# ARB: Advanced Reasoning Benchmark for Large Language Models

**Tomohiro Sawada**[1,2,†], **Daniel Paleka**[1,3], **Alexander Havrilla**[1,2], **Pranav Tadepalli**[1,2], **Paula Vidas**[1]

**Alexander Kranias**[1,2], **John J. Nay**[4,5], **Kshitij Gupta**[1,6], **Aran Komatsuzaki**[1,2,‡‡]

[1] DuckAI [2] Georgia Tech [3] ETH Zürich [4] Nomos AI
[5] Stanford University Center for Legal Informatics [6] MILA

## Abstract

Large Language Models (LLMs) have demonstrated remarkable performance on various quantitative reasoning and knowledge benchmarks. However, many of these benchmarks are losing utility as LLMs get increasingly high scores, despite not yet reaching expert performance in these domains. We introduce ARB, a novel benchmark composed of advanced reasoning problems in multiple fields. ARB presents a more challenging test than prior benchmarks, featuring problems in mathematics, physics, biology, chemistry, and law. As a subset of ARB, we introduce a challenging set of math and physics problems which require advanced symbolic reasoning and domain knowledge. We evaluate recent models such as GPT-4 and Claude on ARB and demonstrate that current models score well below 50% on more demanding tasks. In order to improve both automatic and assisted evaluation capabilities, we introduce a rubric-based evaluation approach, allowing GPT-4 to score its own intermediate reasoning steps. We find promising agreement between annotators and GPT-4 rubric evaluation scores.

## 1 Introduction

Models such as ChatGPT have shown the ability to pass entry-level examinations in fields such as law [Bommarito II and Katz, 2022], medicine [Kung et al., 2023], economics [Caplan, 2023], and mathematics [Shakarian et al., 2023]. Nevertheless, LLM understanding of many fields is reportedly shallow and unreliable [Shapira et al., 2023]. *Expert reasoning* in domains with specialized knowledge is essential for automated systems to augment skilled professionals [Noy and Zhang, 2023].

In this paper, we introduce a new benchmark dataset, **ARB** (**A**dvanced **R**easoning **B**enchmark), designed to evaluate expert reasoning abilities in mathematics, physics, chemistry, biology, and law.[*] To make the benchmark more challenging than previous benchmarks, we extract graduate-level tasks from resources intended for domain professionals. The mathematics and physics portions are substantially more difficult than popular benchmarks such as MATH Hendrycks et al. [2021].

Our dataset offers improvements over existing benchmarks:

- Hundreds of problems requiring expert reasoning in quantitative subjects, where LLMs are known to underperform;
- For mathematics and physics, all problems are short-answer and open-response questions, in contrast to the multiple-choice questions that dominated earlier benchmarks.

[†]Email: tsawada@gatech.edu.
[‡‡]Email: contact@duckai.org.
[*]Instructions for accessing the dataset can be found in the appendix.

37th Conference on Neural Information Processing Systems (NeurIPS 2023) Workshop on MATH-AI.

In addition, we propose an automated rubric-based method allowing self-evaluation of intermediate reasoning steps. While not currently a substitute for human evaluation, rubrics generated by GPT-4 have good coverage, and self-evaluation scores track human grading surprisingly well.

## 2 Related Work

Improving the reasoning capabilities of LLMs had been a subject of recent interest, with a particular focus on chain-of-thought (CoT) techniques [Wei et al., 2022, Kojima et al., 2023, Wang et al., 2023, Yao et al., 2023, Nye et al., 2021]. Such techniques have been successful in solving reasoning problems involving commonsense reasoning and basic mathematics, by inferring many tokens before yielding final answers. Models such as Minerva [Lewkowycz et al., 2022] demonstrate improvement of capabilities by fine-tuning on extensive datasets of mathematical writing. These works typically evaluate techniques against math benchmarks (e.g., GSM8K [Cobbe et al., 2021], MATH [Hendrycks et al., 2021], SVAMP [Patel et al., 2021], ASDiv [Miao et al., 2020], MMLU [Hendrycks et al., 2020], AQuA [Ling et al., 2017], MAWPS [Koncel-Kedziorski et al., 2016], MultiArith [Roy and Roth, 2016]) and commonsense reasoning tasks (e.g., CSQA [Talmor et al., 2018], StrategyQA [Geva et al., 2021], HotpotQA [Yang et al., 2018] ).

