Abstract

The diverse nature of mathematics makes it uniquely relevant to a wide variety of professional fields and facets of daily life. Math is applied in manufacturing cars and designing phones, and it is used in cooking recipes and calculating property taxes. The immense relevance of mathematics combined with the ever-increasing reliance on digital platforms for problem-solving has created the demand for math Q&A platforms, such as Math Stack Exchange (MSE), which are popular due to the ease at which they can be accessed from online**.** Unfortunately, one of the major drawbacks of these platforms is the fact that users are sometimes forced to wait for an unspecified amount of time before receiving an answer to their question. With the recent widespread adoption of Large Language Models (LLM), people have mitigated the issue of needing to wait for a human to provide an answer by instead prompting an LLM for answers, but the accuracy of these generated answers leaves much room for improvement. In this work, we present a similarity search tool that matches existing math questions and answers based off of the similarity of the text and equation portions of the content. This tool is used to improve the accuracy of the answers that LLMs give when prompted with math questions found on MSE. This research presents new ideas for the math Q&A community to consider, concluding that there is potential for the similarity search tool to improve the math question answering capabilities of LLMs.

Introduction

Mathematics plays a critical role in a wide variety of disciplines, including engineering, economics, computer science, biology, etc. It is the foundation for modern advancements in medicine [1], transportation [2], entertainment [8], and many other fields. Outside of a professional setting, math is also an integral part of everyday life, appearing in ordinary everyday settings such as finance management, shopping, cooking, scheduling, etc. The relevance of mathematics extends far into daily life, yet many perceive mathematics as an inherently difficult topic [3], leading to reluctance to engage with it [4]. This is especially problematic as the demand for technical literacy increases [5]. But this affects individuals even outside of a career, as a lack of math skills increases the likelihood of making illogical and poor decisions [6].

The reluctance to engage with math does not reflect the importance of math, and highlights the need to make math learning more accessible. If learning math can be perceived as a less daunting task, then unwillingness to engage with it will diminish. Learning math can be made less intimidating by making math learning more accessible. In a world heavily reliant on the internet, one of the simplest and most accessible ways to learn is through using technology, and therefore online math Q&A systems have become an answer to the problem over the past couple decades as a way to conveniently get answers to math questions [7]. Math Q&A systems allow users to ask questions as a user and answer existing questions as an answer provider. Some Math Q&A systems have a very large user base of people that are knowledgeable about math who are willing to play the role of answer provider as a hobby, all the way to people who professionally practice math at a high level. These systems provide a convenient way of

connecting those who have math questions with those who are able to answer math questions, for a wide range of math topics and levels of difficulty.

The convenience of these math Q&A systems promotes math learning, but one big problem is that sometimes a user has to wait an unspecified amount of time to receive an answer, if it ever comes, perhaps defeating the convenience gained by being accessible from digital platforms. Often, a user's question is time-sensitive because they have a deadline to meet at work, or a due date to meet at school. And quite often a user simply does not want to wait for an unknown amount of time, as they might lose interest or become frustrated with the inability to find an answer in a timely manner. Combined with the recent widespread adoption of LLMs, people with math questions have started to gravitate towards asking LLMs their math questions instead, as LLMs provide instantaneous answers. The drawback is the fact that, while LLMs generally perform well, especially for strictly natural language tasks, they will sometimes generate inaccurate or irrelevant information, especially for tasks that are not LLM strong points, such as mathematics. LLMs generating less than desirable answers can discourage their users, leaving them dissatisfied and potentially harming their desire to learn math. Improving the accuracy of the answers that an LLM generates is thus a critical step towards improving math education.

To address this challenge, we explore the application of a similarity search tool for fine-tuning, with the aim of improving the accuracy of math answers generated by an LLM. The similarity search tool uses the all-MiniLM-L6-v2 sentence transformer combined with Facebook AI Similarity Search (FAISS) to recommend relevant MSE answers to MSE questions. The question and answer pairs are then used to fine-tune an LLM, which is then prompted to answer different MSE questions. The answers from the fine-tuned and not fine-tuned LLMs are then compared to assess the novelty of this method of fine-tuning.

