
Rethinking MCQ Benchmarks: Mandatory Reasoning Evaluation Reveals Significant Performance Drops in Large Language Models

Yue Zhang
Amazon Web Services
zhangany@amazon.com

Nhan Nguyen
Amazon Web Services
ntrnguye@amazon.com

Abstract

Rigorous evaluation of Large Language Models (LLMs) is critical for their adoption in high-stakes applications, particularly in highly technical domains that require deep expertise and specialized training. The proliferation of LLMs from various providers further underscores the need for comprehensive model performance benchmarking. Like many standardized tests and certification exams, several prominent LLM benchmark datasets employ the Multiple Choice Questions (MCQs) format, such as GPQA and MMLU. Despite its convenience, prior research has demonstrated that MCQs oversimplify assessment and may not accurately reflect true model capabilities. Traditional MCQ evaluation only assesses whether the final answer is correct, potentially allowing models to succeed through lucky guessing or flawed reasoning. In the current work, we designed and performed a two-stage evaluation framework that mandates LLMs not only select the correct answer but also provide reasoning or derivation processes and justification for each option. We evaluated 21 LLMs across three model families on two datasets: GPQA and a custom dataset of 500 AWS-related questions curated from public online materials. Our results show that LLMs consistently struggle when both answer accuracy and reasoning quality are required, with performance scores dropping significantly across all models by 10% to 45% absolute points compared to conventional final answer-only evaluation. Based on these findings, we recommend that all MCQ benchmarks incorporate explanation evaluation as a standard component.

1 Introduction

Large Language Models (LLM) have made progress over the past few years. We have seen promising LLM adoptions that include, but not limited to, software engineers using LLMs to write code in an automated or copilot mode [1, 2]; scientists leveraging LLMs for research discovery to spark new ideas and polish papers [3, 4]; enterprises taking advantage of LLMs to launch LLM agents that support customers. As more and more LLM use cases in high-stake applications that require specialized domain knowledge and expertise are unlocked, rigorous evaluations of LLMs become an increasingly essential topic [5]. Evaluating large language models (LLMs) is known to be a challenging research area for various reasons. Most LLMs are auto-regressive models that generates free-form texts, making automated objective metric calculations challenging. Additionally, lack of challenging, uncontaminated benchmark datasets especially in highly specialized domains further render it difficult in designing reliable and robust evaluations. For reasons discussed above, a variety of research over the past two years dedicate to constructing evaluation benchmarks. Many of which become the standard calibration benchmark to measure capability of new models, e.g., GPQA [6], MMLU [7], HumanEval [8], SWEBench [2], MATH [9], DROP [10], etc.

A unique challenge in benchmark dataset is the format. Many datasets are multiple choice questions (MCQ) in nature, to facilitate deterministic and objective metrics calculation. Despite its convenience, there are a number of known issues to the MCQ format, e.g., evaluation results tend to show positional bias [11, 12], model could get the answer correctly by guessing or luck rather than truly understanding the problem, MCQ format also has a huge gap with the generative setting in actual use cases [13], etc. To mitigate these issues, a number of newly released benchmarks embrace generative tasks instead of MCQ formats with detailed grading rubrics that can be applied in the LLM-as-a-judge evaluators (PaperBench [14], HealthBench [15], etc.). Recently, Chandak et al. [16] applied MCQs in generative settings to compare the generative answers to the reference. They observed notable performance drop and ranking shuffles in GPQA and MMLU compared to the standard MCQ accuracy metrics.

In the current work, we propose a new, two-staged evaluation strategy for MCQ benchmarks that not only requires LLMs to get the correct final answer, but also explains the thinking processes to justify why selected answers are correct and explain why unselected answers are incorrect. This strategy was inspired partially from the fact that some exams mandate test-takers to justify their final answers in MCQs so that partial credits can be assigned and students can get more feedback for their improvements areas [13]. With the Chain-of-Thought [17] technique and recent advances in the reasoning capability, we instruct LLMs to generate explanations to its answers and use the CoT generations as the basis for reasoning evaluation.

The paper is structured as follows. In section 2, we describe the two-stage evaluation setup, including datasets, LLMs and evaluation metrics. In section 3, we discuss experimental results on two selected datasets, summarize the key findings and future work. To summarize, our key contributions are

- proposed a two-staged LLM evaluation approach for multiple-choice question benchmarks that mandates the reasoning/justification of each option to be correct in final scores.
- implemented LLM-as-a-judge to automate the reasoning/justification evaluation and evaluated the performance drops of 21 LLMs from three model families on two datasets.
- provided reflections and recommendations to use, design and adapt MCQ benchmarks

2 Methods

2.1 Experiment Setup

Datasets: We experiment on two MCQ datasets, the GPQA dataset [6] and a custom-curated AWS 500 Dataset (see Appendix A for details). **GPQA dataset** consists of 448 challenging, graduate-level multiple-choice questions in the domains of biology, physics and chemistry written and validated by experts with graduate degrees. (Our experiment used the GPQA main set with all 448 questions.) Each GPQA question has one single correct answer and the explanation. **AWS 500 dataset** is a collection of 500 exam-style questions that covers several aspects of AWS cloud technical knowledge, sourced from online public materials. While most questions only have one correct answer from four options, there are questions to select two or three from a list of six options. Like GPQA, each questions has ground truth answers and explanation for each option.