However, state of the art LLMs including GPT-4 [OpenAI, 2023] and Claude achieve big scores on most of these benchmarks. This underscores the need for rapid improvement in evaluation capability to accompany rapid improvement in LLM capability. Recently, several new benchmarks have been introduced for reasoning and planning tasks, such as the GPT-Planning Benchmark [Valmeekam et al., 2023], and ALERT Reasoning Benchmark [Yu et al., 2022]. Apart from using LLMs to generate evaluation tasks [Zhang et al., 2022, Perez et al., 2022], LLMs have increasingly been used as a proxy for human evaluation [Chiang and Lee, 2023, Liu et al., 2023, Fu et al., 2023, Kocmi and Federmann, 2023]. Useful LLM-based evaluation for alignment has been done using rubrics [Bai et al., 2022]. We explore the efficacy of rubrics for evaluation when applied to complex math and physics problems.

## 3 Benchmark Overview

ARB collects difficult, graduate and PhD level questions across multiple academic disciplines to test the limits of current LLM models' reasoning capabilities. We focus on math and physics, while also including law and MCAT questions to provide a more holistic benchmark.

Table 1: Types of problems in the benchmark by subject area.

| Subject | Answer Type | Number |
|---|---|---|
| Physics | Numerical | 113 |
| | Numerical (w/ image) | 18 |
| | Symbolic | 51 |
| | Symbolic (w/ image) | 13 |
| Mathematics | Numerical | 69 |
| | Symbolic | 52 |
| | Proof-like | 19 |
| Law | Multiple Choice | 627 |
| MCAT (Reading) | Multiple Choice | 165 |
| MCAT (Science) | Multiple Choice | 144 |
| | Multiple Choice (w/ image) | 37 |

## 4 Evaluation

We evaluated LLMs on text-only problems from our dataset. In total, we tested: GPT-3.5-turbo (`gpt3.5-turbo-0301`), GPT 3.5 (`text-davinci-003`), GPT-4 (`gpt-4-0314`), and Claude (`claude-v1.3-100k`). We used task-specific instructions and system prompts. Answers were ex-

tracted using the delimiter `ANSWER:` , and any response lacking it was marked incorrect. Evaluation procedures varied for different response types: see Appendix 6 for details.

### 4.1 Results

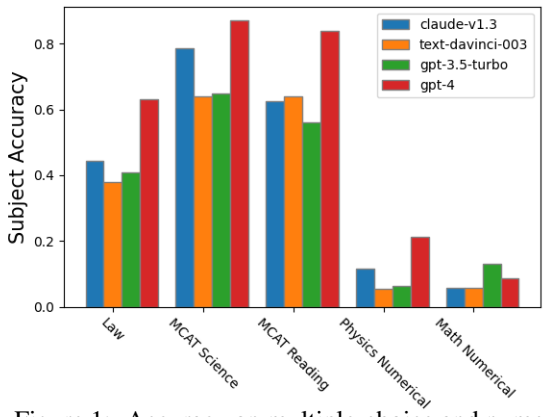


Figure 1: Accuracy on multiple-choice and numerical tasks (relative error threshold of $10^{-2}$).

Table 2: Manually parsed scores for Math Symbolic and Physics Symbolic.

| | Math | Physics |
|---|---|---|
| `gpt-4` | 15% | 20% |
| `gpt-3.5-turbo` | 12% | 8% |
| `text-davinci-003` | 17% | 6% |
| `claude-v1.3` | 10% | 12% |

Models perform well on multiple-choice Law and MCAT, but face difficulties with numerical answers. GPT-4 stands out in simplifying complex expressions, but has issues with arithmetic over long contexts. For multiple-choice questions, only `gpt-3.5-turbo` often fails to follow answer formatting, especially on the Law benchmark, with a 25% unparsable rate. Other models face this issue in under 5% of questions. Symbolic problem performance is generally low across the board, as shown in Table 2. It is possible (and likely) that future prompting methods, especially equipped with external mathematical tools, will get better performance on the benchmark.

As mentioned in Section 2, benchmarks with very high scores are less useful for differentiating model capabilities. The same holds for benchmarks with very low scores across the board. On Math Numerical, GPT-4 has slightly lower accuracy than `gpt-3.5-turbo` on our run; similarly, `text-davinci-003` has similar accuracy as GPT-4 on Math Symbolic. After inspection, this is a combination of two factors: our dataset having several answers exactly 0 (or $\mathbb{Z}$) and weaker models "guessing" correctly; and the memorization / faithful reasoning tradeoff discussed in Section 4.2.

