous with appropriate logical connectives and zero grammatical errors. All original details and necessary constraints are preserved without omission.
- **4 (Good)**: Expression is basically natural with only minor linguistic flaws. The main meaning is preserved, though some non-critical descriptive details may be missing. Sentence structures follow standard norms with negligible errors.
- **3 (Satisfactory)**: Expression is slightly rigid or exhibits a "translation-ese" tone. While the core meaning is conveyed, it may omit some important constraints or include irrelevant information; grammar errors are present but the meaning remains clear.
- **2 (Poor)**: Lacks fluency with abrupt transitions and unclear logical connections. Core information is incomplete, containing significant omissions or serious errors that affect the ability to judge the correct answer.
- **1 (Very Poor)**: Language is extremely unnatural or unintelligible, containing severe grammatical and logical errors. The content organization is chaotic and fails to reflect the original intent of the question.

For 4-hop questions:

• **Logical Integrity and Technical Formatting**: This criterion evaluates the structural rigor of the multi-hop reasoning chain and the standardized use of LaTeX, technical terminology, and code formatting. Annotators follow four steps:

1. *Logic Chain Decomposition.* Break the multi-hop question into an explicit path $A \to B \to C \to D$ and identify each intermediate node.
2. *Input–Output Matching.* Verify that the output of each preceding hop is correctly used as the input condition for the subsequent hop.
3. *Symbol and Code Audit.* Check that all LaTeX formulas, mathematical notation, currency symbols, and code blocks are correctly rendered and unaltered.
4. *Consistency Determination.* Confirm that the logic chain forms a closed loop with no breaks or circular reasoning.

- **5 (Excellent)**: The reasoning chain is perfectly airtight without gaps or circularity. All

LaTeX symbols, mathematical formulas, and currency signs ($) are technically flawless and correctly rendered.

- **4 (Good)**: The logical chain is complete and sound, though the natural language transitions between hops may feel slightly rigid. Technical formatting is basically standardized with only negligible layout flaws.
- **3 (Satisfactory)**: A minor logical flaw exists (e.g., ambiguous pronoun reference like "the artist" when multiple artists are mentioned), requiring the reader to re-read to clarify the steps. Terminology or LaTeX symbols may show slight formatting deviations.
- **2 (Poor)**: Relationships between multiple hops are unclear, or critical premises are lost due to poor phrasing, preventing a successful logical "closed-loop." Formatting is chaotic, with LaTeX symbols or code blocks appearing garbled.
- **1 (Very Poor)**: The logical chain is entirely broken or results in a paradox (e.g., asking for a "year" but requiring a "monetary amount" as the answer). Terminology is highly unprofessional, and formatting has completely collapsed.

• **Contextual Fact Consistency**: This criterion verifies whether all single-hop facts remain factually accurate and conflict-free when amalgamated into a multi-hop narrative. Annotators follow four steps:

1. *Atomic Fact Backtracking.* Compare each background claim in the multi-hop question (e.g., a date, a title, a numeric value) against the original single-hop data.
2. *Spatiotemporal Conflict Detection.* Verify that the combined timeline is logically coherent (e.g., an appointment in 1877 requires the predecessor's tenure to overlap or precede that date).
3. *Modifier Verification.* Check whether qualifiers added during composition (e.g., "the large art school") inadvertently alter the original meaning.
4. *Final Scoring.* If all intermediate-node facts are correct and transitions are natural, assign the full score.

- **5 (Excellent)**: All integrated facts (dates, locations, values) are logically consistent with one another and the real-world background. The combined scenario is realistic (e.g., a museum established in 2000 is correctly described as existing in 2005).
- **4 (Good)**: Core facts are accurate, but minor descriptive biases in non-essential details (e.g., secondary institutional titles or honorifics) occur during amalgamation without affecting the final answer.
- **3 (Satisfactory)**: Core facts remain fundamentally correct, but the timeline or logical background across hops feels slightly "awkward" or strained, though no direct factual conflict is present.
- **2 (Poor)**: Significant factual flaws appear in intermediate steps (e.g., a single-hop answer is "11 years" but is treated as "12 years" during the multi-hop calculation), rendering the final result unreliable.
- **1 (Very Poor)**: A severe factual error exists in at least one intermediate step, or the timeline is logically impossible (e.g., a divorce occurring before the marriage).

• **Answer Obscurity and Leakage Prevention**: This criterion evaluates whether the question stem accidentally reveals the final answer or if the reasoning steps provide a sufficient challenge. Annotators follow four steps:

1. *Keyword Filtering.* Search the question stem for the final answer itself or any strongly characteristic cues that directly point to it.
2. *Shortcut Test.* Attempt to reach the correct answer without completing the intermediate hops—relying only on the final segment of the question or general knowledge.
3. *Distractor Elimination Check.* Assess whether the stem provides enough non-logical information to rule out all incorrect options without genuine reasoning.
4. *Final Scoring.* If every reasoning step is indispensable for reaching the answer, score 5; if the final segment alone makes the answer obvious, lower the score accordingly.

- **5 (Excellent)**: No leakage. The solver must complete every reasoning step to find the answer. The final answer or its distinct characteristics do not appear, directly or indirectly, within the question stem.
- **4 (Good)**: The reasoning chain is intact, though some background descriptions might allow a model to narrow down the answer

range via a process of elimination rather than pure deduction.

- **3 (Satisfactory)**: Partial leakage occurs. Certain phrasing is too direct (e.g., mentioning a highly unique year or rare proper noun), allowing experienced annotators or models to "guess" the answer via shortcuts or common sense.
- **2 (Poor)**: The question contains obvious hints that make the reasoning chain significantly easier to bypass.
- **1 (Very Poor)**: Direct leakage. The final answer appears within the multi-hop description, or the question is phrased in a way that renders the logical steps meaningless.

• **Semantic Completeness and Linguistic Fluency**: This criterion evaluates the retention of all necessary constraints and the naturalness of the linguistic expression. Annotators follow four steps:

1. *Grammar and Rhetoric Scan.* Check long, complex sentences for grammatical errors and ambiguous references (e.g., multiple uses of "the artist" when several artists are mentioned).
2. *Constraint Condition Checklist.* Confirm that all critical constraints from the single-hop questions (e.g., "since 1855," "prior to") are faithfully carried over into the multi-hop question.
3. *Logical Connector Check.* Verify that connectors such as "prior to," "who," and "where" accurately reflect the inter-hop relationships.
4. *Final Scoring.* A question that reads fluently and preserves all constraints scores 5; noticeable "translation-ese" or ambiguous references lower the score to 3 or below.

- **5 (Excellent)**: All necessary constraints (e.g., specific year ranges, "inclusive," rounding requirements) are perfectly preserved. The language is natural, smooth, and adheres to native-speaker habits with precise logical connectives.
- **4 (Good)**: Constraints are complete, but the phrasing is slightly wordy. Language is fluent, though the choice of logical connectors (e.g., "prior to," "who") may be repetitive.
- **3 (Satisfactory)**: The core meaning is clear, but some non-essential constraints are omitted (e.g., missing a rounding instruction). The expression is rigid and exhibits "translation-ese" or a formulaic tone.
- **2 (Poor)**: Critical constraints required for derivation are missing, or redundant information is added that interferes with understanding. Logical connectors are used incorrectly.
- **1 (Very Poor)**: Key constraints are missing, making it impossible to determine a unique answer. The language is broken and the logical relationships are erroneous, making the text unreadable.