Models: Our analyses broadly cover 21 LLMs from three model families (Anthropic Claude [18], Mistral AI [19], Meta LLama 3 [20]). All models available on Amazon Bedrock and experiments are run via Bedrock API. The full list of models considered in current work is provided in Appendix B.

Prompting: We provide our prompt templates in Appendix C. Briefly, zero-shot prompts are the baselines for both datasets. For few-shot prompting, we varied the number of examples (1, 3 and 5) for AWS 500 Dataset. For GPQA, we followed the 5-shot example setting as previously reported. The few-shot examples are randomly selected from all questions, excluding the question to be answered. We instructed the model to generate reasoning via Chain-of-Thought prompts. (Our prompts also clearly state that the reasoning will be evaluated as part of the grading process). Note that all our CoT prompting involve only one single LLM call, which is slightly different from the two LLM calls configuration in the original GPQA code repo [6].

2.2 Evaluation metrics

Our proposed two-stage MCQ evaluation assess LLMs not only on the final answer correctness but also on the derivation steps or explanation of each option. We define three metrics,

Answer Score: the answer score is the standard MCQ accuracy metric, defined as the ratio of questions with correct final answer to the total number of questions. The answer scores range from 0 to 100 (%) and no reasoning evaluation is involved in this score. Note that some AWS 500 questions require selecting two or three options from the list, we count success when all selections are correct.

Explanation Score: the explanation score measures how many questions that are correctly answered have the correct and comprehensive reasoning steps, solid justification of why answer is selected and comprehensive explanation of why other answers are incorrect. The score ranges from 0 to 100(%), but note that the denominator here is the count of correctly answered questions not full questions set.

Final Score: the full score is the most strict score, which is calculated as the number of questions that LLMs get both final answer and the explanation correct divided by the total question counts. The final score ranges from 0 to 100 (%) and is equal to or lower than the answer score.

Implementation of Answer Score and Final Score are deterministic and similar to the standard MCQ accuracy metrics. For the explanation score, we leveraged the detailed explanations in the ground truth sets that are available for both GPQA and AWS 500 datasets. We implemented a reference-based, LLM-as-a-judge evaluator [21] to compare the LLM reasoning with golden explanations. The judge prompt (shown in Appendix C) has detailed rubrics and returns a binary TRUE/FALSE, which will be used to calculate the explanation scores. For each reasoning evaluation, we apply the self-consistency technique [22] and run the judge model 3 times with temperature $T = 1$ with majority vote. For GPQA reasoning evaluation, we adopted Claude Sonnet 4 as the judge model for its superior performance. For AWS 500 Dataset, we calibrated all 21 models (shown in Figure 2) and chose Claude Sonnet 3.7 as the performances are comparable with Sonnet 4.

3 Results and Discussions

3.1 Gap between answer vs full scores — Do LLMs really saturate on MCQ benchmarks?

Recent reports argue that LLMs saturate on challenging benchmarks including GPQA and thus more benchmark datasets are needed [23]. Figure 1 shows that on both GPQA and AWS 500 Dataset, recent models demonstrate strong performance and the answer scores (shown in blue bars) are rather high and improvements are somewhat saturated. However, comparing the full scores with the proposed two-stage evaluation (shown in orange bars), model performances drop notably, leading to significant performances ranking shuffle. For GPQA dataset, the averaged performance drop (Δ) clusters around 20% with a 5% standard deviation. Latest LLMs like Sonnet 3.5, Sonnet 3.7 and Sonnet 4 show lowest performance drop (all around 10% to 15%), while the largest gap is 26.3% for Llama 3.3 70B (despite the fact that its answer score is similar to that of Sonnet 3.5 v1). We observed similar trends on the AWS 500 Datasets and the drop is even more notable with larger variance across models (as large as 45%). This indicates that while answer scores seem to saturate, the full scores still have significant room for improvement. Suffice it to note that from the current observation, the more stringent, two-stage evaluation is a better metrics to investigate the emergent abilities [24, 25] than the conventional answer score on MCQ datasets.

From Fig. 1 a& b, we observed that for both datasets, Sonnet 3.5, 3.7 and Sonnet 4 show smallest gap and Llama 3.3 70B show largest gap. However, other models ranked in the middle present drastically different rankings on performance drop for the two datasets. For example, Mistral Small drops only 14% on GPQA (fifth smallest) but on AWS 500 dataset it presented a significant 28.4% drop. Haiku 3 gap is just around 25.8% on AWS yet 22% (third largest drop) on GPQA. These differences could be explained by the task nature for the two different datasets. GPQA questions are biology, physics and chemistry questions that require more reasoning and derivation, while AWS 500 Dataset is more on knowledge tests. This observation strengthens the argument that model evaluation must be performed for each specific use cases, especially for the small- to mid-sized models.

