TAU-Exam: A LLM Benchmark for Theoretical Computer Science Reasoning

Orr Eilat Abdo Amer

Abstract

This paper introduces the TAU dataset, a benchmark derived from Tel Aviv University's Data Structures course exams, designed to evaluate the reasoning and problem-solving capabilities of Large Language Models (LLMs) in theoretical computer science. It differs from contemporary datasets both in its topic and in the abilities required for solving it correctly. In addition, we evaluate two well-known open source models on this dataset, and show that current LLMs struggle with this type of questions.

1 Introduction

In recent years, Large Language Models (LLMs) have demonstrated remarkable success across various tasks, such as commonsense reasoning (Yang et al., 2018), coding (Jain et al., 2024; Chai et al., 2024a), mathematics (Satpute et al., 2024; Chai et al., 2024b), and dialogue systems (Young & Shishido, 2023). However, despite the growing capabilities of these models, the question remains whether scaling up their parameters and expanding training datasets will show diminishing returns, and what are the domains in which LLMs stil struggle. related to this question, is the need for new benchmarks that can better measure LLMs' performance and incentivize researchers for improving LLMs' abilities in new domains.

In this paper, we propose the **TAU dataset**, a set of questions derived from exams in the Tel Aviv University Data Structures course. These questions are designed not only to assess knowledge of data structures and algorithms but also to measure students ability to abstract, combine, and apply this knowledge in novel ways. We hypothesize that the ability to solve these types of questions is crucial for LLMs as well, making this a valuable benchmark for testing their reasoning and problem-solving capabilities.

The introduction of such a benchmark is particularly timely in the context of computer science education. The integration of LLMs in educational settings holds significant promise for transforming how knowledge is taught and assessed. By focusing on benchmarks like the TAU dataset, we can create more dynamic and challenging learning environments that go beyond rote memorization and foster deeper understanding. LLMs, if capable of solving these benchmarks, could play a transformative role in shaping the future of education by providing personalized feedback, enhancing course content, and supporting adaptive learning methods.

To assess the current capabilities of LLMs in this area, we evaluate two well-known open-source models—LLaMa 3 8B (Meta, 2024) and MathΣtral 7B (Mistral, 2024) by Mistral AI — on the TAU dataset. Our results reveal that both models achieve low scores, suggesting that the challenge of solving such complex questions remains far from solved. This indicates that significant progress is needed to improve LLMs, not only in terms of their accuracy but also in their ability to adapt and apply knowledge in novel ways.

The primary contributions of this paper are:

1. **Introducing the TAU dataset**, a new benchmark for evaluating reasoning on theoretical

data structures and algorithms problems.

2. **Evaluating contemporary LLM performance** on this benchmark, showing the current limitations of these models.
3. **Highlighting the potential for evolving computer science education** through the integration of LLMs in the education process.

Through this work, we aim to inspire further research into developing more sophisticated benchmarks and advancing LLMs, with the ultimate goal of enhancing computer science education and research. By addressing these challenges, we can contribute to the evolution of educational practices and better prepare students for the rapidly changing technological landscape.

2 Related Work

In recent years, LLMs have shown high performance on pre-LLM NLP reasoning benchmarks such as GLUE (Wang et al., 2018) and Wingorad (Levesque et al., 2011). In addition, LLMs have shown remarkable capabilities in areas like coding, math and science. This rapid progress requires new benchmarks that can measure LLM abilities in these new fields.
Some of the new LLM-related benchmarks include:

Mathematical reasoning benchmarks GSM8K (Cobbe et al., 2021) focuses on grade-school level math problems, with mostly arithmetic operations. MATH (Hendrycks et al., 2021), a harder benchmark, measures mathematical problem solving in competition-level math problems and more math benchmarks.

Coding Benchmarks CodeBLEU (Ren et al., 2020) adapts the BLEU metric (Papineni et al., 2002) for code generation evaluation. CodeXGLUE(Lu et al., 2021) measures understanding and generation. The HumanEval benchmark (Chen et al. 2021) by OpenAI measures functional correctness of code generated from docstrings. MBPP (Austin et al., 2021) focuses on entry-level short programms from natural language descriptions. DS-1000 (Lai et al. 2022) specializes in measuring code generation for data-science tasks. MTPB (Nijkamp et al. 2022) introduces problem sets that are factorized into multi-turn prompts.

Reasoning in the domain of computer science Even though benchmarks have been released to measure LLM reasoning in many domains, the domain of computer science remains unexplored. Our benchamrk aims to measure the reasoning capabilities of LLMs in the theoretical topics studied in higher education. Our problem sets differ from the coding benchmarks, in that they focus on understanding abstract concepts and applying them in undergraduate exam-level questions.

3 The Dataset

Dataset Origin The benchmark questions are taken from the quizzes and exams given in the data Structures course in Tel Aviv University, between the years 2001-2018. The questions are based on the course material, which overlaps with the material in most data structures and algorithm courses in higher education programs around the world. Solving the questions correctly requires a combination of understanding the principles behind it, and being able to apply them in novel ways.