There are a small handful of other instances of research that tackle the problem of improving mathematical capabilities of LLMs through fine-tuning or similarity techniques, but no previous research has taken the approach described above, and this research aims to assess its novelty and viability. This approach is also notable for its effectiveness on more lightweight LLMs (<10 billion parameters), as modern research tends to focus on LLMs with tens of if not hundreds of billions of parameters.

Related Works

The recognition of the performance gap between natural language tasks and technical tasks (such as math) has driven research into how to improve the technical and mathematical capabilities of LLMs. Incorporating training techniques such as fine-tuning, chain of thought reasoning, etc., researchers have improved the mathematical reasoning capabilities of LLMs. This section provides a brief overview of a few notable systems addressing these challenges.

- Minerva [9]: Minerva has Google's general-purpose PathWays Language Model (PaLM) as the base model, and then it further trains the model on a large swath of scientific and mathematical content. Using techniques such as few-shot prompting, chain of thought,

and scratchpad prompting, to name a few, Minerva manages to be proficient at natural language tasks as well as technical tasks, including answering math questions.

- Rejection Sampling Fine-Tuning (RFT) [10]: RFT improves the mathematical capabilities of LLMs by training LLMs on automatically augmented math data. RFT generates and collects reasoning paths as data, and uses this data to fine-tune LLMs and increase performance for mathematical tasks.
- Reasoning with Reinforced Fine-Tuning (ReFT) [11]: ReFT first uses reinforced fine-tuning to warm up an LLM on chain of thought data, and then it uses proximal policy automation reinforcement learning to further train LLMs. The performance of the LLMs are increased by rewarding correct intermediate steps rather than only final answers.
- Retrieval-augmented Generation (RAG) [12]: RAG focuses on improving LLMs math reasoning capabilities in a primary educational setting. By training LLMs on textbook content, RAG manages to reduce hallucinations and improve the relevancy of generated answers to math questions.

Our approach to enhancing the math capabilities of LLMs uses fine-tuning with data retrieved by a similarity search tool. This approach shares some similarities with some of the above models, such as the consideration of math equations in tandem with semantic information. Having said that, our research introduces the novel combination of a similarity search tool built with sentence transformers and FAISS to improve the math capabilities of light-weight LLMs.

Methodology

Our approach to improving the ability of LLMs to generate accurate answers to math questions involves fine-tuning models using relevant Math Stack Exchange (MSE) content identified through a specialized similarity tool. This tool leverages both textual and mathematical equation comparison techniques to retrieve the most appropriate content for training.

Similarity Search Tool

The similarity search tool uses the all-MiniLM-L6-v2 sentences transformer with Facebook AI Similarity Search (FAISS) to quickly and efficiently compare and retrieve similar content from a large corpus of MSE postse.[1].

all-MiniLM-L6-v2

The all-MiniLM-L6-v2 sentence transformer is used to generate vector representations of text. This transformer model is pre-trained on a large corpus of text so that it learns to understand context, syntax, and other semantics. It then uses this knowledge to convert new text into dense vector embeddings, where each piece of text is represented as a vector in a high-dimensional space, allowing for retention of the semantic meaning of the text. This process ensures that

[1] The complete code for the similarity search tool is available at https://github.com/ElemehnoP/CS497R

semantically similar content lies in close proximity, which is measured by cosine similarity. The close proximity in a high-dimension space is represented by the direction of the vector. Two similar texts will have vectors that have a similar direction, so the similarity of two texts can be represented by the angle between the two vectors, where a value closer to 1 is similar and a value closer to 0 is dissimilar (see Table 1.0 for an example of the text comparison process).

While this model is primarily designed for natural language text, it can also be effectively applied to mathematical equations. All-MiniLM-L6-v2's textual comparison ability is surprisingly effective for comparing math equations represented in MathML. Much like semantic text, MathML is structured so that items' meanings are based on the context they are used in. By treating MathML equations as plain text strings, rather than as XML, all-MiniLM-L6-v2 is easily able to process the information like it would semantic text, with minimal meaning loss.