Our investigate demonstrated that prompting itself does not fully solve the performance gap between full and answer scores across all models. Results presented in Figure 1 were using zero-shot prompt setting. More details on prompting ablation studies can be found in Appendix D.2. Briefly, we tested the performance gap using zero-shot, five-shot with reasoning in examples and five-shot without reasoning in examples for GPQA. It was found that the score gaps for most models, especially larger models, show minimal sensitivity to the prompt configurations. However, with five-shot reasoning examples, Haiku 3.5 and Llama 3.3 70B, the gaps reduced from 22.1% to 15.6% and 26.3% to 19.9%,

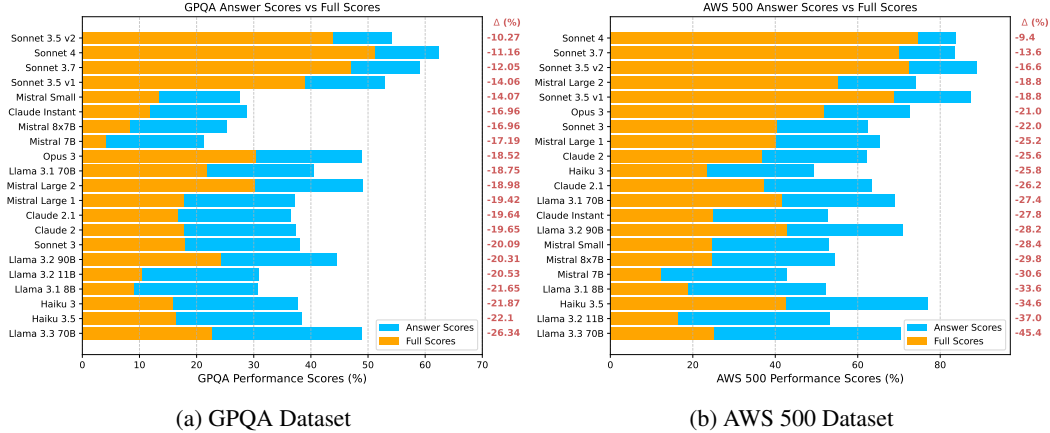


Figure 1: Model performance comparison of the conventional answer score (% , blue bars) and the proposed full score (% , orange bars) that requires two-staged, mandatory reasoning evaluation. The Δ (%) values present the gap between full scores and answer scores for each model, ranked in ascending orders for each dataset. Raw data can be found in Table 3 and Table 2 in Appendix.

respectively, yet the gaps for smaller models including Mistral Small and Mistral 8x7B increased from 14.1% to 19.6% and 17% to 23%, respectively. We attribute such observations to the fact that smaller models do not have the capacity to fully process the few-shot examples and the reasoning traces provided and may get confused and distracted, leading to decrease in the quality of responses.

3.2 How could model explanation go wrong?

In this section, we investigated the reasoning evaluation to understand how LLMs make mistakes in explanations and highlight improvement areas for future research. Common errors observed from the analyses show that for knowledge tests in AWS 500 Datasets, even if LLMs pick the correct answer, it either could not explain why other options are incorrect or the explanation shows incorrect technical understanding of the concepts. This lack of solid understanding contributes to the fact that certain LLMs have large score variance across multiple runs. For GPQA, we also observed the scenarios where LLMs used completely incorrect derivation steps or calculations that led to a result different from any of the options given. In those scenarios the LLM just picked one option closest to its incorrect answer. As one example shown in Appendix E, LLM incorrectly derived the final product of a chemical reaction but get the final answer right when counting the oxygen atom by luck. The analyses and categorization of incorrect reasoning for all cases merits further investigation.

3.3 Reflection on design of benchmark dataset

Despite issues with MCQ benchmarks and many new datasets moving to generative format, we argue that MCQ is still useful given its convenience and we have not yet truly saturated on them. Based on our experiments, we reflect and provide the following suggestions on MCQ benchmarks

Distractor-options are essential in useful MCQ benchmarks. Unlike open-ended questions, MCQs can confuse and distract test-takers with carefully curated distractor option beyond just the problem statement. With the enforced explanation grading and curated distractors, even MCQ can pose notable challenges to LLMs and test takers.

Explanation and justifications in the ground truth answers are valuable assets to systematic LLM evaluations. In developing the MCQ benchmark dataset (or adapting existing ones), we recommend to have both the ground truth answers and the reference justifications. These contents can be readily used as the reference in the LLM-as-a-judge evaluator to measure the LLM reasoning quality in an automated manner.