• **Reasoning Complexity and Domain Diversity**: This criterion measures the degree of cross-domain complexity and the depth of the logical jumps. Annotators follow four steps:

1. *Domain Counting.* Identify the number of distinct knowledge domains spanned by the question (e.g., Literature, Geography, Art History, Arithmetic).
2. *Hop Counting.* Count the number of explicit logical transitions from the starting entity to the final answer.
3. *Depth and Dependency Analysis.* Determine whether each hop requires domain-specific knowledge that cannot be bypassed through common sense alone.
4. *Level Determination.* Assign a score based on the number of domains crossed (4+ domains = top tier) and the number of non-trivial hops.

- **5 (Excellent)**: The logical chain spans 4 or more distinct domains (e.g., Art History $\rightarrow$ Geography $\rightarrow$ Law $\rightarrow$ Financial Arithmetic). Each step is strictly dependent on the previous one and cannot be bypassed via common sense.
- **4 (Good)**: The logic spans 3 distinct domains and involves at least 4 explicit logical jumps.
- **3 (Satisfactory)**: The reasoning chain involves 3 or more deep logical steps within at least 2 distinct domains. It effectively distinguishes models with single-domain knowledge.
- **2 (Poor)**: The chain involves 2 domains but only 1–2 simple logical jumps, or one of the domains relies on common knowledge.
- **1 (Very Poor)**: "Pseudo-multi-hop." The logic is extremely simple, consisting merely

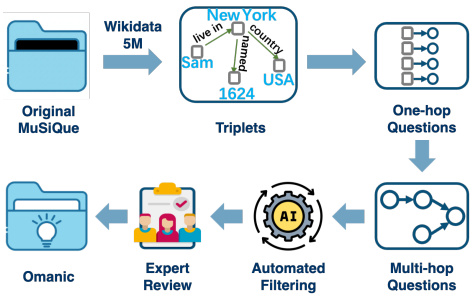


Figure 4: Overview of Omanic construction pipeline, where rectangles denote entities and circles denote single-hop questions.

Single-hop Sub-questions			
	Factuality	**Distractor**	**Fluency**
Mean_{STD}	$4.62_{0.54}$	$4.31_{0.72}$	$4.47_{0.61}$
4-hop Questions			
	Logic	**Consistency**	**Obscurity**
Mean_{STD}	$4.53_{0.58}$	$4.41_{0.65}$	$4.18_{0.79}$
	Fluency	**Complexity**	
Mean_{STD}	$4.44_{0.62}$	$4.56_{0.53}$	

Table 4: Human annotation scores on a 1–5 scale over 967 test instances. Each cell reports the mean score with the standard deviation in subscript (mean_{std}).

of the additive stacking of facts within the same domain.

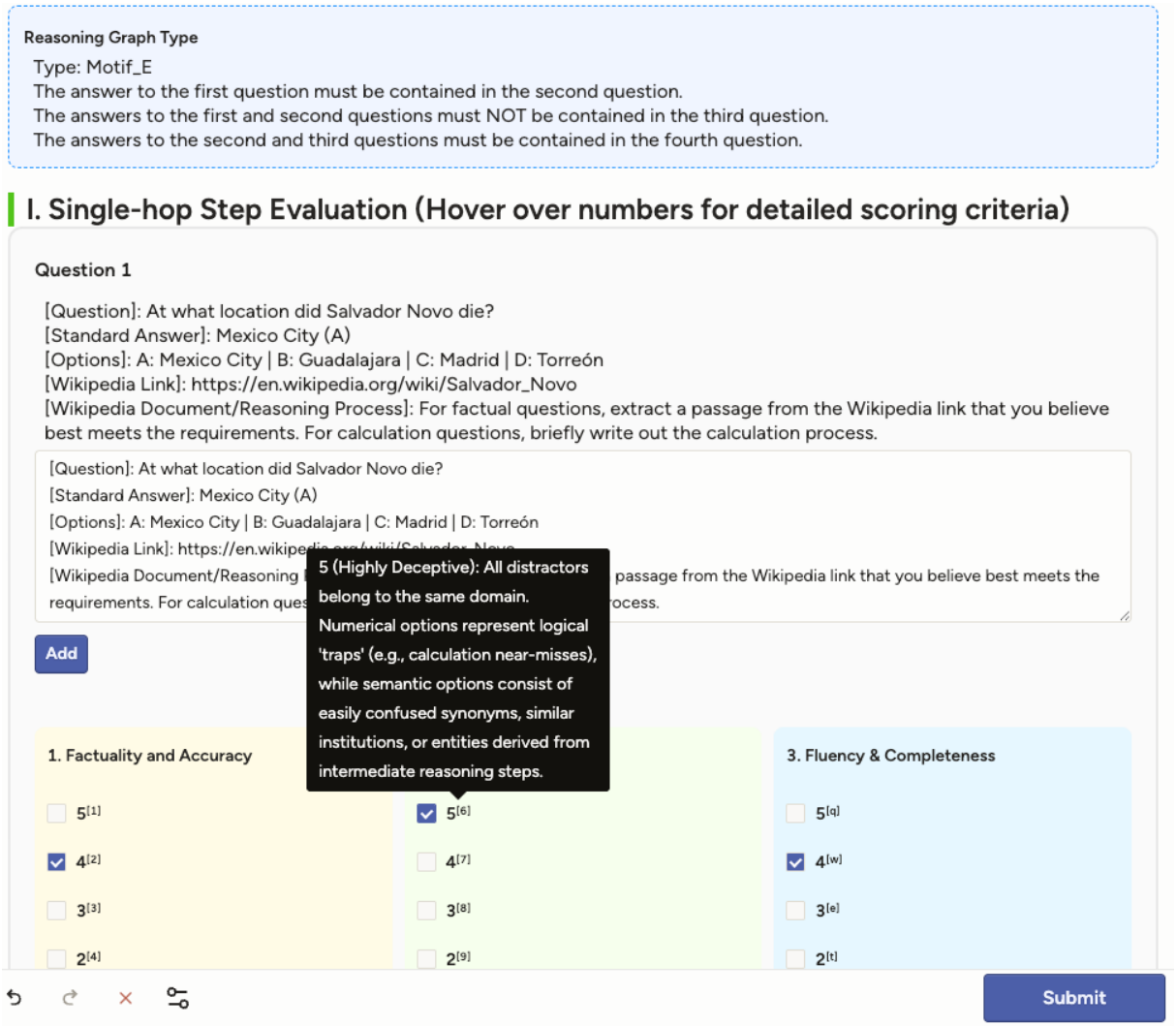


Figure 5: Annotation interface for single-hop question.