The similarity search tool utilizes all-MiniLM-L6-v2's ability to generate accurate vector representations of text to process both semantic text and MathML equations, allowing for the most accurate similarity comparison when comparing questions to answers. Additionally, all-MiniLM-L6-v2's relatively small size and efficient inference capabilities make it suitable for processing large volumes of MSE content without excessive computational demands.

FAISS

Once the text and equations are transformed into vector embeddings using the all-MiniLM-L6-v2 model, the results are fed into FAISS to generate an index of math questions and answers. FAISS is a library that enables efficient similarity search by looking up information stored as vectors in an index. FAISS is particularly well-suited for our application due to its ability to handle large datasets and perform nearest neighbor searches with minimal computational overhead while retaining high accuracy. This is accomplished by using techniques such as product quantization and optimized clustering, which reduce dimensionality while preserving approximate similarity relationships.

In order to most effectively take advantage of the speed offered by the FAISS index that is populated with the output of MSE information fed through the all-MiniLM-L6-v2 sentence transformer, the tool will first generate an index of MSE questions, rather than MSE answers. This is because each math question has anywhere from 5-30 corresponding answers, and despite FAISS's efficient similarity search, indexing and searching through all answers would be computationally expensive and reduce performance. So instead, the MSE question that is selected as the query is instead compared to all the MSE questions in the FAISS index, and the top 5 most similar questions are retrieved. Then, a second FAISS index is created and populated with the answers of the 5 most similar questions that were retrieved. This keeps the FAISS indexes large enough to make use of FAISS's efficiency, but small enough to remain within manageable computational bounds. The query is then compared with the compiled math answers, and the top 5 most similar math answers are returned. Similarity is determined by computing the distances between both semantic text embeddings and equation embeddings.

Fine-tuning LLMs

Fine-tuning refers to the process of taking a pre-trained language model and adapting it to perform better on a specific task or domain by further training it on a smaller, task-specific dataset. This is different from training a model from scratch, as the model already possesses a general understanding of language, fine-tuning simply adjusts its parameters to improve performance on a specialized task. This approach not only boosts task-specific performance but also avoids the enormous computational cost of full model training, making it far more practical and accessible. In our case, the goal of fine-tuning is to enhance performance on mathematical question answering, particularly within the domain of MSE.

In our research, we selected three language models for experimentation: LLaMA3.1-8B, Qwen2.5-7B, and Gemma3-4B. These models were chosen primarily for their reasonable size, which allows for experimentation without requiring extensive computational resources. While larger models may offer superior performance in some scenarios, the selected models provide a good balance between capability, efficiency, and hardware requirements.

The novelty of using our similarity search tool for fine-tuning is proven by comparing the baseline performance of the LLMs to the fine-tuned but otherwise identical LLMs.

Each LLM is first queried with a subset of MSE questions. This establishes a baseline performance on mathematical queries without any fine-tuning. Then, the prompt is modified to include both questions and answers retrieved by our similarity tool from a different subset of MSE posts. Although the initial MSE question subset and the tool-based fine-tuning subset are distinct, augmenting the prompt with retrieved, contextually relevant content results in measurable improvements in answer quality and accuracy. The fine-tuning demonstrates that even when different data subsets are used, targeted and contextually relevant information enhances LLM performance. When the same model is evaluated under the same conditions, with the only difference being the fine-tuning process, the resulting performance gains confirm the effectiveness of our retrieval-based fine-tuning strategy.

Experimental Results

In this section, we address the following Research Questions (RQ):

- RQ1: What improvements are seen when comparing the fine-tuned LLMs to the not fine-tuned LLMs?
- RQ2: What are the limitations of using the similarity search tool to fine-tune LLMs to improve relevant answer generation for math questions?
- RQ3: What do the observed patterns of performance gains and regression reveal about the robustness of this approach and future optimization using the similarity search tool?

Process

We conducted our evaluation by initially prompting the LLMs with a set of 10 MSE questions (see Figure 1 for the prompt's content). The LLMs are instructed to generate 10

answers that are diverse and accurate, providing the most accurate answers first and least accurate last. Once the results are recorded, the LLMs are prompted with the same questions, but the prompt is modified to include the separate MSE questions and the similarity search tool's answers to those questions. The new prompt is designed to be very similar to the original prompt except it also provides contextually relevant information with the goal of enhancing the models' performance.