### 4.2 Memorization

While it is difficult to quantify problem leakage and memorization of similar problems, some outputs suggest this might be a relevant factor. In Table 12, GPT-3.5-turbo appears to "know" the answer to the problem, given that it gets the formula for the capacitance $C$ correct despite the wrong reasoning. In contrast, GPT-4 stays faithful to its reasoning chain, resulting in a wrong final answer. Note that the overall performance of all models is still low, hence not majorly affected by memorization.

### 4.3 What kind of errors does GPT-4 make?

The GPT-4 evaluation paper [Bubeck et al., 2023] classified errors GPT-4 makes in single-pass evaluation on GSM8K [Cobbe et al., 2021] and MATH [Hendrycks et al., 2021] into three types: *arithmetic mistakes*, *misunderstood statement*, and *wrong approach*. We make a more fine-grained analysis and extended it to a sample (see Appendix 9) of math and physics problems in our dataset:

- Misunderstanding / answering only a part of the question / misread problem;
- Wrong approach: the model's early chain of thought does not guess the right approach;
- Logical errors: the model uses a false implication between two statements;
- Hallucinating facts or theorems: the model confabulates a false/inapplicable statement;
- Arithmetic error: the model multiplies incorrectly, omits a term in an expression, etc.

Table 3: Mistakes on mathematics and physics problems in ARB, GPT-4.

| | Misread problem | Wrong approach | Logical error or hallucination | Arithmetic mistake | Correct answer | Correct reasoning |
|---|---|---|---|---|---|---|
| Math Numerical | 0% | 25% | 88% | 48% | 3% | 3% |
| Math Symbolic | 16% | 50% | 29% | 4% | 16% | 16% |
| Math Proof-like | 5% | 50% | 72% | 16% | n/a | 5% |
| Physics Numerical | 0% | 80% | 53% | 6% | 6% | 6% |
| Physics Symbolic | 0% | 37% | 68% | 31% | 28% | 12% |

We graded GPT-4 using the above as a guideline. Our grading of the model's CoT answers is not mutually exclusive. Note that the errors might not be independent: arithmetic mistakes could be more or less frequent in wrong approach solutions as opposed to the solutions with the correct idea.

When the model outputs a chain of implications, it is not always clear whether some false statement is due to a logical error, or it is a straight-out confabulation. We merge those two error types in Table 3.

Where applicable, slight discrepancy with automatic evaluation is also possible due to the error tolerance. It is possible that our graders underestimate the rate of arithmetic mistakes in when the approach is clearly wrong. We note that many of the problems in Physics Symbolic have correct answers even when there are flaws in the chain of thought reasoning. This is likely due to some kind of memorization, although not necessarily from the same sources: see Table 12 for an example.

### 4.4 Model-based Rubric Evaluation

As task complexity increases, evaluating the CoT reasoning becomes challenging. Ideally, LLMs would grade against a reference solution, but this has reliability issues. To improve this, we introduce *rubrics* as an intermediate step: the model forms a rubric from the reference and grades using it.

We rate GPT-4's rubrics manually, with results shown in Table 4. GPT-4's rubrics capture key solution steps, but sometimes misallocate point values. This improves over GPT-3.5-turbo, which often gives points to irrelevant steps. A clear limitation of rubric scoring is overlooking solutions different from the reference solution. However, on ARB, GPT-4 rarely produces a correct solution outside the rubric.

Table 4: Evaluations of rubric quality and GPT-4 rubric evaluation failure cases. Rubric coverage and rubric point spread are on a 1-5 Likert scale. Alternative solutions count solutions that are correct and not covered by the rubric. Extra/reduced credit track how often GPT-4 erroneously assigns or deducts points; hallucinated tracks how often GPT-4 gives points not actually present in the rubric.

| | Physics Symbolic | Math Symbolic | Proof-like |
|---|---|---|---|
| Rubric coverage | 4.42 | 4.26 | 3.94 |
| Rubric point spread | 4.16 | 4.00 | 4.06 |
| Alternative solutions | 5% | 2% | 0% |
| Extra credit | 27% | 18% | 40% |
| Reduced credit | 11% | 12% | 5% |
| Hallucinated rubric | 0% | 15% | 0% |

Table 5: Average scores (out of 10 points) when assigned by human annotators versus GPT-4. Correlation is the Pearson correlation coefficient between the two scores, over all problems.

| | Physics Symbolic | Math Symbolic | Proof-like |
|---|---|---|---|
| Human eval score | 5.00 | 3.13 | 2.65 |
| Model eval score | 5.05 | 3.37 | 3.80 |
| Correlation | 0.91 | 0.78 | 0.82 |

To distinguish two failure modes of creating the rubric and rubric-based grading, we manually graded GPT-4's solutions based on its rubrics and compared to our full method (see Table 5). GPT-4's

evaluation score correlates closely with the manual score, occasionally deviating by a point or two. The biggest discrepancy occurs when the model awards partial credit for entirely incorrect solutions, while human scores remain zero. These findings suggest that rubric-based evaluation is a promising automated evaluation method.