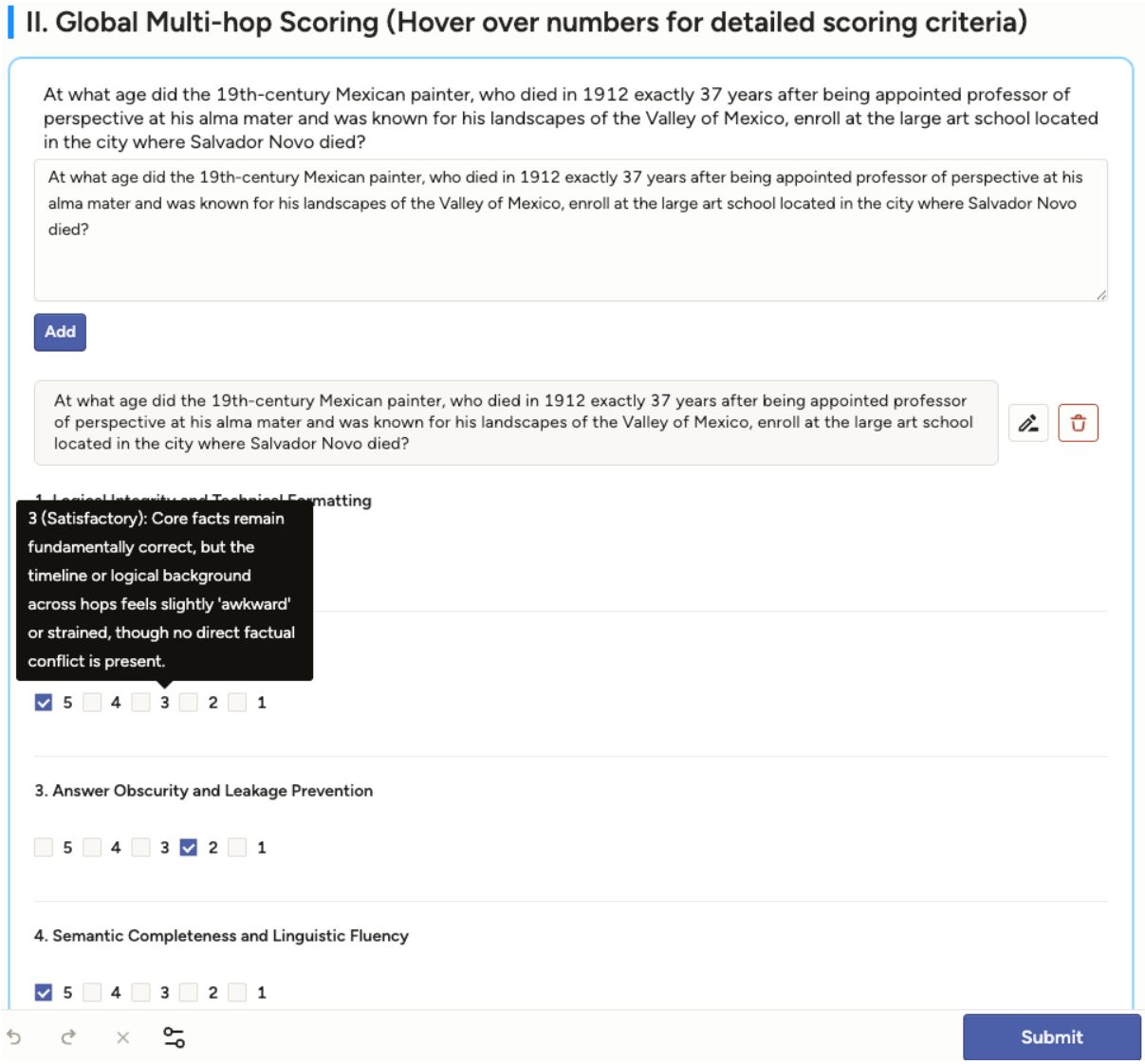


Figure 6: Annotation interface for 4-hop question.

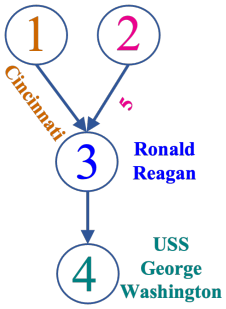
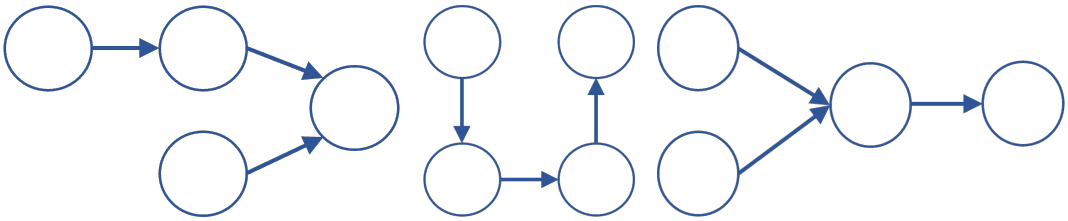
Reasoning Graph	Multi-hop Question	Single-hop Composition
	If a political regime began in 1969 and lasted for a number of years equal to 7 times the count of Apollo moon landing missions that occurred during the presidency of Eisenhower's vice president, and if the U.S. was one of 2 major powers that recognized this regime's leader early on, what was the other major power? Soviet Union *[A: United Kingdom B: France C: Soviet Union D: West Germany]*	1. Who served as Eisenhower's vice president? Nixon 2. How many successful Apollo moon landing missions occurred while Nixon was president of the U.S.? 6 3. If a political regime lasted for 6 times 7 years starting from 1969, whose face was most closely associated with Libya's government during this period? Gaddafi 4. If the U.S. was one major power that recognized Gaddafi's government at an early date, and there were 2 major powers total that did so early on, what was the other major power besides the U.S.? Soviet Union
	Great American Ball Park is the home stadium of an MLB team that has won the World Series multiple times. Multiply their total championship count by the number of players permanently banned for fixing the 1919 World Series in the Black Sox Scandal. The carrier named after the president with that ordinal number replaced which vessel at Naval Station Yokosuka, Japan, in 2015? USS George Washington *[A: USS Nimitz B: USS Carl Vinson C: USS George Washington D: USS John C. Stennis]*	1. In which U.S. city is Great American Ball Park located? Cincinnati 2. How many World Series championships have the Cincinnati Reds won in total? 5 3. Eight Chicago White Sox players were permanently banned from baseball for their role in fixing the 1919 World Series (the Black Sox Scandal). Multiply this number by the Cincinnati Reds' total World Series championships (5). Which U.S. President held the ordinal number equal to this product? Ronald Reagan 4. USS Ronald Reagan replaced which aircraft carrier as the U.S. Navy's forward-deployed vessel at Naval Station Yokosuka, Japan, in 2015? USS George Washington

Table 5: Example multi-hop reasoning graph and its step-by-step question decomposition.

Dataset	Open Domain	# Hops	Explicit step-wise chain	Topology Annotated	Math reasoning	Expert-review
HotpotQA-SubQ (Tang et al., 2021)	Yes	2	Yes	No	No	No
MuSiQue (Trivedi et al., 2022)	Yes	2-4	No	No	No	Yes
FanOutQA (Zhu et al., 2024)	Yes	**5-6**	Yes	No	No	Yes
MoreHopQA (Schnitzler et al., 2024)	No	1-5	Yes	No	Yes	Yes
CofCA (Wu et al., 2025)	Yes	2-4	No	No	No	No
SynthWorlds (Gu et al., 2026)	No	2-4	No	No	No	No
OmanicBench (ours)	**Yes**	4	**Yes**	**Yes**	**Yes**	**Yes**

Table 6: Qualitative comparison between OmanicBench and representative multi-hop or reasoning benchmarks. "# Hops" reports the supported hop range; for MuSiQue and CofCA, around 80% of the questions are 2-hop. "Explicit step-wise chain" indicates whether a benchmark provides explicit step-level annotations, such as decomposed sub-questions and intermediate answers, for diagnosing reasoning failures. "Topology Annotated" indicates whether a benchmark explicitly defines and annotates reasoning topologies (e.g., chain, tree, DAG) for its questions.



(a) Bridge (b) Chain (c) Converging

Figure 7: Three reasoning graph topologies used in Omanic for organizing 4-hop questions.

