Can LLMs Master Math? Investigating Large Language Models on Math Stack Exchange

Ankit Satpute†‡§, Noah Gießing†, André Greiner-Petter§, Moritz Schubotz†, Olaf Teschke†, Akiko Aizawa§, Bela Gipp§
Ankit.Satpute@fiz-karlsruhe.de
†FIZ Karlsruhe Berlin, §University of Göttingen Germany, ‡NII Japan

ABSTRACT

Large Language Models (LLMs) have demonstrated exceptional capabilities in various natural language tasks, often achieving performances that surpass those of humans. Despite these advancements, the domain of mathematics presents a distinctive challenge, primarily due to its specialized structure and the precision it demands. In this study, we adopted a two-step approach for investigating the proficiency of LLMs in answering mathematical questions. First, we employ the most effective LLMs, as identified by their performance on math-question-answer benchmarks, to generate answers to 78 questions from the Math Stack Exchange (MSE). Second, a case analysis is conducted on the LLM that showed the highest performance, focusing on the quality and accuracy of its answers through manual evaluation. We found that GPT-4 performs best (nDCG of 0.48 and P@10 of 0.37) amongst existing LLMs fine-tuned for answering mathematics questions and outperforms the current best approach on ArgMath3 Task1, considering P@10. Our Case analysis indicates that while the GPT-4 can generate relevant responses in certain instances, it does not consistently answer all questions accurately. This paper explores the current limitations of LLMs in navigating complex mathematical problem-solving. Through case analysis, we shed light on the gaps in LLM capabilities within mathematics, thereby setting the stage for future research and advancements in AI-driven mathematical reasoning. We make our code and findings publicly available for research: https://github.com/gipplab/LLM-Investig-MathStackExchange

CCS CONCEPTS

• Computing methodologies → Discourse, dialogue and pragmatics; • Mathematics of computing → Mathematical software performance.

KEYWORDS

Math Language Models, Math Stack Exchange, Math Question Answer

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ACM Reference Format:

1 INTRODUCTION

Large Language Models (LLMs) have been very popular because of their ability to solve natural language tasks, sometimes with human-like accuracy [1, 2, 15]. Pre-training and fine-tuning the models have led to a performance-winning approach in tasks such as translation, writing code, passing professional exams, etc [19]. The LLMs are used in academia and scientific research for knowledge extraction, ideas, processing, comparison, and across multiple disciplines [18]. LLMs have also been successful in Question-Answering tasks (QA) in which they provide human-like answers to questions in natural language [20]. Evaluation of LLMs on QA has been especially useful in detecting how accurate LLMs are in generating answers and finding cases in which it could hallucinate [12, 14].

Given that mathematical content is prevalent in science, technology, engineering, and mathematics (STEM), it is imperative to evaluate the capacity of LLMs to engage with mathematical language. The challenges of LLMs with mathematics are manifold; mathematics, with its rigorous logic and abstract concepts, is communicated through a specialized language that mixes symbols and syntax in complex arrangements [24]. Unlike natural language, mathematical expressions rely on unmentioned rules and assumptions, demanding explicit knowledge and a level of precision. Mathematical reasoning still poses a sizable challenge to even the most significant language models [10, 30]. There exists an evaluation of LLMs in answering math questions in which the expected answer is straightforward numerical value [9], and an LLM-based solution [8] is the leading method on the performance scoreboard. In contrast, answers for proof or conceptual questions, which form a large part of the MSE, are not verifiable without human aid yet. However, the evaluation of models in generating answers for unorganized open questions has not been performed.