Question Types We partitioned the data into 4 types:

- **A** Open questions.
- **B** Closed questions with short explanations required.
- **C** Multiple choice with short explanations required.
- **D** Multiple choice – no explanations.

Question Topics All questions were taken from past Data Structures course exams. Each question was tagged as belonging to one or more of 16 topics. The topics, their frequencies and their identifying letter in our dataframe are given in Table 1.

Solutions Solutions are provided to 63% of the questions. The solutions are not full solutions, but rather short and concise answers provided by the course staff to the students after the exams.

Translation The questions, originally in Hebrew, were translated to English by calling the OpenAI API (`version: gpt-4o-2024-08-06`).

Data Contamination Because of the translation, the questions translated via the API are at risk of contamination. That is, being used in the training data of future models. To minimize the impact of this, we shuffled the questions and answers, and sent them to translation separately.

4 Experimental Setup

Models We evaluate two well-known LLMs:

- **LLaMA 3 8B (Meta, 2024)**: an open-source, decoder-only language model developed by Meta as part of the LLaMA (Large Language Model Meta AI) series. This model, part of the third-generation LLaMA family, comes with an 8 billion parameter count, optimized to balance performance and computational efficiency for diverse natural language processing tasks. Like its predecessors, LLaMA 3 is pretrained using a standard language modeling objective, drawing from an extensive text corpus to enhance its generative abilities across various applications.
- **MathΣtral 7B (Mistral, 2024)**: an inference-only, open-source decoder-based language model released by Mistral. With 7 billion parameters, MathΣtral is specifically optimized for mathematical reasoning and problem-solving tasks. Its architecture and training data are designed to enhance performance on computations and symbolic mathematics, making it suitable for high-precision, inference-based applications without the need for further fine-tuning.

To ensure a fair comparison, we use the same task-specific prompt templates and the same random seeds for both models. Question types Evaluated In the evaluation, we used only multiple-choice questions (types C and D).

Seed Each prompt was sent to the models 5 times, with seeds 0-4.

Prompting We tested two prompting methods: *Chain of Thought* and *none*. In each run, the model was initially given this prompt, followed by the exam question. In the CoT setting, the suffix "Solve it step by step." was added after the question. in the none setting, the prompt was passed without the suffix:

First Prompt

”Here is a multiple-choice/true-false question from the Data Structures course. Solve the question.”
+
Question
+
”Solve it step by step.” (optional)

After receiving the model's answer, a second prompt was sent to the model, asking it to write the single digit or letter of the correct answer.

The model's response for the second prompt, should summarize the solution and return a single digit or letter.

Second Prompt

”Write the final answer as a single digit or letter, without additional text.”

Evaluation Metric The model's output after the second prompt (a single letter or digit) was automatically compared to the target answer using

Id Letter	Topic	Percentage
a	Complexity analysis	15.18%
f	Balanced BST (AVL, B)	7.59%
g	Hash tables	6.32%
k	Lower bound for comparison-based sorting	5.90%
h	Binary heaps	5.82%
p	OTHER	5.56%
e	Binary search trees	5.56%
i	Binomial heaps, Fibonacci heaps	5.14%
d	Arrays and linked lists	4.89%
m	Selection and median-of-median algorithm	4.72%
b	Amortized analysis	3.20%
n	Union-Find	2.19%
j	Quick-sort	1.60%
l	Non-comparison based sorting (radix, bucket, counting)	1.10%
c	Recursive algorithms	1.01%
o	Suffix trees	0.76%

Table 1: Topics and their corresponding percentages.

exact matching, with some basic processing: letters were converted to lowercase, digits were mapped to letters (e.g., 1 to ‘a,’ 2 to ‘b’), and punctuation was removed. If the model failed to answer in the correct format, the question would be recognized by the automatic evaluation as false (a manual check showed that such cases were rare).

Evaluation We assessed the model's performance by running it with and without a chain-of-thought (CoT) prompt on each question, repeating each configuration five times with different randomly selected seeds. For each run, we applied our evaluation metric to determine if the model's answer was correct. We considered a question “solved” if the model answered correctly in at least 3 out of the 5 runs.

5 Results

The final performance metric represents the total number of questions the model solved on average across all questions. The results are summarized in Table 2. Overall, we observed that LLaMA 3 8B (Meta, 2024) outperformed MathΣtral 7B (Mistral, 2024) in solving the questions, both with and without prompts. The highest accuracy achieved was 32.89%, which remains well below the 50% threshold. Interestingly, our results indicate that the use of chain-of-thought prompting had a negative impact on LLaMA 3 8B's (Meta, 2024) performance in solving the questions.

Models	None	CoT
LLaMA 3 8B	32.885%	22.8187%
MathΣtral 7B	5.369%	10.738%

Table 2: Performance of models with and without Chain-of-Thought (CoT) prompting.