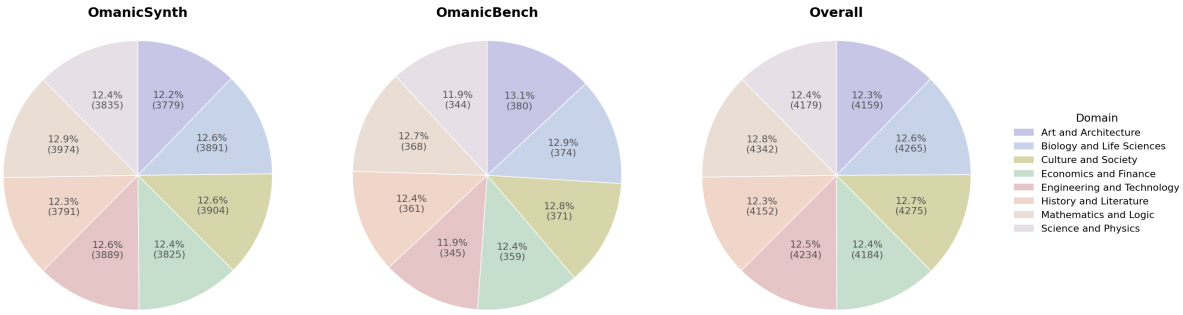


Figure 8: Domain distribution of single-hop sub-questions across OmanicSynth, OmanicBench, and the overall dataset.

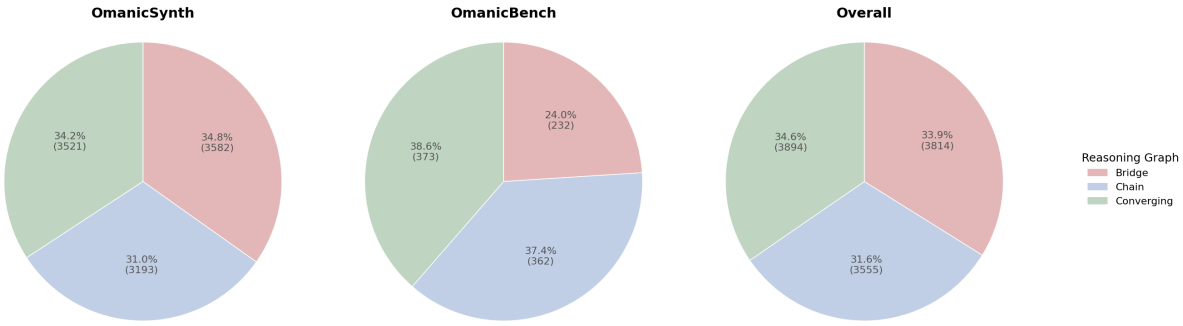


Figure 9: Reasoning graph topology distribution across OmanicSynth, OmanicBench, and the overall dataset.

C Full Results

C.1 Implementation details

For supervised fine-tuning, we construct a mixed training set in which half of the instances are formatted as multi-choice question examples and the other half are formatted as open-ended generation examples. For Qwen3-8B, we adopt a two-stage training framework: we first perform full-parameter SFT, and then conduct GRPO-based (Shao et al., 2024) reinforcement learning starting from the fine-tuned model. For LLaMA-3.3-70B, we perform LoRA-based (Hu et al., 2023) SFT on the same mixed training data. All experiments are conducted on 4 NVIDIA 96GB H100 NVL GPUs, and the detailed training hyperparameters are reported in Tables 8, 9, and 10. The evaluation templates are summarized in Table 7.

Setting	Template
Direct multi-choice question	Select the correct option from four candidates and return only the answer letter (A/B/C/D).
Direct open-ended	Answer as concisely as possible and provide only the final answer without explanation.
CoT multi-choice question	Think step by step and end the response with "The answer is X", where X is A, B, C, or D.
CoT open-ended	Think step by step and write the final answer on a separate line in the format "FINAL ANSWER: <answer>".

Table 7: Evaluation templates for direct and CoT prompting.

Parameter	Value
Cutoff length	2048
Per-device batch size	32
Gradient accumulation	2
Learning rate	1.0×10^{-5}
Epochs	3.0
LR scheduler	Cosine
Warmup ratio	0.1
Precision	BF16

Table 8: Hyperparameters for Qwen3-8B Full SFT.

Parameter	Value
Train batch size	512
Max prompt length	1024
Max response length	1024
Actor learning rate	1.0×10^{-5}
PPO mini-batch size	512
PPO micro-batch size / GPU	32
PPO epochs	1
KL loss	Enabled
KL coefficient	0.01
Number of rollouts	5
Total epochs	3

Table 9: Hyperparameters for Qwen3-8B GRPO training.

Parameter	Value
LoRA rank	8
Cutoff length	2048
Per-device batch size	32
Gradient accumulation	2
Learning rate	1.0×10^{-4}
Epochs	3.0
Precision	BF16

Table 10: Hyperparameters for LLaMA-3.3-70B LoRA SFT.

C.2 Discussion

As marked by † in Table 2, Claude-Sonnet-4.6 exhibits a unique failure mode under CoT prompting: its MCQ accuracy improves substantially, yet open-ended performance (EM/F1) degrades. Manual inspection reveals that Claude's CoT responses are substantially longer (avg. 1,812 tokens vs. GPT-5.4's 451 tokens), with the final answer frequently buried in extended prose rather than presented in an extractable format. This suggests the degradation reflects an answer extraction failure rather than a reasoning regression, underscoring the importance of evaluating reasoning capability (MCQ) and answer articulation (EM/F1) as complementary, not interchangeable, axes.

	Direct	CoT
OpenAI		
GPT-5.4	13	451
GPT-5.2	15	502
GPT-5.1	12	1,126
GPT-4o	16	1,314
Anthropic		
Claude-Sonnet-4.6	611	1,812
Claude-Sonnet-4.5	82	1,379
Claude-Opus-4.5	332	1,469
Claude-Opus-4.1	196	1,221
Claude-Sonnet-4	76	1,316
Claude-Opus-4	112	1,223
Google		
Gemini-3.1-flash-Lite	10	1,423
Gemini-3-Flash-Preview	9	1,587
Gemini-2.5-Flash	8	3,837
Gemini-2.5-Flash-Lite	8	6,453
Meta		
LLaMA-3.3-70B	38	1,694
LLaMA-3-8B	22	1,469
LLaMA-3-70B	11	735
Alibaba		
Qwen3-Max	14	4,841
Qwen3-32B	33	920
Qwen3-8B	24	357
Qwen2.5-72B	12	1,260
Qwen2.5-7B	15	1,313
DeepSeek		
DeepSeek-V3.2	11	2,548
DeepSeek-R1-Distill-LLaMA-70B	111	652
DeepSeek-R1-Distill-Qwen-32B	68	429

Table 11: Average output length under Direct and CoT prompting.

	MCQ	Exact Match	F1-Score
OpenAI			
GPT-5.2	38.51	18.20	29.09
GPT-5.2$_{\text{CoT}}$	65.98	34.13	46.93
GPT-5.1	39.40	18.55	26.62
GPT-5.1$_{\text{CoT}}$	71.68	39.96	51.18
GPT-4o	39.50	16.65	28.93
GPT-4o$_{\text{CoT}}$	66.91	37.85	47.04
Anthropic			
Claude-Sonnet-4.5	51.24	24.56	34.04
Claude-Sonnet-4.5$_{\text{CoT}}$	75.88	44.40	54.89
Claude-Opus-4.5	61.97	10.17	16.00
Claude-Opus-4.5$_{\text{CoT}}$	76.68	**44.50**	**55.44**
Claude-Opus-4.1	54.40	19.31	28.50
Claude-Opus-4.1$_{\text{CoT}}$	**76.95**	39.75	49.81
Claude-Sonnet-4	51.40	21.92	31.80
Claude-Sonnet-4$_{\text{CoT}}$	74.15	39.09	49.40
Claude-Opus-4	52.02	22.45	32.16
Claude-Opus-4$_{\text{CoT}}$	73.10	40.44	50.18
Google			
Gemini-3-Flash-Preview	67.22	21.30	31.34
Gemini-3-Flash-Preview$_{\text{CoT}}$	75.90	40.54	50.19
Gemini-2.5-Flash	62.77	18.20	25.10
Gemini-2.5-Flash$_{\text{CoT}}$	54.60	27.71	36.55
Gemini-2.5-Flash-Lite	32.68	9.31	14.04
Gemini-2.5-Flash-Lite$_{\text{CoT}}$	32.26	22.54	29.24

Table 12: Expanded proprietary LLMs results on OmanicBench.