## References


Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, ..., and Jared Kaplan. Constitutional AI: Harmlessness from AI feedback, 2022.

Michael Bommarito II and Daniel Martin Katz. GPT takes the bar exam. *arXiv preprint arXiv:2212.14402*, 2022.

Volodymyr Brayman and A. G. Kukush. *Undergraduate Mathematics Competitions (1995-2016): Taras Shevchenko National University of Kyiv*. Springer, 2018.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with GPT-4. *arXiv preprint arXiv:2303.12712*, 2023.

Candice McCloskey Campbell, Shaun Murphree, Jennifer M. Warner, Amy B. Wachholz, Kathy A. Zahler, and George J. Hademenos. *McGraw-Hill Education 3 MCAT Practice Tests, Third Edition*. McGraw-Hill Education, Jan 2017. ISBN 1259859622.

Bryan Caplan. GPT retakes my midterm and gets an A, 2023. URL `https://betonit.substack.com/p/gpt-retakes-my-midterm-and-gets-an`.

Ji-Xiu Chen and Daqian Li. *Problems and solutions in Mathematics*. World Scientific, 1998.

Cheng-Han Chiang and Hung-yi Lee. Can Large Language Models be an alternative to human evaluations? *arXiv e-prints*, art. arXiv:2305.01937, may 2023. doi: 10.48550/arXiv.2305.01937.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021. URL `https://arxiv.org/abs/2110.14168`.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. GPTScore: Evaluate as you desire. *arXiv e-prints*, art. arXiv:2302.04166, feb 2023. doi: 10.48550/arXiv.2302.04166.

Răzvan Gelca and Titu Andreescu. *Putnam and beyond*. Springer, 2017.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies, 2021.

Department of Mathematics Harvard University. Qualifying examination for fall 2021, Aug 2021. URL `https://www.math.harvard.edu/media/quals-F21_with_solutions.pdf`.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2020. URL `https://arxiv.org/abs/2009.03300`.

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. *CoRR*, abs/2103.03874, 2021. URL `https://arxiv.org/abs/2103.03874`.

Tom Kocmi and Christian Federmann. Large language models are state-of-the-art evaluators of translation quality. *arXiv e-prints*, art. arXiv:2302.14520, feb 2023. doi: 10.48550/arXiv.2302.14520.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners, 2023.

Rik Koncel-Kedziorski, Subhro Roy, Aida Amini, Nate Kushman, and Hannaneh Hajishirzi. MAWPS: A math word problem repository. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1152–1157, San Diego, California, jun 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1136. URL `https://aclanthology.org/N16-1136`.

Tiffany H Kung, Morgan Cheatham, Arielle Medenilla, Czarina Sillos, Lorie De Leon, Camille Elepaño, Maria Madriaga, Rimel Aggabao, Giezel Diaz-Candido, James Maningo, et al. Performance of ChatGPT on USMLE: Potential for AI-assisted medical education using large language models. *PLOS Digital Health*, 2 (2):e0000198, 2023.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models, 2022. URL https://arxiv.org/abs/2206.14858.

Swee Cheng Lim, Choy Heng Lai, and Leong Chuan Kwek. *Problems and solutions on optics*. World Scientific, 2019.

Yung-kuo Lim. *Problems and solutions on thermodynamics and Statistical Mechanics*. World Scientific, 1996.

Yung-kuo Lim. *Problems and solutions in quantum mechanics: Major, American universities ph. D. qualifying questions and, solutions*. World Scientific, 1998.

Yung-kuo Lim. *Problems and solutions on electromagnetism*. World Scientific Pub. Co, 2007.

Yung-kuo Lim and Yuan-qi Qiang. *Problems and solutions on Mechanics*. World Scientif., 2001.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. Program induction by rationale generation: Learning, to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167, Vancouver, Canada, jul 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1015. URL https://aclanthology.org/P17-1015.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: NLG evaluation using GPT-4 with better human alignment. *arXiv e-prints*, art. arXiv:2303.16634, mar 2023. doi: 10.48550/arXiv.2303.16634.