The experiments were run independently on three models: LLaMA3.1-8B, Gemma3-4B, and Qwen2.5-7B. For each model, we compared the performance before and after fine-tuning using the metrics described below.

Metrics

The quality of MSE answers are determined by MSE user ratings. The higher the rating, the better the answer is. The models' performances are measured by their ability to generate answers that have significant overlap with the top 5 rated MSE answers for a particular problem. Significant overlap is determined by manual inspection of the MSE answers and generated answers. If there is significant overlap between the generated answer and the content of any of the top 5 MSE answers, then the generated answer is considered relevant, else it is considered irrelevant. This binary classification (relevant = 1, irrelevant = 0) is used to calculate the following metrics that quantify the performance of the LLMs:

- Precision @3 (P@3): Measures the proportion of relevant answers among the top 3 responses. This metric provides insight into the model's ability to rank highly relevant answers in the very first few outputs.
- Precision @5 (P@5): Similar to P@3, but calculated over the top 5 responses. This gives a broader view of early retrieval quality.
- Average Precision (AP): Averages precision scores at the ranks where relevant answers occur, summarizing the precision across all positions.
- Reciprocal Rank (RR): The inverse of the rank at which the first relevant answer is found. A higher RR indicates that relevant answers appear sooner.
- RR @5: Restricts the RR calculation to the first 5 results, ensuring that only early retrieval is considered.
- Normalized Discounted Cumulative Gain (nDCG): Evaluates the ranking quality by giving higher scores when relevant answers appear earlier in the result list.
- nDCG @5: Computes nDCG for only the top 5 answers, emphasizing performance in the most critical positions.

Table 2: LLM performance metrics (rounded). Green indicates improvement, red indicates regression.

Model	Fine-tuned	P@3	P@5	AP	RR	RR@5	nDCG	nDCG @5
LLaMA3.1-8B	No	0.633	0.560	0.6106	0.683	0.683	0.695	0.707
	Yes	0.633	0.580↑	0.689↑	0.750↑	0.750↑	0.774↑	0.785↑
Qwen2.5-7B	No	0.833	0.780	0.868	1.000	1.000	0.950	0.975
	Yes	0.967↑	0.900↑	0.943↑	0.950↓	0.950↓	0.972↑	0.976↑
Gemma3-4B	No	0.600	0.640	0.681	0.817	0.800	0.777	0.757
	Yes	0.633↑	0.500↓	0.612↓	0.650↓	0.650↓	0.656↓	0.668↓

Results:

- RQ1: *What improvements are seen when comparing the fine-tuned LLMs to the not fine-tuned LLMs?*
  The results (see Table 2) indicate that the retrieval-based fine-tuning process generally enhances the mathematical question answering capabilities of the LLMs, though the degree of improvement varies. Qwen2.5-7B demonstrates the most significant performance boost. While fine-tuning has slightly worse RR and RR@5 values, all other metrics show improvement, most notably P@3, which increases from 0.833 to 0.967. LLaMA3.1-8B shows more moderate improvements after fine-tuning. Every metric improves after fine-tuning except for P@3, which remains unchanged. While the improvements vary from category to category, overall, the improvements seen in Qwen2.5-7B and LLaMA3.1-8B suggest that the similarity search tool is effective for fine-tuning LLMs using the similarity search tool in order to achieve better math answer relevance.
- RQ2: *What are the limitations of using the similarity search tool to fine-tune LLMs to improve relevant answer generation for math questions?*
  In sharp contrast to Qwen2.5-7B and LLaMA3.1-8B, Gemma3-4B's results show that it performs better without fine-tuning than with. P@5 (0.640 → 0.500), AP (0.681 → 0.612), RR (0.817 → 0.650), and nDCG (0.777 → 0.656) all decrease after fine-tuning, with only P@3 (0.600 → 0.633) showing a modest improvement. A possible explanation for these results is perhaps the similarity search tool is effective at improving larger LLMs (7B+) but not LLMs as small as Gemma3-4B (4B).
- RQ3: *What do the observed patterns of performance gains and regression reveal about the robustness of this approach and future optimization using the similarity search tool?*

These observations suggest that, while the retrieval-based fine-tuning strategy is effective for certain models, further investigation is needed to understand its limitations and to optimize it for models like Gemma3-4B. Factors such as model size, internal architecture, training strategies, etc. are all worth considering.