In this paper, we investigate using LLMs to answer open-ended questions (questions that cannot be answered with a yes or no response) from mathematics. We utilized questions from Math Stack Exchange (MSE), a platform dedicated to solving a wide array of mathematical questions posed by users, offering an ideal testing ground. We evaluate LLMs to find the most relevant answer. MSE encompasses questions that range from elementary level to advanced mathematics MSE questions require correctly applying
2 RELATED WORK

Research on Large Language Models (LLMs) has extended into mathematical problem solving, albeit primarily focused on pre-university mathematics. Several datasets featuring high-school-level math questions and answers exist, such as those introduced by Mao et al. [17], Urrutia et al. [29], and Deb et al. [7]. Hendrycks et al. [9] introduced the MATH dataset with 12,500 challenging competition mathematics problems. Cobbe et al. [6] developed the GSM8K dataset containing a linguistically diverse collection of high school math problems. MathQA dataset by Amini et al. [3] contains 37k English multiple-choice math word problems covering multiple math domain categories by modeling operation programs corresponding to word problems in the AQuA dataset [13]. The availability of datasets is hindered as primarily physical records are used for writing question answers instead of machine-processable and computer-aided resources [5, 25].

Specialized LLMs have been developed for particular mathematical fields. For instance, Trinh et al. [28] released AlphaGeometry, a theorem prover for Euclidean plane geometry, which solved 25 out of 30 latest Olympiad-level problems within the time limit. To mitigate the issue of LLMs generating inaccurate information, Romera-Paredes [22] developed FunSearch, which combines a pre-trained LLM with a systematic evaluator to outline verified problem-solving approaches. Moreover, LLMs equipped with integrated verification codes have demonstrated remarkable success on the MATH and GSM8K datasets, achieving accuracies near 100% in some cases [34]. The ToRA series [8], based on LLaMA-2 [27] and Codellama [23], represents the forefront of open-source models on the MATH dataset. ToRA models are based on LLaMA-2 [27] and Codellama [23] and were finetuned on a dataset of reasoning paths produced by GPT-4 [21] for the MATH and GSM-8 datasets. These models are fine-tuned on reasoning paths generated by GPT-4 for the MATH and GSM-8 datasets and self-verify their solutions by generating and executing code. However, leveraging executable code for Multiple-Selection Questions (MSE) potentially poses challenges due to the complexity of verifying reasoning steps and has not been evaluated so far. The MAMmoTH [31] family of models is trained on a dataset of chain-of-thought and program-of-thought rationales produced by GPT-4, but draws from a more diverse distribution of questions that were asked. The Llama [4] family of models have instead been finetuned on a general corpus of math-related documents and is intended to be a generalist model, leaving open the option of finetuning to specific tasks.

3 DATASET

Manual verification of answers to questions from the Mathematics Stack Exchange (MSE) is impractical due to the interdisciplinary nature of the questions and the expertise required to assess answer correctness. The ArqMATH competition dataset [16] offers a collection of MSE question-answer pairs. Task 1 from the competition’s third edition focused on retrieving relevant answers from MSE for 78 undergraduate-level mathematics questions. The relevancy of the top-ranked answers, as determined by competing systems, was assessed by student evaluators, averaging 450 evaluated answers per topic. While automated evaluation is feasible by excluding answers without evaluation scores, student assessments have shown some inaccuracies [26].

4 METHODOLOGY

For our evaluation, we select 6 LLMs. ToRA [8], LLeMa [4], GPT-4 [21], and MAmmoTH [31] give their performance of existing MathQA datasets. We also consider MABOWDOR [33], the best-performing approach for ArqMATH Task-1. Last, we consider Mistral 7B [11], general purpose LLM that has shown sub-level performance with LLaMa, GPT-4 in prompt-based tasks.

We explore two scenarios for our experiments by employing LLMs to answer MSE questions. We perform a two-step procedure in the first scenario (Answer generation). First, we gave 78 questions and generated answers using selected LLMs except MABOWDOR. MABOWDOR uses BERT-based Desne Passage Retrieval, which can only generate embeddings. Second, we indexed the answer as embedding and all the other answers from ArqMATH and found the answer most similar to the generated answer from ArqMATH. In the second scenario (Question-Answer comparison), we use selected LLMs to generate embeddings of all potential answers from ArqMATH and generate embeddings of 78 questions as well. Ultimately, we find out which answer is most similar to the question.