Aaron Meurer, Christopher P. Smith, Mateusz Paprocki, Ondřej Čertík, Sergey B. Kirpichev, Matthew Rocklin, AMiT Kumar, Sergiu Ivanov, Jason K. Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian E. Granger, Richard P. Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, Matthew J. Curry, Andy R. Terrel, Štěpán Roučka, Ashutosh Saboo, Isuru Fernando, Sumith Kulal, Robert Cimrman, and Anthony Scopatz. Sympy: Symbolic computing in python. *PeerJ Computer Science*, 3: e103, jan 2017. ISSN 2376-5992. doi: 10.7717/peerj-cs.103. URL https://doi.org/10.7717/peerj-cs.103.

Shen-yun Miao, Chao-Chun Liang, and Keh-Yih Su. A diverse corpus for evaluating and developing English math word problem solvers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 975–984, Online, jul 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.92. URL https://aclanthology.org/2020.acl-main.92.

Shakked Noy and Whitney Zhang. Experimental evidence on the productivity effects of generative artificial intelligence. *Available at SSRN 4375283*, 2023.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models, 2021.

OpenAI. GPT-4 technical report, 2023.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are NLP models really able to solve simple math word problems?, 2021.

Ethan Perez, Sam Ringer, Kamilė Lukošiūtė, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, et al. Discovering language model behaviors with model-written evaluations. *arXiv preprint arXiv:2212.09251*, 2022.

Subhro Roy and Dan Roth. Solving general arithmetic word problems, 2016.

Paulo Shakarian, Abhinav Koyyalamudi, Noel Ngu, and Lakshmivihari Mareedu. An independent evaluation of ChatGPT on mathematical word problems (MWP). *arXiv preprint arXiv:2302.13814*, 2023.

Natalie Shapira, Mosh Levy, Seyed Hossein Alavi, Xuhui Zhou, Yejin Choi, Yoav Goldberg, Maarten Sap, and Vered Shwartz. Clever Hans or Neural Theory of Mind? Stress testing social reasoning in large language models. *arXiv preprint arXiv:2305.14763*, 2023.

Paulo N de Souza and Jorge N. Silva. *Berkeley problems in Mathematics*. Springer New York, 2008.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. *CoRR*, abs/1811.00937, 2018. URL http://arxiv.org/abs/1811.00937.

Karthik Valmeekam, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. Large language models still can't plan (a benchmark for LLMs on planning and reasoning about change), 2023.

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

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models, 2022. URL https://arxiv.org/abs/2201.11903.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering, 2018.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models, 2023.

Ping Yu, Tianlu Wang, Olga Golovneva, Badr Alkhamissy, Gargi Ghosh, Mona Diab, and Asli Celikyilmaz. ALERT: Adapting language models to reasoning tasks, 2022.

Sarah Zhang, Reece Shuttleworth, Derek Austin, Yann Hicke, Leonard Tang, Sathwik Karnik, Darnell Granberry, and Iddo Drori. A dataset and benchmark for automatically answering and generating machine learning final exams. *arXiv preprint arXiv:2206.05442*, 2022.

Hefei Zhongguo-Kexue-Jishu-Daxue. *Major American universities Ph. D. qualifying questions and solutions. 5. Problems and solutions on thermodynamics and statistical mechanics*. World Scientific, 1990.

# 5 Benchmark Format and Sources

## 5.1 Output Formats

The benchmark consists of three types of questions: multiple choice, short answer, and open response, in descending order of proportion in the dataset.

- **Multiple choice** questions consist of a question and four to five possible answers, and the correct answer is the one that best answers the question. Those are only included in the MCAT and Law portions of the dataset.
- **Short answer questions**, on the other hand, ask for final answers in the format of a short phrase or mathematical expression. They were sourced from problem books such as Souza and Silva [2008], Gelca and Andreescu [2017], and physics book series Lim and Qiang [2001], Lim [2007], Lim [1998], Lim et al. [2019], and Lim [1996]. We generally avoided including nontrivial algebraic expressions due to technical difficulties in the grading process.
- **Open response** questions are more challenging and always require manual grading. Those were sourced from problem books and exams, such as the Harvard PhD comprehensive exams in mathematics [Harvard University, 2021].