References

- [1] Muhammad Shihab Rashid, Christian Bock, Yuan Zhuang, Alexander Buccholz, Tim Esler, Simon Valentini, Luca Franceschi, Martin Wistuba, Prabhu Teja Sivaprasad, Woo Jung Kim, et al. Swe-polybench: A multi-language benchmark for repository level evaluation of coding agents. *arXiv preprint arXiv:2504.08703*, 2025.
- [2] Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? *arXiv preprint arXiv:2310.06770*, 2023.
- [3] Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The ai scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024.
- [4] Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024.
- [5] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.
- [6] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark, 2023.
- [7] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyang Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.
- [8] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.
- [9] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.
- [10] Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv:1903.00161*, 2019.
- [11] Ruizhe Li and Yanjun Gao. Anchored answers: Unravelling positional bias in gpt-2’s multiple-choice questions. *arXiv preprint arXiv:2405.03205*, 2024.
- [12] Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. *arXiv preprint arXiv:2308.11483*, 2023.
- [13] Pablo Frank Bolton, Liberty Rose Lehr, Rahul Simha, and Michelle Lawson. The justification effect on two-tier multiple-choice exams. In *2024 ASEE Annual Conference & Exposition*, 2024.
- [14] Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, et al. Paperbench: Evaluating ai’s ability to replicate ai research. *arXiv preprint arXiv:2504.01848*, 2025.
- [15] Rahul K Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, et al. Healthbench: Evaluating large language models towards improved human health. *arXiv preprint arXiv:2505.08775*, 2025.
- [16] Nikhil Chandak, Shashwat Goel, Ameya Prabhu, Moritz Hardt, and Jonas Geiping. Answer matching outperforms multiple choice for language model evaluation. *arXiv preprint arXiv:2507.02856*, 2025.

- [17] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- [18] Anthropic. Model card for claude. https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf, 2023. Accessed: 2024-04-30.
- [19] MistralAI. Mistral large. <https://mistral.ai/news/mistral-large/>, 2024. Accessed: 2024-04-30.
- [20] MetaAI. Llama 3: Model card. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md, 2024. Accessed: 2024-04-30.
- [21] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023.
- [22] 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.
- [23] Douwe Kiela, Tristan Thrush, Kawin Ethayarajh, and Amanpreet Singh. Plotting progress in ai. *Contextual AI Blog*, 2023. <https://contextual.ai/blog/plotting-progress>.
- [24] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.
- [25] Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo. Are emergent abilities of large language models a mirage? *Advances in neural information processing systems*, 36:55565–55581, 2023.
- [26] Sewon Min, Xinxin Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work?, 2022.

Appendix

A AWS 500 Dataset

As the largest global cloud service provider, Amazon Web Services (AWS) offers hundreds of cloud services and helps customer troubleshoot technical issues, proactively suggest optimization and cost-saving strategies, etc. The technical skills that AWS covered is broad, and they require specialized domain knowledge and training that may not necessarily be captured by the off-the-shelf LLMs. To our knowledge, there has not been a systematic and comprehensive evaluation for understanding LLMs’ performance on technical skills related to AWS.

A useful resource to test model’s AWS expertise is using the cloud certification exams provided by AWS. The exams aim to validate AWS technical skills and cloud expertise of cloud engineers. Exams are offered in four levels, foundational, associate, professional and specialty, with increasing difficulty and depth of the knowledge covered. At each level, there are different exams for a diverse range of AWS topics (e.g., DevOps, Solution Architect and Advanced Networking). These standardized test are structured in the multiple choice format (choose one option from four options) and multiple response (choose two or three from five to six options) format. To pass the exams, test takers need to complete 60-80 questions within 2-3 hours and answer around 70%-80% of questions correctly.

We curated a total of 500 exam-style questions about AWS related knowledge that span seven topics with three difficulty levels, as shown in Table 1. The questions for foundational levels are neglected in the current study with the assumption that they might not be challenging enough for LLMs.

Table 1: Decomposition of AWS 500 Questions by exam level and topics

Exam Level	Exam Topic	No. of Questions
Specialty	Advanced Networking Specialty (ANS-C01)	60
	Security Specialty (SCS-C02)	60
	Database Specialty (DBS-C01)	30
Professional	Solution Architect Professional (SAP-C02)	70
	DevOps Engineer Professional (DOP-C02)	70
Associates	Solution Architect Associates (SAA-C03)	70
	SysOps Administrator Associate (SOA-C02)	70
	Developer Associate (DVA-C02)	70

The questions were sourced from the publicly available (e.g., Google search, Github repos, etc.) sample exam prep materials of the associated exam topic. We developed a suite of pre-processing scripts to parse the contents into question statement, options list, golden answers and golden explanations for each option in the list (i.e., explanation of why correct answers are selected and how incorrect answers are wrong). To mitigate the potential data contamination and positional bias, we shuffle the option list with random orders and changed the correct answer/explanation accordingly.