5 EVALUATION

This section presents the evaluation of two scenarios using metrics such as Mean Average Precision (mAP), Precision@10 (P@10), normalized Discounted Cumulative Gain (nDCG), and Binary Preference (BPref), with scores derived exclusively from evaluated ArqMATH dataset answers.

5.1 Answer generation

In this phase, answers were generated using the six selected models. These models’ outputs served as queries for retrieval within the ArqMATH answer pool. For retrieval, we used DPR vector embeddings [33] and cosine similarity. Table 1 displays the comparative performance across all models. Additionally, we executed a DPR run for comparison, akin to the method utilized by MABOWDOR. The outcome reveals that models fine-tuned on mathematical tasks underperformed relative to the DPR benchmark. Among the three 7B models (Tora, LLeMa, and MAMmoTH), variations in their performance align with their results on the MATH dataset. Notably, increasing the model size of the top performer did not yield better results. The Mistral [11] model, despite being the lowest scorer on the MATH dataset among the models considered, delivered performance comparable to that of Tora-7b. This suggests that models

1https://github.com/gipplab/LLM-Investig-MathStackExchange
Can LLMs Master Math? Investigating Large Language Models on Math Stack Exchange

### Table 1: Performance of Large Language Models in generating accurate answers.

<table>
<thead>
<tr>
<th>Model</th>
<th>nDCG'</th>
<th>mAP'</th>
<th>p@10</th>
<th>BPref</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPR (Cocomae)</td>
<td>0.464</td>
<td>0.191</td>
<td>0.324</td>
<td>0.192</td>
</tr>
<tr>
<td>Tora-7b-Code</td>
<td>0.400</td>
<td>0.159</td>
<td>0.279</td>
<td>0.170</td>
</tr>
<tr>
<td>Tora-13b-Code</td>
<td>0.355</td>
<td>0.140</td>
<td>0.266</td>
<td>0.154</td>
</tr>
<tr>
<td>LLeMa-7b</td>
<td>0.194</td>
<td>0.069</td>
<td>0.157</td>
<td>0.092</td>
</tr>
<tr>
<td>MAMMoTH-7b-Code</td>
<td>0.369</td>
<td>0.145</td>
<td>0.253</td>
<td>0.166</td>
</tr>
<tr>
<td>Mistral-7b-Instruc</td>
<td>0.396</td>
<td>0.155</td>
<td>0.267</td>
<td>0.166</td>
</tr>
<tr>
<td>GPT-4§ (right trunc.)</td>
<td>0.486</td>
<td>0.219</td>
<td>0.374</td>
<td>0.225</td>
</tr>
<tr>
<td>GPT-4§§ (left trunc.)</td>
<td>0.473</td>
<td>0.210</td>
<td>0.367</td>
<td>0.215</td>
</tr>
</tbody>
</table>

Table 2: Performance Answer retrieval given question as query embedding.

<table>
<thead>
<tr>
<th></th>
<th>nDCG'</th>
<th>mAP'</th>
<th>p@10</th>
<th>BPref</th>
</tr>
</thead>
<tbody>
<tr>
<td>MABOWDOR</td>
<td>0.132</td>
<td>0.063</td>
<td><strong>0.330</strong></td>
<td>0.088</td>
</tr>
<tr>
<td>Tora-7b-Code</td>
<td>0.095</td>
<td>0.037</td>
<td>0.149</td>
<td>0.064</td>
</tr>
<tr>
<td>Ada-002</td>
<td>0.121</td>
<td>0.049</td>
<td>0.248</td>
<td>0.069</td>
</tr>
<tr>
<td>GPT-4</td>
<td>0.153</td>
<td>0.079</td>
<td>0.321</td>
<td><strong>0.092</strong></td>
</tr>
</tbody>
</table>

5.2 Question-Answer comparison

This evaluation segment focused on matching questions with the most relevant answers using embeddings. Given the models’ original design for prompt-based answering, adjustments were necessary to facilitate embedding generation. To this end, we prefixed the prompt “This passage text means in one word: “ and used the embedding of the last token’s last hidden state. To guide the LLMs, we introduced three math-related example answers: First, “This passage: ‘Circle’”. Third, “This passage: ‘The distance between the center of an ellipse and either of its two foci.’ means in one word: ‘Eccentricity’.”