	MCQ	Exact Match	F1-Score
Meta			
LLaMA-3.3-70B	40.04	11.77	20.47
LLaMA-3.3-70B$_{\text{CoT}}$	59.57	31.33	39.43
LLaMA-3-8B	25.23	4.96	8.78
LLaMA-3-8B$_{\text{CoT}}$	42.50	16.55	22.78
LLaMA-3-70B	38.47	12.10	18.84
LLaMA-3-70B$_{\text{CoT}}$	57.39	29.09	37.49
Alibaba			
Qwen3-32B	52.22	12.10	18.35
Qwen3-32B$_{\text{CoT}}$	54.86	30.37	38.98
Qwen3-8B	25.65	9.26	13.77
Qwen3-8B$_{\text{CoT}}$	56.44	21.76	29.97
Qwen2.5-72B	42.19	13.24	19.43
Qwen2.5-72B$_{\text{CoT}}$	60.81	32.06	41.31
Qwen2.5-7B	30.51	7.76	14.31
Qwen2.5-7B$_{\text{CoT}}$	46.85	19.44	25.60
DeepSeek			
DeepSeek-V3.2	43.33	12.82	19.36
DeepSeek-V3.2$_{\text{CoT}}$	70.94	35.92	46.19
DeepSeek-R1-Distill-LLaMA-70B	**75.92**	1.72	13.71
DeepSeek-R1-Distill-LLaMA-70B$_{\text{CoT}}$	72.67	**41.27**	**51.08**
DeepSeek-R1-Distill-Qwen-32B	69.52	11.92	22.99
DeepSeek-R1-Distill-Qwen-32B$_{\text{CoT}}$	69.29	35.51	46.00

Table 13: Expanded Open-source LLMs results on OmanicBench.

Model	Step 1	Step 2	Step 3	Step 4
GPT-4o	89.76	66.39	72.08	49.02
GPT-5.1	87.49	70.63	72.49	51.71
GPT-5.2	82.01	71.04	73.73	53.67
Claude-Sonnet-4	89.97	89.04	86.35	67.32
Claude-Opus-4	94.62	91.93	90.49	72.80
Claude-Opus-4.1	95.35	92.55	90.69	72.08
Claude-Sonnet-4.5	91.52	88.21	86.45	70.53
Claude-Opus-4.5	93.28	91.73	88.73	74.56
Gemini-2.5-Flash-Lite	72.80	66.29	68.77	45.50
Gemini-2.5-Flash	91.52	94.52	90.80	75.59
Gemini-3-Flash-Preview	97.31	96.69	92.76	79.42
Qwen3-Max	87.69	85.42	85.94	67.11

Table 14: Step-level accuracy under direct prompting on OmanicBench.

Model	Step 1	Step 2	Step 3	Step 4
GPT-4o	92.55	94.11	89.35	75.49
GPT-5.1	94.83	94.73	91.93	78.70
GPT-5.2	90.69	94.11	90.18	77.77
Claude-Sonnet-4	93.38	96.79	92.45	80.77
Claude-Opus-4	96.07	95.66	92.66	79.94
Claude-Opus-4.1	96.79	95.97	92.86	80.97
Claude-Sonnet-4.5	95.86	96.48	93.17	81.80
Claude-Opus-4.5	95.76	96.59	93.17	80.35
Gemini-2.5-Flash-Lite	72.70	47.36	40.95	26.78
Gemini-2.5-Flash	81.39	82.63	73.11	56.26
Gemini-3-Flash-Preview	96.17	95.76	92.45	80.87
Qwen3-Max	91.93	93.17	91.21	78.08

Table 15: Step-level accuracy under CoT prompting on OmanicBench.

C.3 Full results on Knowledge Floor and the Error Propagation

For the step-wise error rates under chain evaluation, we exclude single-hop questions that can be answered without relying on preceding steps, since they do not reflect dependency-sensitive error propagation. For example, under the Bridge topology, the third hop is omitted because it can be answered independently of earlier hops.

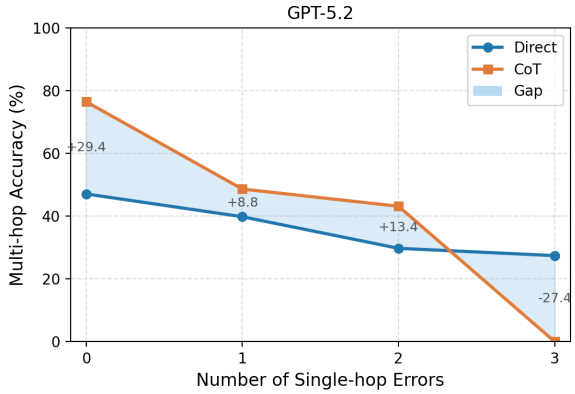


Figure 10: Multi-hop accuracy by number of single-hop errors for GPT-5.2.

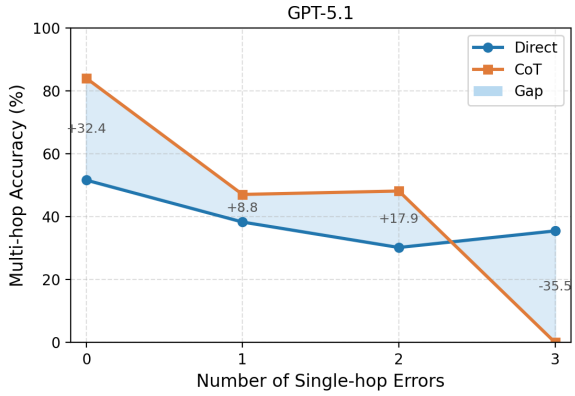


Figure 11: Multi-hop accuracy by number of single-hop errors for GPT-5.1.

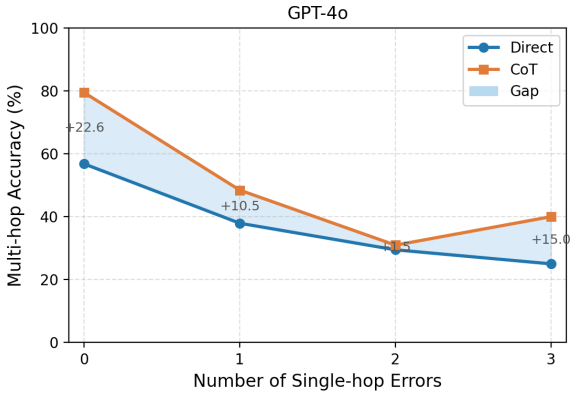


Figure 12: Multi-hop accuracy by number of single-hop errors for GPT-4o.