Comparison to Unification-Based Algorithm

In this section, we review the unification-based recommendation algorithm introduced in our previous research [15], compare it to the similarity search tool and fine-tuning framework by assessing the novelty of this tool, and explain why we instead adopted the similarity search tool presented in this paper. We focus on differences in design, scoring, computational efficiency, and integration with LLMs.

Summary of Unification-Based Algorithm

Our previous research proposes a matching algorithm for math Q&A systems that operates exclusively on the MathML representations of equations. Its primary goal is to improve answer recommendation quality on platforms like Math Stack Exchange by computing a similarity score between the query equation and the answer equation. The method comprises four stages:

1. Tree Traversal and Branch Extraction
    - Each MathML equation is parsed into an XML tree.
    - A depth first traversal produces two "arrays of arrays", where each subarray corresponds to the path from the <math> root node to a leaf node.
    - Numeric indices are injected into the subarrays to distinguish numerator and denominator, base and exponent, etc., so the uniqueness of two branches is reflected in the calculated similarity score of those branches.
2. Depth Scoring with Weights
    - The query equation is referred to as the "Right Tree", and the equation in the candidate answer is referred to as the "Left Tree". The Left Tree and Right Tree are compared by computing the length of the longest contiguous subsequence of MathML tags.
    - Certain tags are weighted to penalize structural mismatches that drastically alter an equations structure. The weighted tags <mover>, <munder>, and <munderover> are used for annotations like accents, limits, or combined notation respectively, and have weights of 0.005. The weighted tags <mfrac>, <msup>, <msubsup>, and <msqrt> are used for annotations like fractions, superscript, superscript and subscript, and square roots, and have weights of 0.5.
    - The per-branch depth score (between 0 and 1) is multiplied by the product of weights for any unmatched weighted tags, yielding a final depth score for each branch.
3. Complexity Adjustment

- A complexity score is computed as *max(|L|,|R|) / min(|L|,|R|)*, where |L| and |R| are the total tag-counts in the Left and Right trees.
- This penalizes cases where a small query superficially matches a tiny portion of a much larger answer, preventing inflated similarity from repeated branch matches.

4. Final Similarity Score
    - The final similarity score is computed as *complexity_score * X + X * (1 - X),* where *X* is the averaged weighted depth score across all branches.
    - This formula is designed to give complexity a greater impact on the total score when branch depth indicates a strong match.

Finally, the unification-based algorithm's effectiveness is tested using 10 MSE questions. The unification-based algorithm runs in 0.114 seconds per query vs. GPT-4's 4.296 seconds, while matching GPT-4's 0.8 accuracy in identifying the highest rated answer equation.

Novelty and Limitations of the Unification-Based Algorithm

The innovation of the unification-based algorithm lies in the novel usage of unification, which is traditionally used to demonstrate equivalence of math information such as sets and expressions. Adjusting unification for the task of determining equation similarity allowed for effective and simple comparison of math equations. The scoring mechanism that determines the similarity is thorough, using weights and multiple scores to arrive at the most accurate final score. All the while, the algorithm is relatively simple and fast, distinguishing it from modern math question and answer recommendation tools, such as GPT-4.

However, despite the accurate equation comparison, the unification-based algorithm was severely limited by its equation-only scope, as well as the limited testing it underwent. The algorithm ignores surrounding natural language context, which very often provides crucial context for the question and answer. The unification-based algorithm was also not applied to highly relevant math Q&A tools such as LLMs, nor did it undergo as thorough testing as the similarity search tool underwent.