## 5.2 Mathematics

This part of the dataset includes mainly problems taughs in univesities at undergraduate and introductory graduate level, as well as some college contest problems. Contest problems are sourced from Gelca and Andreescu [2017] and Brayman and Kukush [2018], and university mathematics problems are sourced from Souza and Silva [2008], Chen and Li [1998] and Harvard University [2021]. The dataset does not include high school contest problems because those are already covered in other well-known benchmarks [Hendrycks et al., 2021]. We also select Harvard University [2021] because it covers topics that other collections of exams rarely cover, such as representation theory of finite groups and algebraic topology. Most sources contain official solutions, which we also include in the dataset. This can be useful for automating the grading process, which we explore in Section 4.4.

### 5.3 Physics

The physics problems are structured similarly as the math problems. The main difference is that some physics problems contain figures, and there are more problems with numerical answers. The problems were sourced from the Major American Universities PhD Qualifying Questions and Solutions series [Zhongguo-Kexue-Jishu-Daxue, 1990].

### 5.4 MCAT

The MCAT test contains multiple choice problems testing biology, psychology, chemistry, physics, and reading comprehension. The MCAT problems are sampled from the third edition of McGraw-Hill Education 3 MCAT Practice Tests [Campbell et al., 2017] As in the previous categories, we pick problems which are self-contained. As in Physics, some MCAT problems are accompanied by figures.

### 5.5 Law

Application of legal knowledge to a particular scenario requires logical reasoning. Most lawyers in the U.S. go to law school, graduate, then study for the Bar Examination, and then must pass the bar before going on to practice law professionally. To evaluate legal understanding of the models, we use an older Bar Examination practice set that is less likely to be available online in a way that could have led to its legal inclusion in training data for the language models that we are assessing.

## 6 Evaluation Details

Temperature in all experiments was set to 0.7.

**Multiple Choice**: Extracted answers (letters) were matched to ground truth for correctness.

**Numerical**: Answers, often in units, were extracted and units removed using regex. Similarly to [Lewkowycz et al., 2022], we utilized Python's SymPy library [Meurer et al., 2017] for parsing. Correctness was determined using a relative error threshold of $10^{-2}$.

**Symbolic**: Again using SymPy, we attempt to check the equivalence of symbolic equations. However, not all symbolic answers can be checked in this way, requiring human evaluation. In addition to human evaluation, we explore using ChatGPT to check the equivalence of symbolic final answers.

**Proof-like**: Mathematical proofs were evaluated manually by authors with mathematical training, due to the difficulty of automating the process reliably (see Section 4.4 for more details).

## 7 Dataset format

The benchmark dataset is available in .jsonl format, containing problem statements, ground truth solutions, and final ground truth answers for each entry. We additionally include metadata such as subject names and problem topics, where available.

## 8 Accessing the dataset

We facilitate access to the dataset through API calls [†]. The problems are separated by subject area and output type.

We use an API to host the data (rather than making it publicly available on popular platforms such as Hugging Face or GitHub) out of concern for data contamination. Many models released in the last few years have been trained on benchmark datasets [OpenAI, 2023], an approach that could inadvertently inflate the model's performance on the associated benchmark. There are other proposed strategies to mitigate this problem, including dataset poisoning and canary text methods. We choose restricting dataset accessibility to plain web crawlers to ensure that the dataset must have been deliberately downloaded by the model creators to end up in the training data.

[†]The documentation is hosted here: https://advanced-reasoning-benchmark.netlify.app/docs. The root URL for the API requests is https://advanced-reasoning-benchmark.netlify.app/api/lib/. To download problems, please use endpoints as described in the documentation.

## 9 Error Analysis Details

The distribution of problems might be representative only of a subset of the entire dataset, because the grading was done before the dataset was finalized; the problems added later are tagged as *"additional"* in the dataset entries. For the Symbolic and Numerical subsets (see Table 1), we subsample the problems to between 20 and 40 per subject area to minimize human grading effort. this is enough for a ballpark estimate of the frequency of different errors, and is not worth increasing because attributing error types is inherently fuzzy.

## 10 Intended Use and Licensing Information

**Dataset Intended Uses.** The ARB benchmark dataset, as documented in the paper, is primarily intended for research purposes. We do not condone using this dataset to train models that help students cheat on exams or for other commercial purposes. We hope that the research community will use this benchmark to better assess reasoning capabilities of language models.