For reranking, we limited our focus to the top 10 results per query as determined by MABOWDOR [33].

Following the discussions from Zhong et al. [32], not all systems were assessed for answer reranking due to the underperformance of Tora-7b compared to the average ArqMATH approach. Given that LLeMa and MAMMoTH lagged behind Tora in the MATH and GSM benchmarks, their reranking efficacy was anticipated to be inferior. The analysis revealed Tora-7b’s Precision@10 to be inferior to all runs depicted in Table 1. This indicates that comparison question and answer embeddings might not solve the problem of retrieving relevant answers.

6 CASE STUDY

This case study examines the performance of answer generation by GPT-4 for a selected question, conducted by two annotators with expertise in Mathematics and Computer Science. Both annotators

### Figure 1: Frequency of differences in P@10 values of DPR and GPT-4 (P@10\textsuperscript{GPT-4} - P@10\textsuperscript{DPR}).

I am currently taking MIT6.006, and I came across this problem on the problem set. Despite the fact I have learned Discrete Mathematics before, I have never seen such notation before, and I would like to know what it means and how it works. Thank you:

\[
f_3(n) = \binom{n}{2}
\]

Figure 2: Question which is correctly answered by GPT-4.

consulted at zbMATH Open\footnote{https://zbmath.org/}, a comprehensive multilingual abstracting and reviewing service in pure and applied mathematics. The focus was on comparing the retrieval performance of GPT-4 and Dense Passage Retrieval (DPR), particularly on questions where GPT-4 enhanced retrieval and where DPR outperformed GPT-4. The analysis, depicted in Figure 1, indicates that GPT-4 improved precision in 38 of the 78 evaluated Mathematics Stack Exchange (MSE) questions, suggesting its efficacy in generating relevant answers for open-ended math questions on MSE compared to models fine-tuned on various MATHQA datasets. The study also explores the reasons behind ToRA’s underperformance on MSE questions despite its training on the MATH dataset.

6.1 GPT-4

We delve into two specific cases: one where GPT-4’s answers improved retrieval performance in Precision@10 (P@10) and another where it had the opposite effect. These cases are outliers, with most topics showing P@10 variations between −0.1 and 0.2. We do not mention answers due to space limitations, but all the recorded answers for GPT-4\footnote{https://github.com/gipplab/LLM-Investig-MathStackExchange/blob/main/topics-and-qrels/topics.arqmath-2022-gpt4-generated-answers.csv} and all the other LLMs are available in the repository.

6.1.1 A.384: Retrieval Boost. The question shown in Figure 2, GPT-4’s answer strongly shows increases P@10 (from 0.0 of DPR to 0.6 of GPT-4). The reason is that the first retrieved results by the DPR do not feature a binomial coefficient at all, but the first results of the GPT-4 retrieved result include at least an expansion of \(\binom{n}{3}\). We
Question. Suppose that all the tangent lines of a regular plane curve pass through some fixed point. Prove that the curve is part of a straight line. Prove the same result if all the normal lines are parallel. I am working on differential geometry from the book by Pressley and I have a doubt in the solution of the above question whose (brief) solution is given by: Solution: We can assume that the curve γ is unit-speed and that the tangent lines all pass through the origin (by applying a translation to γ). Then, there is a scalar $\lambda(t)$ such that $y'(t) = \lambda(t)y(t)$ for all t. Then, $y'' = \lambda'y + \lambda y' = (\lambda' + \lambda^2)y$. Can anyone please explain me how does this line follow? “Then, there is a scalar $\lambda(t)$ such that $y'(t) = \lambda(t)y(t)$ for all t.” Thanks in advance.