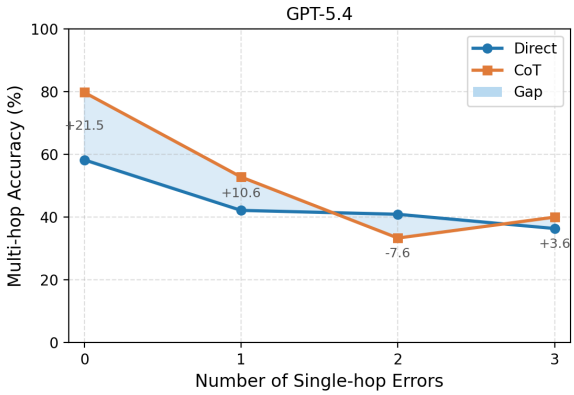


Figure 13: Multi-hop accuracy by number of single-hop errors for GPT-5.4.

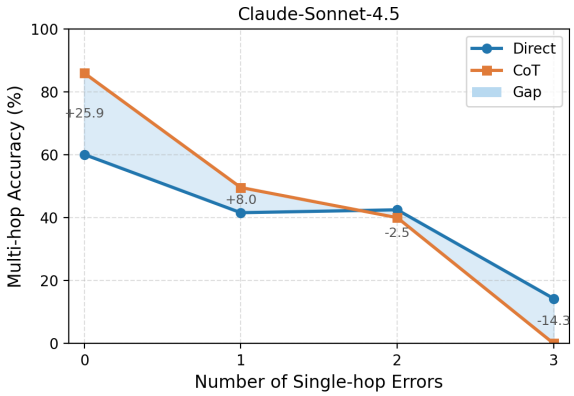


Figure 14: Multi-hop accuracy by number of single-hop errors for Claude-Sonnet-4.5.

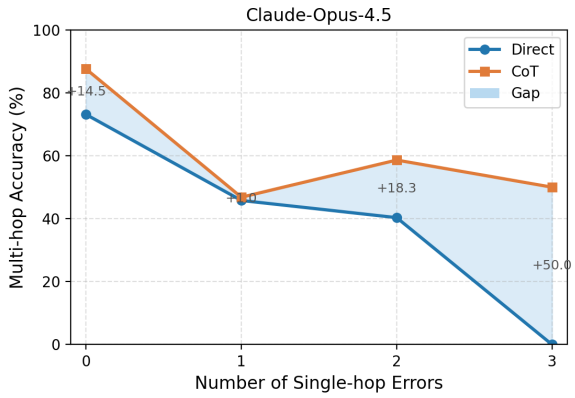


Figure 15: Multi-hop accuracy by number of single-hop errors for Claude-Opus-4.5.

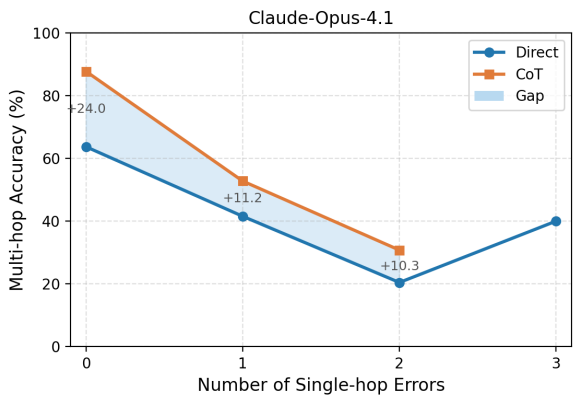


Figure 16: Multi-hop accuracy by number of single-hop errors for Claude-Opus-4.1.

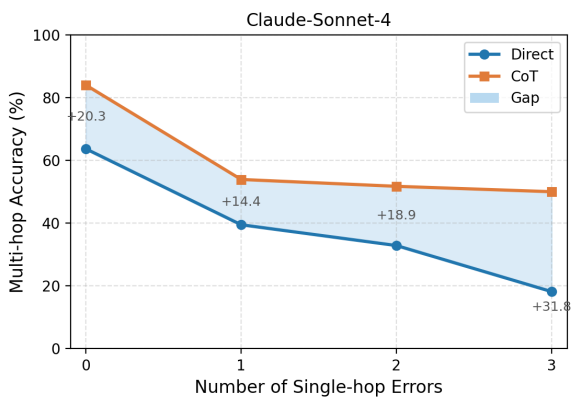


Figure 17: Multi-hop accuracy by number of single-hop errors for Claude-Sonnet-4.

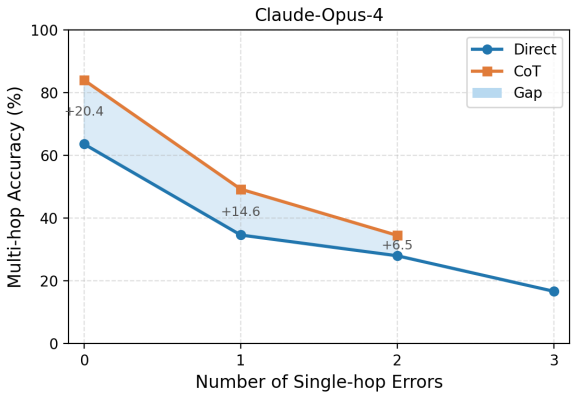


Figure 18: Multi-hop accuracy by number of single-hop errors for Claude-Opus-4.

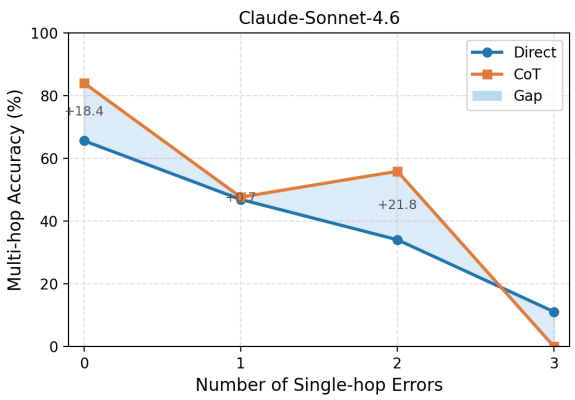


Figure 19: Multi-hop accuracy by number of single-hop errors for Claude-Sonnet-4.6.

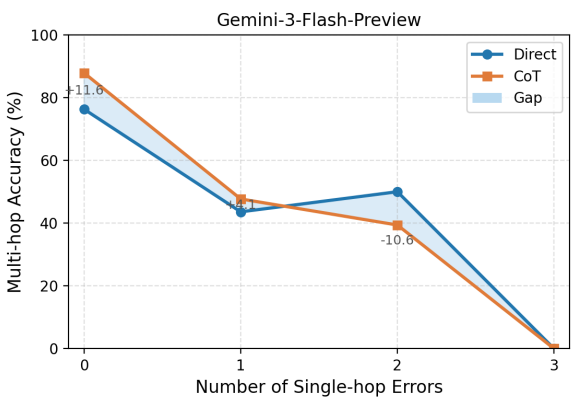


Figure 20: Multi-hop accuracy by number of single-hop errors for Gemini-3-Flash-Preview.

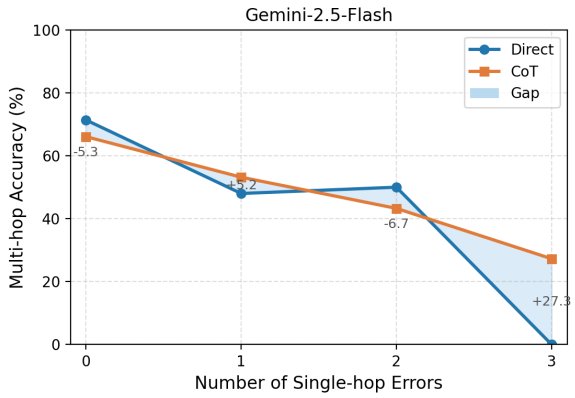


Figure 21: Multi-hop accuracy by number of single-hop errors for Gemini-2.5-Flash.