**Author Statement and License.** The ARB dataset is licensed under CC BY 4.0, and all helper code we release or expose via the API is under the MIT license. For all problems originating in books listed in Section 3, following [Hendrycks et al., 2021], we abide by Fair Use §107: "the fair use of a copyrighted work, including such use by ... scholarship, or research, is not an infringement of copyright", where fair use is determined by "the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes" and "the effect of the use upon the potential market for or value of the copyrighted work".

## 11 Prompts for Section 4

For OpenAI chat models (`gpt-4` and `gpt-3.5-turbo`), we prepend the system prompt following best practices. We do not use the system prompt for the other models, due to there not being a well-established way to do so, and out of concerns of hurting performance.

Table 6: Prompt used for multiple-choice MCAT and Law problems.

| | |
|---|---|
| System | You are a top graduate student taking an open-ended qualifying exam. Your final answer should always be in the last line of your response, preceded by ANSWER:. |
| User | You are a top graduate student taking a qualifying exam. Below you will find a multiple choice question.<br><br>Question: *{Problem_Statement}*<br><br>Answer Choices: *{Answer_Choices}*<br><br>Now it is time to choose an answer. Think carefully and go step by step.<br>Make sure to justify all your work. Your final answer should be one of A,B,C,D,... given at the end of your work and preceded by ANSWER:. For example, if you think the answer is B, the last line of your answer should be ANSWER: B<br><br>Solution: |

Table 7: Prompt used for numerical problems.

| | |
|---|---|
| System | You are a top graduate student taking an open-ended qualifying exam. Your final answer should always be in the last line of your response, preceded by ANSWER:. |
| User | You are a top graduate student taking an open-ended qualifying exam. Below you will find a question requiring you to compute a numerical value.<br><br>Question: *{Problem_Statement}*<br><br>Now it is time to give your answer. Think carefully and go step by step. Make sure to justify all your work. Please simplify all expressions as much as possible and do not leave any variables in your final answer.<br>Your final answer should NOT contain units and should be given at the end of your work and preceded by ANSWER:<br>For example, if you think the answer is $2.4$ meters, the last line of your answer should be ANSWER: $2.4$.<br><br>Solution: |

Table 8: Prompt used for symbolic problems.

| | |
|---|---|
| System | You are a top graduate student taking an open-ended qualifying exam. Your final answer should always be in the last line of your response, preceded by ANSWER:. |
| User | You are a top graduate student taking an open-ended qualifying exam. Below you will find a question requiring you to give a symbolic answer.<br><br>Question: *{Problem_Statement}*<br><br>Now it is time to give your answer. Think carefully and go step by step. Make sure to justify all your work.<br>Your final answer should NOT contain units and should be given at the end of your work and preceded by ANSWER:<br>For example, if you think the answer is $x * y$, the last line of your answer should be ANSWER: $x * y$<br><br>Solution: |

Table 9: Prompt used for proof-like problems.

| | |
|---|---|
| System | You are a top graduate student taking an open-ended qualifying exam. Your final answer should always be in the last line of your response, preceded by ANSWER:. |
| User | You are a top graduate student taking an open-ended qualifying exam. Below you will find a question requiring you to prove the given statement.<br><br>Question: *{Problem_Statement}*<br><br>Now it is time to give your answer. Think carefully and go step by step. Make sure to justify all your work.<br><br>Solution: |

Table 10: Prompt used for GPT-4 rubric self-evaluation.

| | |
|---|---|
| System | You are a top professor grading an open-ended qualifying exam. |
| User | Problem Statement: *{Problem_Statement}*<br>Rubric: *{rubric}*<br>Student Answer: *{response}*<br><br>Now it is time to grade the student answer. Make sure to check each point of the rubric step by step. And make sure to print the total number of earned points at the end of your grading. For example, if the student earned 8 points, print Rubric Score: 8 points<br><br>Rubric Evaluation: |

Table 11: Prompt used for GPT-4 rubric design.

| | |
|---|---|
| System | You are a top professor grading an open-ended qualifying exam. |
| User | Problem Statement: *{Sample_Problem_Statement}*<br>Reference Solution: *{Sample_Reference_Solution}*<br>Rubric: *{Handcrafted_Rubric}*<br><br>Problem Statement: *{ Problem_Statement }*<br>Reference Solution: *{ Solution }*<br><br>Now it is time to write the rubric. Make sure to think carefully and go step by step, breaking down the problem into multiple parts. The total number of possible points should sum to 10.<br><br>Rubric: |

Table 12: Example of memorization in ARB. GPT-3.5-turbo's response indicates memorization of the correct expression. GPT-4 commits to its reasoning better, resulting in the wrong final answer.