B Bedrock Models

Model Family	Model Name	Bedrock ID
Anthropic	Sonnet 4	us.anthropic.claude-sonnet-4-20250514-v1:0
	Sonnet 3.7	us.anthropic.claude-3-7-sonnet-20250219-v1:0
	Sonnet 3.5 v2	us.anthropic.claude-3-5-sonnet-20241022-v2:0
	Sonnet 3.5 v1	us.anthropic.claude-3-5-sonnet-20240620-v1:0
	Haiku 3.5	us.anthropic.claude-3-5-haiku-20241022-v1:0
	Opus 3	us.anthropic.claude-3-opus-20240229-v1:0
	Sonnet 3	us.anthropic.claude-3-sonnet-20240229-v1:0
	Haiku 3	us.anthropic.claude-3-haiku-20240307-v1:0
	Claude 2.1	anthropic.claude-v2:1
	Claude 2	anthropic.claude-v2
	Claude Instant	anthropic.claude-instant-v1
Mistral	Mistral Large 2	mistral.mistral-large-2407-v1:0
	Mistral Large 1	mistral.mistral-large-2402-v1:0
	Mistral Small	mistral.mistral-small-2402-v1:0
	Mistral 8x7B	mistral.mixtral-8x7b-instruct-v0:1
	Mistral 7B	mistral.mistral-7b-instruct-v0:2
Llama	Llama 3.3 70B	us.meta.llama3-3-70b-instruct-v1:0
	Llama 3.2 90B	us.meta.llama3-2-90b-instruct-v1:0
	Llama 3.2 11B	us.meta.llama3-2-11b-instruct-v1:0
	Llama 3.1 70B	us.meta.llama3-1-70b-instruct-v1:0
	Llama 3.1 8B	us.meta.llama3-1-8b-instruct-v1:0

C Prompt template and experiment design

In the current analyses consist of three modules: system prompt, user prompt and assistant prompt. The system prompt gives overall instructions and contexts for the question and is consistent among all experiments. The user prompt includes the question and option list, along with few shot examples if applicable. The assistant prompt ends with an open xml tag to facilitate easier parsing of the structured LLM output. We further explore some basic prompting techniques including Chain of Thought (CoT) [17] and few shot In Context Learning [26]. Key experiment setup are listed below.

- **Baseline:** zero-shot prompting for LLM to generate only final multiple choice answer.
- **Chain-of-Thought (CoT):** generate reasoning sequences before giving the final answer.
- **Few-shot prompting:** add 1, 3 or 5 randomly selected examples to the prompt but only give final answers in examples.
- **Few-shot example reasoning:** add 1, 3 or 5 randomly selected examples to the prompt, each example has final answer and the explanation from official solutions.

C.1 Prompt templates to generate answers and reasoning

System Prompts for Answers without Reasoning Requirements

You are given a question that tests your knowledge for Amazon Web Services (AWS).
Read the following instructions before you attempt the test

`<instructions>`

1. The question by default asks you to only select ONE option from the list. However, if you see (Select TWO.) or (Select THREE.) in the question, you should select two or three options accordingly.
2. In your response, only return your answer and enclose it in `<answer></answer>` and do not return anything else. For example, `<answer>B</answer>`
3. If you need to choose multiple options, separate them by comma, for example `<answer>B,C</answer>`.

`</instructions>`

System Prompts for Answers with Reasoning Requirements

You are given a multiple choice question in an exam and you need to answer it. Read and follow the instructions below closely to understand the grading criteria and prepare your answer.

`<instructions>`

1. This is a multiple-choice questions and there is only ONE single correct answer, so you should only choose one option in your answer.
2. You need to first read through the questions and a list of options carefully and understand the problem.
3. You then solve the problem step by step and write down your reasoning/solution processes and enclose it in `<thinking></thinking>` tag. Be concise, do not write more than 4 or 5 sentences.
4. Now return your final answer and enclose it in `<answer></answer>`. For example, `<answer>B</answer>`
5. Important - Your grades depend not only on the final answer but also the reasoning you process provide to explain your choices.

For questions that requires step-by-step derivation process that eventually leads to a number, you need to provide detailed explanations and the step-by-step derivation processes.

For conceptual questions where options are some statements/terms, you should explain both why you chose your final answer and why you do not choose others.

`</instructions>`

C.2 GPQA reasoning evaluation prompts

GPQA Reasoning Evaluation System Prompts

You are an expert grader for graduate-level biology, physics, and chemistry exam questions. Your task is to evaluate whether test takers truly understand the material or simply guessed correctly.

`<task_overview>`

You will receive:

- A multiple choice question with the correct answer
- Golden reasoning from official solutions
- A test taker's response with their reasoning process

Your goal: Determine if the test taker demonstrates genuine understanding rather than lucky guessing or elimination tricks.

`</task_overview>`

`<grading_process>`

`<step_1>`

Identify Question Type: Read the golden reasoning to determine: Derivation-focused: Primarily step-by-step mathematical calculations/derivations Conceptual-focused: Systematic explanations of why each option is correct/incorrect based on concepts, statements, or terminology

`</step_1>`

`<step_2>`

Evaluate Test Taker's Reasoning: Compare the test taker's response to the golden reasoning using the appropriate criteria below.