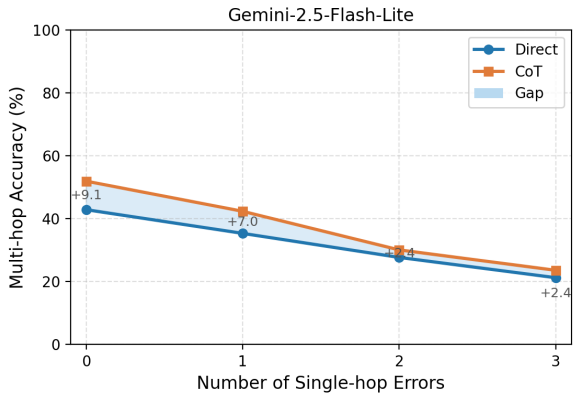


Figure 22: Multi-hop accuracy by number of single-hop errors for Gemini-2.5-Flash-Lite.

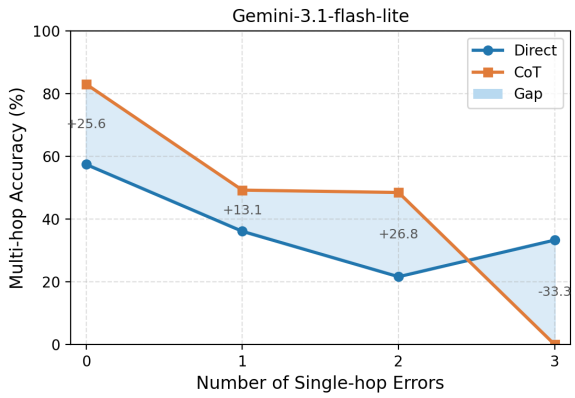


Figure 23: Multi-hop accuracy by number of single-hop errors for Gemini-3.1-flash-lite.

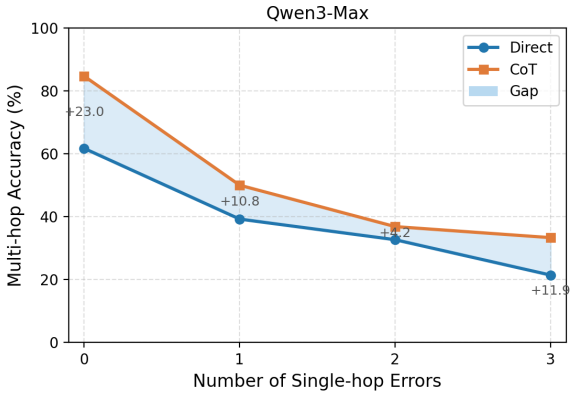


Figure 24: Multi-hop accuracy by number of single-hop errors for Qwen3-Max.

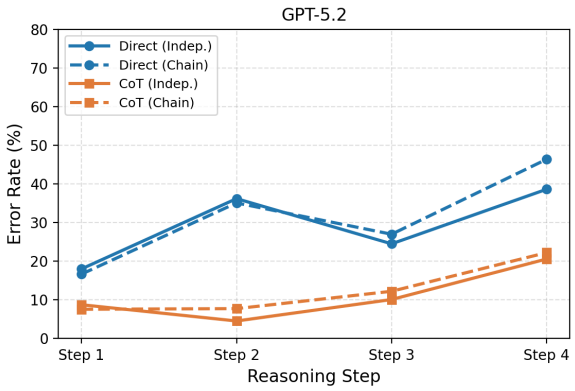


Figure 25: Step-wise error rates under independent and chain evaluation for GPT-5.2.

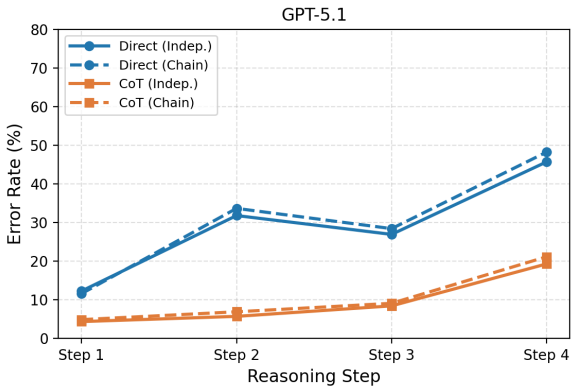


Figure 26: Step-wise error rates under independent and chain evaluation for GPT-5.1.

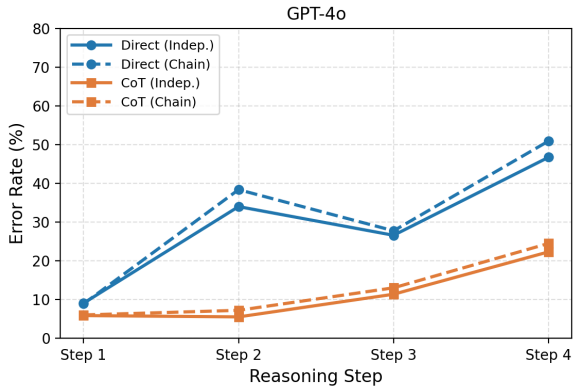


Figure 27: Step-wise error rates under independent and chain evaluation for GPT-4o.

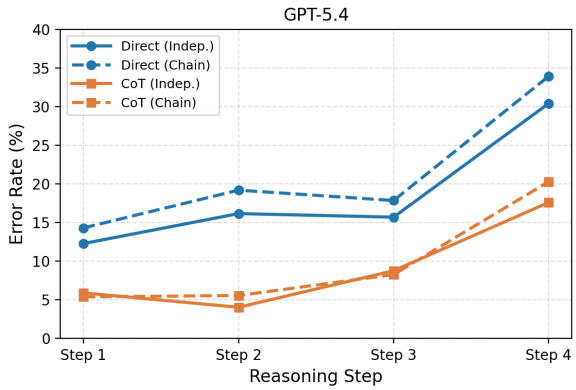


Figure 28: Step-wise error rates under independent and chain evaluation for GPT-5.4.

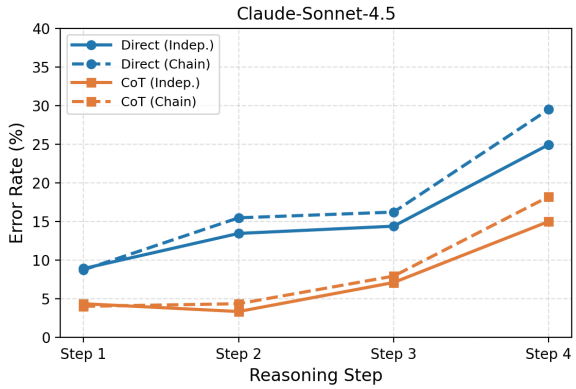


Figure 29: Step-wise error rates under independent and chain evaluation for Claude-Sonnet-4.5.

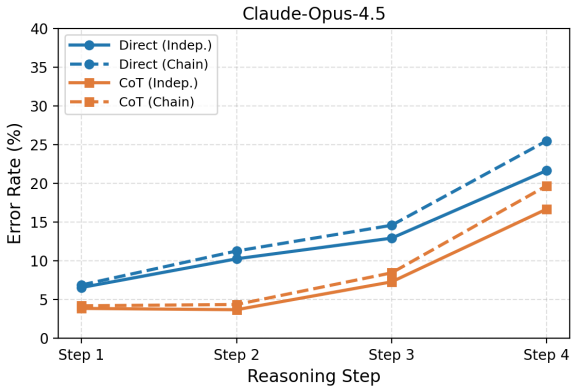


Figure 30: Step-wise error rates under independent and chain evaluation for Claude-Opus-4.5.