| | |
|---|---|
| Problem | A cylindrical capacitor of length $L$ consists of an inner conductor wire of radius $a$, a thin outer conducting shell of radius $b$. The space in between is filled with nonconducting material of dielectric constant $\varepsilon$.<br>Find the capacitance. |
| Reference Solution | Supposing that the charge per unit length of the inner wire is $-\lambda$ and using cylindrical coordinates $(r, \varphi, z)$, we find the electric field intensity in the capacitor by Gauss' theorem to be<br>$$\mathbf{E} = -\frac{\lambda}{2\pi\varepsilon r}\mathbf{e}_r = \frac{-Q}{2\pi\varepsilon L r}\mathbf{e}_r.$$<br>The potential difference between the inner and outer capacitors is<br>$$V = -\int_a^b \mathrm{E}\cdot d\mathrm{r} = \frac{\lambda}{2\pi\varepsilon}\ln\left(\frac{b}{a}\right).$$<br>Hence the capacitance is<br>$$C = \frac{\lambda L}{V} = \frac{2\pi\varepsilon L}{\ln\left(\frac{b}{a}\right)}$$ |
| `gpt-3.5-turbo` Solution | The capacitance of a cylindrical capacitor is given by the equation:<br>$$C = \frac{2\pi\varepsilon L}{\ln(b/a)}$$<br>where $\varepsilon$ is the dielectric constant, $L$ is the length of the capacitor, $a$ is the radius of the inner conductor, and $b$ is the radius of the outer conductor.<br>Therefore, the capacitance of the given cylindrical capacitor is:<br>$$C = \frac{2\pi\varepsilon L}{\ln(b/a)} = \frac{2\pi\varepsilon L}{\ln(b) - \ln(a)}$$<br>ANSWER: $\frac{2\pi\varepsilon L}{\ln(b)-\ln(a)}$ |

Table 12 (continued)

To find the capacitance of the cylindrical capacitor, we will use the general formula for capacitance: $C = \frac{Q}{V}$, where $Q$ is the charge stored in the capacitor and $V$ is the potential difference between the inner conductor wire and the outer conducting shell. First, let's find the electric field $E$ between the two conductors. We can use Gauss's law, which states that the electric flux through a closed surface is equal to the charge enclosed by the surface divided by the permittivity of free space $\varepsilon_0$:

$$\oint \vec{E} \cdot d\vec{A} = \frac{Q_{enclosed}}{\varepsilon_0}.$$

We will choose a cylindrical Gaussian surface of radius $r$ and length $L$, where $a < r < b$. The electric field is radial, so the electric flux through the curved surface of the cylinder is:

$$\oint \vec{E} \cdot d\vec{A} = E \oint dA = E(2\pi r L).$$

The charge enclosed by the Gaussian surface is the charge on the inner conductor wire, which is given by its linear charge density $\lambda$ multiplied by the length of the Gaussian surface, $L$: $Q_{enclosed} = \lambda L$.
Now we can use Gauss's law to find the electric field:

$$E(2\pi r L) = \frac{\lambda L}{\varepsilon_0} \Rightarrow E = \frac{\lambda}{2\pi r \varepsilon_0}.$$

gpt-4 Solution

However, we need to account for the dielectric material in between the conductors. The electric field in a dielectric material is related to the electric field in vacuum by the dielectric constant $\varepsilon$:

$$E_{dielectric} = \frac{E}{\varepsilon}.$$

So, the electric field in the dielectric material is:

$$E_{dielectric} = \frac{\lambda}{2\pi r \varepsilon_0 \varepsilon}.$$

Now we can find the potential difference between the inner conductor wire and the outer conducting shell by integrating the electric field along the radial direction:

$$V = -\int_a^b E_{dielectric} dr = -\int_a^b \frac{\lambda}{2\pi r \varepsilon_0 \varepsilon} dr.$$

Integrating, we get:

$$V = -\frac{\lambda}{2\pi \varepsilon_0 \varepsilon} \ln(\frac{b}{a}).$$

Finally, we can find the capacitance using the formula $C = \frac{Q}{V}$:

$$C = \frac{\lambda L}{-\frac{\lambda}{2\pi \varepsilon_0 \varepsilon} \ln(\frac{b}{a})}.$$

Simplifying, we get:
ANSWER: $C = \frac{2\pi \varepsilon_0 \varepsilon L}{\ln(\frac{b}{a})}$.