Reason for worsened retrieval. Imagine a car driving along the curve, and at the point of interest, the car hits an oil slick and shoots off in the straight line of its travel as it hits the point. The line of the car gives the tangent line at the point (provided the curve is differentiable at the point). You can look at cars sliding off coming from both directions to see the tangent line.

Figure 3: Question which is incorrectly answered by GPT-4. The reason for worsened retrieval shows generated answers irrelevant to the question.

observe that, without context, DPR cannot infer any meaning from the formula. Hence, GPT-4 shows a good contextual understanding of the ground truth formula.

6.1.2 A.337: Retrieval Worsened Figure 3. P@10 decreases from 0.5 for the DPR run to 0.1 for GPT-4 for the following question (only the title is shown due to length): The answer given by GPT-4 misses the point since it does not explain how the specific line the poster asks about follows from the assumptions. We observe a pattern representing how GPT-4-generated answers can guide the retrieval system in the wrong direction. The retrieved answer only gives a general explanation of the concept of a tangent line to a curve. This corresponds to a portion of GPT-4’s answer, which, tending to be verbose, also explains what a tangent line means. This shows that GPT-4 cannot answer the questions about complex interactions between mathematical concepts.

6.2 Tora-7b-Code

The answers obtained by all the smaller models (basically every model except GPT-4) are of quite low quality. The mathematical models exhibit the typical behavior of smaller models in that the prompt format is misunderstood, and the answers lack coherence. Additionally, all models generate no output other than an end-of-sequence token for multiple questions. (6 in the case of Tora-7b-Code). In contrast, Mistral is much better at keeping a conversational tone than Tora. Additionally, the reasoning appears to be more structured. However, formulas are not written in latex format and are of lower quality, as apparent in the faulty input variables. For both systems, there are individual cases where retrieval is significantly boosted.

6.2.1 A.309 - Tora-7b-Code boosts retrieval. Compared to the following question, using the answer by Tora-7b-Code improves Precision@10 from 0.5 to 0.8. Instead of an actual answer, ToRA simulates

Question: I have a question regarding the number of solutions of a equation over a finite field $\mathbb{F}_p$. First of all, consider the equation $x^3 = a$ over $\mathbb{F}_p$, where $p$ is a prime such that $p \equiv 2 \pmod 3$. The book that I’m currently reading says that this equation has exactly one solution in $\mathbb{F}_p$ for every $a \in \mathbb{F}_p$, because gcd$(3, p − 1) = 1$, but the book does not prove this. Unfortunately, this doesn’t convince me enough. Is there a convincing elementary straightforward proof justifying why is this true?

Ansew by ToRA Proof: Suppose to the contrary that there are more than one solution in $\mathbb{F}_p$. By Fermat’s little theorem, $a^{p-1} = 1$, then $a$ is congruent to 1 modulo $p - 1$. Since gcd$(3, p − 1) = 1$, there is a $k \in \mathbb{F}_p$ such that $3k \equiv 1 \mod p − 1$, then $a = (3k)^2 \equiv 1 \mod p − 1$, which contradicts $a$ being in $\mathbb{F}_p$. Thus, the equation $x^3 = a$ has exactly one solution in $\mathbb{F}_p$. I don’t really get the last part: “Since gcd$(3, p − 1) = 1$, there is a $k \in \mathbb{F}_p$ such that $3k \equiv 1 \mod p − 1$. Could you please elaborate on this part? Comment: I suggest you look into [this] ([...]) and [this]([...]). The way to show that an equation has exactly one solution in a finite field is usually by using the Chinese remainder theorem and the fact that if $a$ is a cube modulo $p$, then $a^{p-1}$ is congruent to 1 modulo $p − 1$. Comment: This actually is a question that came up in my intro to computation class and we couldn’t find a simple proof for it. I am looking for a basic level proof. Comment: If $p \equiv 2 \pmod 3$, then gcd$(p − 1, 3) = 1$. By the Euclidean Algorithm, there is some $k \in \mathbb{F}_p$ such that $3k \equiv 1 \mod p − 1$. Therefore, $a \equiv k^2 \pmod p − 1$. This contradicts [...]