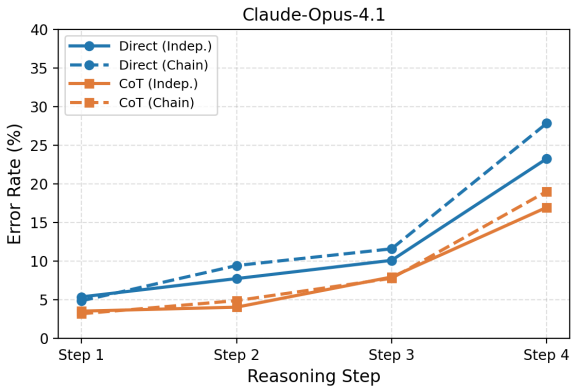


Figure 31: Step-wise error rates under independent and chain evaluation for Claude-Opus-4.1.

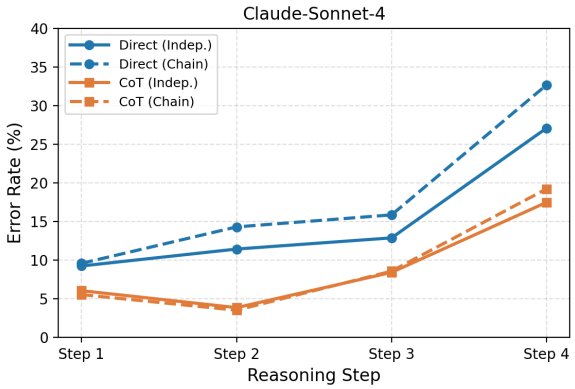


Figure 32: Step-wise error rates under independent and chain evaluation for Claude-Sonnet-4.

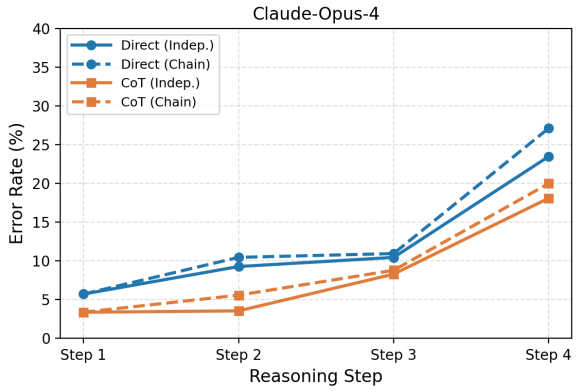


Figure 33: Step-wise error rates under independent and chain evaluation for Claude-Opus-4.

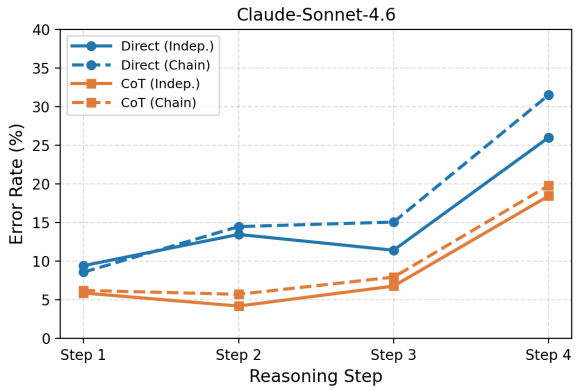


Figure 34: Step-wise error rates under independent and chain evaluation for Claude-Sonnet-4.6.

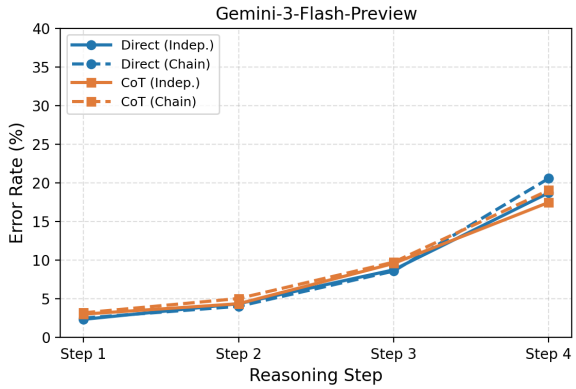


Figure 35: Step-wise error rates under independent and chain evaluation for Gemini-3-Flash-Preview.

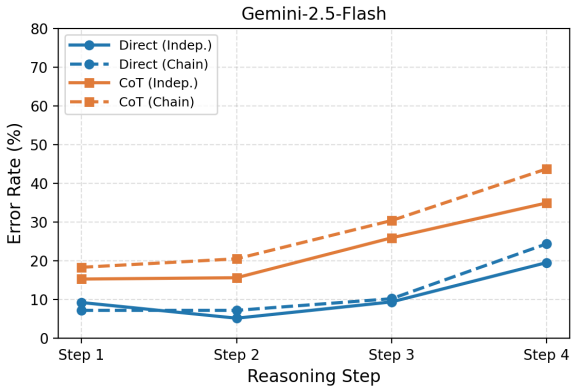


Figure 36: Step-wise error rates under independent and chain evaluation for Gemini-2.5-Flash.

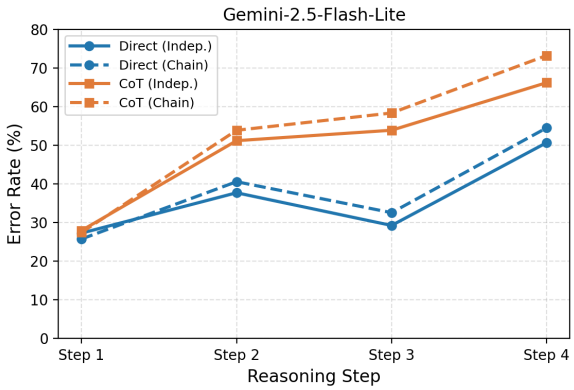


Figure 37: Step-wise error rates under independent and chain evaluation for Gemini-2.5-Flash-Lite.

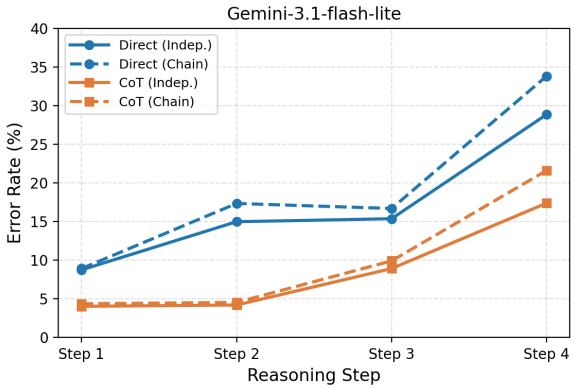


Figure 38: Step-wise error rates under independent and chain evaluation for Gemini-3.1-flash-lite.

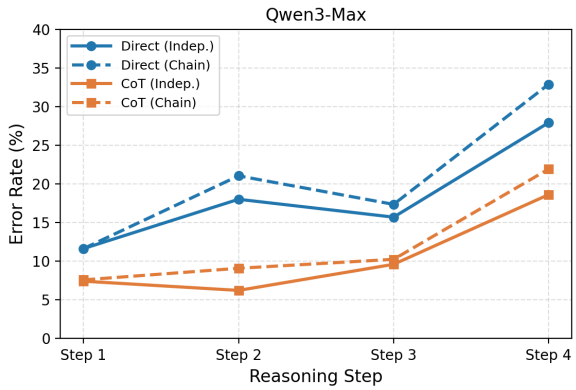


Figure 39: Step-wise error rates under independent and chain evaluation for Qwen3-Max.