`</step_2>`

`<step_3>`

Apply Grading Rubrics:

For Derivation-focused Questions:

REQUIRED: Correct mathematical steps and key derivations leading to the answer

EXPECTED: Understanding of why the correct answer is correct

NOT REQUIRED: Explanations of why other numerical options are wrong (no penalty for missing this)

For Conceptual-focused Questions:

REQUIRED: Clear understanding of why the correct answer is correct

REQUIRED: Explanations of why key incorrect statements/claims are wrong (when covered in golden reasoning)

EXPECTED: Any mathematical steps present should be correct

Universal Standards:

Minor wording differences are acceptable

Clear contradictions with golden reasoning result in grade loss

Fundamental conceptual mistakes result in grade loss

`</step_3>`

`<step_4>`

Make Final Decision: Assign FALSE only for:

Major mathematical errors in derivation steps (derivation questions)

Fundamental misunderstanding of the correct answer (both question types)

Complete failure to explain why key incorrect concepts are wrong (conceptual questions only, and only when golden reasoning covers this)

`</step_4>`

`</grading_process>`

`<output_format>`

Analysis: Briefly summarize your comparison in tags (3-4 sentences), enclose the thoughts in `<thinking>``</thinking>`.

Decision: `<reasoning_grade>` TRUE `</reasoning_grade>` or `<reasoning_grade>`

FALSE `</reasoning_grade>`. Grade TRUE if the test taker demonstrates solid understanding appropriate to the question type without major errors.

`</output_format>`

GPQA Reasoning Evaluation User Prompts

Here is the question you will be grading

<question>

{question}

</question>

Here are the options for the question above:

<options>

{options}

</options>

Here is the golden answer provided by the official solution:

<golden_answer>

{golden_answer}

</golden_answer>

Here is the golden reasoning process provided by the official solution:

<golden_reasoning>

{golden_reasoning}

</golden_reasoning>

Given the context and the golden answer above, your need to grade the following reasoning processes from the test taker:

<test_taker_response>

{test_taker_response}

</test_taker_response>

D Additional Experiment Results

D.1 AWS 500 Dataset Results

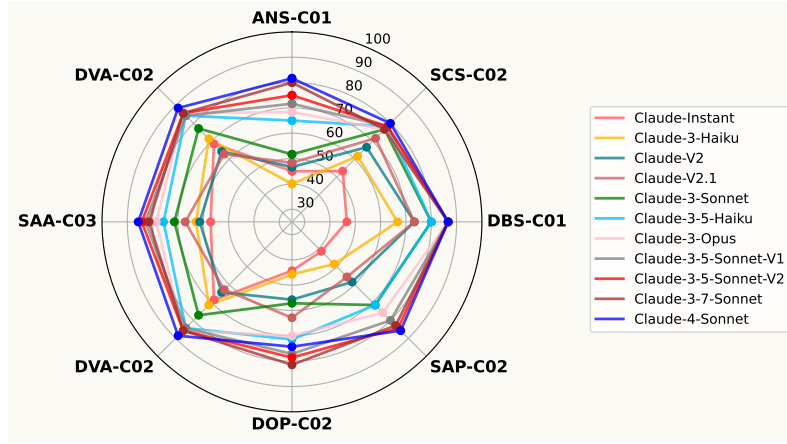
Figure 2 shows the conventional model answer scores on the AWS 500 Dataset that tests its knowledge on AWS related topics. The results presented in the polar chart were generated from zero-shot prompts without CoT reasoning. As can be seen from the figure, the performances of the latest Anthropic models including Sonnet 4, Sonnet 3.7 and Sonnet 3.5 v2/v1 are comparable. They reach around 80% on each of the eight exam topics. For this reason, we adopted Sonnet 3.7 as the judge model for AWS 500 Dataset to balance the performance and compute quota availability.

Among the eight exam topics, the advanced networking and DevOps (ANS-C01 and DOP-C02) are the two most challenging topics for most of the models considered in the current study.