6 Analysis

Our results demonstrates that current models struggle to achieve high performance on Academic level theoretical computer science reasoning tasks. This type of benchmark is unique, as previous research has focused primarily on code-based tasks and math-oriented benchmarks, often overlooking data structure and algorithmic problem-solving.

Interestingly, while MathΣtral 7B (Mistral, 2024) was designed for mathematical reasoning, achieving high accuracy on math-specific tasks, it underperformed on our domain-specific questions. In contrast, LLaMA 3 8B (Meta, 2024) outperformed MathΣtral 7B (Mistral, 2024) on these questions, suggesting that its broader training data may have helped it handle the reasoning required by this our benchmark more effectively. Additionally, our results showed that the chain-of-thought (CoT) prompt significantly improved MathΣtral 7B's (Mistral, 2024) performance, doubling its accuracy compared to runs without the prompt. However, we observed a surprising contrast with LLaMA 3 8B (Meta, 2024), where the CoT prompt actually decreased its accuracy, suggesting that the effectiveness of such prompts may vary significantly depending on the model and task domain. It could even be that models LLaMA are already trained to construct their answers in methods that outperform Chain of Thought. These findings emphasize the need for further research and development of LLMs' performance in the domain of theoretical computer science reasoning. Progress in this domain can be valuable for integrating LLMs in research and education in this field.

7 Limitations and Future Work

Our study faced two main limitations. First, we had limited access to data, restricting us to data from a single university due to licensing requirements. The limited data distribution may have affected the generalizability of our findings. Second, we lacked suitable evaluation metrics to assess questions of types A and B. This limitation led us to reduce the number of questions used in our benchmarking process. Including more diverse data sources and better evaluation metrics would enable future progress in this domain.

8 Acknowledgements

We thank Adi Haviv, Michal Kleinbort, Amir Rubinstein and Hanoch Levy for the their guidance and support.

References

Austin, J., Odena, A., Nye, M., Bosma, M., Michalewski, H., Dohan, D., et al. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732.*

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. D. O., Kaplan, J., et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374.*

Chai, L., Liu, S., Yang, J., Yin, Y., Jin, K., Liu, J., Sun, T., Zhang, G., Ren, C., Guo, H., et al. 2024a. MCEval: Massively multilingual code evaluation. *arXiv preprint arXiv:2406.07436.*

Chai, L., Yang, J., Sun, T., Guo, H., Liu, J., Wang, B., Liang, X., Bai, J., Li, T., Peng, Q., et al. 2024b. XCOT: Cross-lingual instruction tuning for cross-lingual chain-of-thought reasoning. *arXiv preprint arXiv:2401.07037.*

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168.*

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., et al. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300.*

Jain, N., Han, K., Gu, A., Li, W.-D., Yan, F., Zhang, T., Wang, S., Solar-Lezama, A., Sen, K., and Stoica, I. 2024. LiveCodeBench: Holistic and contamination-free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974.*

Lai, Y., Li, C., Wang, Y., Zhang, T., Zhong, R., Zettlemoyer, L., et al. 2023. DS-1000: A natural and reliable benchmark for data science code generation. In *Proceedings of the International Conference on Machine Learning*, pp. 18319-18345. PMLR.

Levesque, H., Davis, E., and Morgenstern, L. 2012. The Winograd Schema Challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning.*

Lu, S., Guo, D., Ren, S., Huang, J., Svyatkovskiy, A., Blanco, A., et al. 2021. CodeXGLUE: A machine learning benchmark dataset for code understanding and generation. *arXiv preprint arXiv:2102.04664.*

Meta AI. 2024. LLaMA 3: Large Language Model Meta AI – Third Generation. URL: `https://ai.facebook.com/research/llama`.

Mistral AI. 2024. Mistral: A state-of-the-art open-weight language model. URL: `https://mistral.ai`.

Nijkamp, E., Pang, B., Hayashi, H., Tu, L., Wang, H., Zhou, Y., et al. 2022. CodeGen: An open large language model for code with multi-turn program synthesis. *arXiv preprint arXiv:2203.13474.*

Papineni, K., Roukos, S., Ward, T., and Zhu, W. J. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311-318.

Ren, S., Guo, D., Lu, S., Zhou, L., Liu, S., Tang, D., et al. 2020. CodeBLEU: A method for automatic evaluation of code synthesis. *arXiv preprint arXiv:2009.10297.*

Satpute, A., Gießing, N., Greiner-Petter, A., Schubotz, M., Teschke, O., Aizawa, A., and Gipp, B. 2024. Can LLMs master math? Investigating large language models on Math Stack Exchange. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2316–2320.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. 2019a. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations.* URL: `https://openreview.net/forum?id=rJ4km2R5t7`.

Yang, Z., Qi, P., Zhang, S., Bengio, Y., Cohen, W. W., Salakhutdinov, R., and Manning, C. D. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600.*

Young, J. C., and Shishido, M. 2023. Investigating OpenAI's ChatGPT potentials in generating chatbot's dialogue for English as a foreign language learning. *International Journal of Advanced Computer Science and Applications*, 14(6).