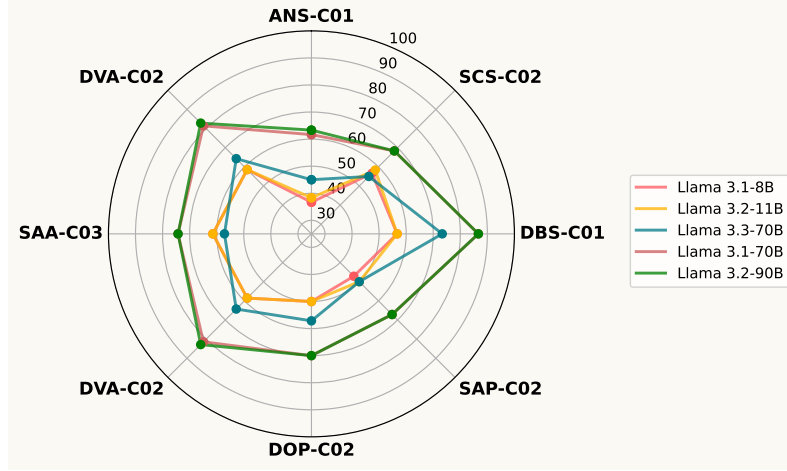
Table 2 shows the results of performance drop on AWS 500 dataset after applying the two-staged evaluation proposed in the current analyses. The performance drop varies drastically for different models. For Sonnet 4 the performance only drops 9.4% from 83.8% points to 74.4% point, indicating that Sonnet 4 can solve the AWS 500 questions correctly and reliably. For Llama 3.3 70B, the performance drop is 45.4% absolute points (from 70.4% to 25%).

Table 2: AWS 500 Dataset performance scores following the proposed two-staged evaluation approach

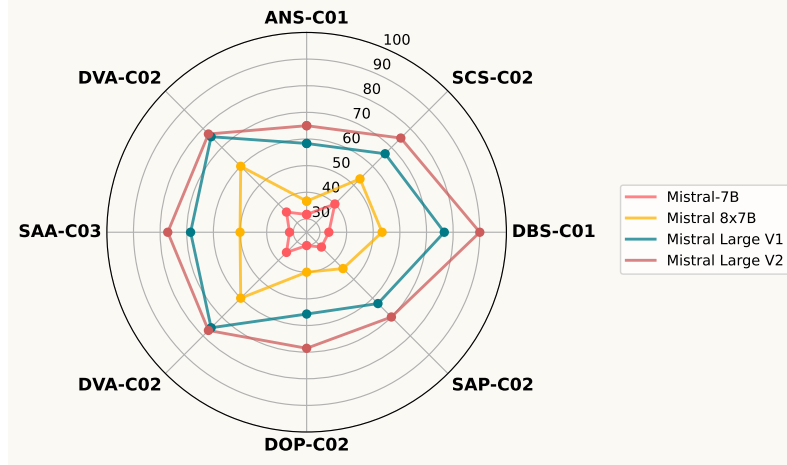
Model	Explanation Score (%)	Answer Score (%)	Full Score (%)	Δ
Sonnet 4	88.8	83.8	74.4	-9.4
Sonnet 3.7	83.7	83.4	69.8	-13.6
Sonnet 3.5 v2	81.3	88.8	72.2	-16.6
Sonnet 3.5 v1	78.5	87.4	68.6	-18.8
Mistral Large 2	74.6	74	55.2	-18.8
Opus 3	71.1	72.6	51.6	-21
Sonnet 3	64.7	62.4	40.4	-22
Mistral Large 1	61.3	65.2	40	-25.2
Claude 2	58.8	62.2	36.6	-25.6
Haiku 3	47.6	49.2	23.4	-25.8
Claude 2.1	58.7	63.4	37.2	-26.2
Llama 3.1 70B	60.2	68.8	41.4	-27.4
Claude Instant	47.1	52.6	24.8	-27.8
Llama 3.2 90B	60.2	70.8	42.6	-28.2
Mistral Small	46.4	53	24.6	-28.4
Mistral 8x7B	45.2	54.4	24.6	-29.8
Mistral 7B	28.5	42.8	12.2	-30.6
Llama 3.1 8B	35.6	52.2	18.6	-33.6
Haiku 3.5	55.1	77	42.4	-34.6
Llama 3.2 11B	30.5	53.2	16.2	-37
Llama 3.3 70B	35.5	70.4	25	-45.4



(a)



(b)



(c)

Figure 2: Model performances using zero-shot prompting, no CoT on all 21 LLMs from three model families. The radius represents the score and the eight angles represent eight exam topics as illustrated in Table 1. Note that ANS-C01, SCS-C02 and DBS-C01 are specialty level, SAP-C02 and DOP-C02 are professional level and the remaining are associates level.

D.2 GPQA Dataset Results

We conducted analyses of the performance drop from answer scores to full scores for all 21 LLMs tested, each under three different prompting configurations: zero shot (#1), five-shot without example reasoning traces (#2) and five-shot with example reasoning traces (#3), see Table 3 for details.

Table 3: GPQA performance scores following the proposed two-staged evaluation approach, where #1, #2 and #3 represent zero-shot, 5-shot without and with reasoning traces in examples, respectively. Δ shows the performance drop after enforcing explanation evaluations.

Model	Explanation Scores			Answer Scores			Full Scores			Δ (Answer – Full)		
	#1	#2	#3	#1	#2	#3	#1	#2	#3	#1	#2	#3
Sonnet 4	83.3	82.1	80	62.3	62.7	63.6	51.1	52.2	50.9	11.2	10.5	12.7
Sonnet 3.7	79.6	79.6	80.1	58.9	61.4	62.7	46.9	48.9	50.2	12.1	12.5	12.5
Sonnet 3.5 v2	76.3	81	82.1	54	58.5	56.3	43.8	44.6	46.2	10.3	13.8	10
Sonnet 3.5 v1	72.1	73.4	71.7	52.9	52	55.1	38.8	37.5	39.5	14.1	14.5	15.6
Haiku 3.5	46.2	42.4	57.3	38.4	37.7	36.6	16.3	17.4	21	22.1	20.3	15.6
Opus 3	59.1	62.1	63	48.9	48	47.1	30.4	28.4	29.7	18.5	19.6	17.4
Sonnet 3	47	47.1	49.4	38	40.4	38	17.9	19	18.8	20.1	21.4	19.2
Haiku 3	46.6	42	41	37.7	39.3	42	15.9	18.3	17.2	21.9	21	24.8
Claude 2.1	44.5	46	46.5	36.4	34.6	35.5	16.7	15.4	16.5	19.6	19.2	19
Claude 2	39.2	47.3	42.8	37.3	35.3	40.2	17.6	13.8	17.2	19.6	21.4	23
Claude Instant	35.5	41.1	36.8	28.8	30.8	34.6	11.8	10.9	12.7	17	19.9	21.9
Mistral Large v2	59.3	61.4	62.2	49.1	48.2	49.6	30.1	28.6	30.8	19	19.6	18.8
Mistral Large v1	45.7	47.6	52.9	37.1	36.6	38.4	17.6	16.7	20.3	19.4	19.9	18.1
Mistral Small	39.2	48.8	38.5	27.5	33	31.9	13.4	13	12.3	14.1	20.1	19.6
Mistral 8x7B	30.2	32.7	24.8	25.2	30.4	30.6	8.3	9.2	7.6	17	21.2	23
Mistral 7B	21.2	19	18.1	21.2	25.2	23.4	4	5.4	4.2	17.2	19.9	19.2
Llama 3.3 70B	46.3	46.1	60.1	48.9	50.7	49.8	22.5	23.4	29.9	26.3	27.2	19.9
Llama 3.2 90B	55.7	54.3	53.3	44.4	44.9	44	24.1	25	23.4	20.3	19.9	20.5
Llama 3.2 11B	26.7	33.3	18.2	30.8	30.1	31.9	10.3	8	5.8	20.5	22.1	26.1
Llama 3.1 70B	55.4	53.6	56.7	40.4	43.5	43.3	21.7	24.1	24.6	18.8	19.4	18.8
Llama 3.1 8B	31.9	29.2	22.2	30.6	32.1	30.1	8.9	10.3	6.7	21.7	21.9	23.4

E Example what could go wrong in the explanation

Question and Options

Methyl 2-oxocyclohexane-1-carboxylate is heated in the presence of aqueous NaOH. Then the reaction mixture is acidified with aqueous HCl, after which heating is continued. How many oxygen atoms are there in the main product of this reaction?

(A) 1 (B) 3 (C) 2 (D) 4

Ground truth answers

(A), the starting material will be saponified to form 2-oxocyclohexane-1-carboxylic acid, which will subsequently decarboxylate to form the final product cyclohexanone, which has Chemical Formula: C₆H₁₀O.

LLM answer and Reasoning

The given reaction involves heating methyl 2-oxocyclohexane-1-carboxylate in the presence of aqueous NaOH, followed by acidification with aqueous HCl and further heating. This reaction is a decarboxylation reaction, which results in the removal of a carboxyl group (COOH) from the molecule. The main product of this reaction is methyl cyclopentanone. Now, let's count the number of oxygen atoms in the main product. Methyl cyclopentanone has one oxygen atom in its carbonyl group (C=O).

Reasoning Evaluation by LLM Judge

Let me analyze this step by step.

First, I need to identify the question type. This is a conceptual-focused question that requires understanding of organic chemistry reaction mechanisms - specifically saponification followed by decarboxylation.

Looking at the golden reasoning: The process involves (1) saponification of the methyl ester to form 2-oxocyclohexane-1-carboxylic acid, then (2) decarboxylation to form cyclohexanone (C₆H₁₀O), which has 1 oxygen atom.

Now evaluating the test taker's response:

Positive aspects:

- Correctly identifies this as a decarboxylation reaction
- Understands that a carboxyl group (COOH) is removed
- Correctly counts that the final product has 1 oxygen atom
- Gets the right answer (A)

Critical errors:

- Incorrectly identifies the final product as "methyl cyclopentanone" instead of cyclohexanone
- This shows a fundamental misunderstanding of the structure - the starting material has a 6-membered ring (cyclohexane), not a 5-membered ring (cyclopentane)
- The "methyl" designation is also incorrect since the methyl ester group would be lost during saponification

The test taker demonstrates some understanding of the reaction type and can count oxygen atoms correctly, but shows a significant conceptual error in identifying the actual product structure. This represents a fundamental misunderstanding of the molecular transformation occurring.