Why the Similarity Search Tool was Chosen

The novelty of the similarity search tool, as detailed in this paper, lies in the usages of FAISS and the all-MiniLM-L6-v2 sentence transformer. While these tools are traditionally used for natural language processing tasks, they are also suitable for capturing important relationships between math symbols represented as text. While the equation matching is less thorough than the unification-based approach, the usage of FAISS and sentence transformers allows for the comparison between equations and also natural language, which is present in nearly every math question and answer. Therefore, the math question and answer comparison is very accurate, but slower due to the use of powerful tools such as FAISS.

The novelty of the similarity search tool was proven with extensive testing, as well as incorporation with LLMs, a technology relevant in every professional field, especially mathematics. Due to the more thoroughly proven novelty, as well as the more relevant applicability of the similarity search tool, we decided to pursue research into the similarity search tool.

Conclusion

Math is a very diverse and important field of study. It is relevant to daily life directly and indirectly, as it is used in manufacturing cars and developing new medicine, as well as shopping and finance management. Better math skills lead to better decision making capabilities [14], but math learning is inhibited by the fact that there is a general reluctance to engage with it. Making math learning more accessible is crucial to solving this problem, and math Q&A platforms address this need. However, the inconvenience of not receiving a timely answer to a question can be a roadblock to math learning. People with math questions attempt to mitigate this issue by turning to LLMs for math answers, but the effectiveness of LLMs for answering math questions is heavily dependent upon the quality of the generated answers. Inaccurate answers can discourage those with questions, and so we propose a similarity search tool that uses sentences transformers, FAISS, and fine-tuning to attempt to improve the quality of generated answers. The proposed system is notable for its unique design, as well as its effectiveness on LLMs with less than 10 billion parameters. Conducted empirical analysis has proven the novelty and the effectiveness of the proposed system.

Figure 1: The LLM prompt (question not included).
Note: The bracketed text is only included when fine-tuning.
It is omitted when not fine-tuning.

You will receive a math question, which may contain MathML equations. Provide 10 independent, complete, and concise answers. Each answer should fully address the question on its own.
Accuracy is the top priority. However, vary wording or approach where possible. If few correct answers exist, minor variations are acceptable, but avoid exact duplicates. List the best answers first. Number them 1-10 and format equations in standard, human-readable notation (not MathML).
Do not include any extra text—only the 10 numbered answers.
[You have been given a block of related Q&A pairs in the system message. Use this as context to improve your answers. Some answers in the block may be more useful than others.]
With that said, here's the question:

Table 1: Example of sentence comparison using the
all-MiniLM-L6-v2 sentence transformer.

#	Sentence content.	Vector embeddings.	Cosine similarity with sentence 1.
1	The quick brown fox jumps over the lazy dog.	[0.04393356 0.05893438 0.04817837 0.07754807 0.02674437 -0.03762959, -0.0026051 -0.05994307 -0.00249604 0.02207284, … (374 more elements)]	1.0
2	The fast dark-colored fox leaps over the sleepy canine.	[4.8916392e-02 5.3246785e-03 5.7936970e-02 1.3848297e-01, 3.3720445e-02 -1.4107534e-02 -1.8166670e-03 -4.5666382e-02, 9.3039256e-03 -7.8561796e-05, … (374 more elements)]	0.8091
3	The health of the economy depends on many factors.	[0.07817048 -0.00386362 -0.00703631 0.00449317, 0.03409512 0.02474042, -0.00538037 0.0010287 -0.0720941 0.01274734, … (374 more elements)]	0.0053

[1] Y. K. Wan, C. Hendra, P. N. Pratanwanich, and J. Göke, “Beyond sequencing: machine learning algorithms extract biology hidden in Nanopore signal data,” *Trends in Genetics*, Oct. 2021, doi: https://doi.org/10.1016/j.tig.2021.09.001.

[2] Y. Safadi and J. Haddad, “Optimal combined traffic routing and signal control in simple road networks: an analytical solution,” *Transportmetrica A: Transport Science*, vol. 17, no. 3, pp. 308–339, Jul. 2020, doi: https://doi.org/10.1080/23249935.2020.1783023.

[3] A. Fritz, V. G. Haase, and P. Räsänen, Eds., *International Handbook of Mathematical Learning Difficulties*. Cham: Springer International Publishing, 2019. doi: https://doi.org/10.1007/978-3-319-97148-3.

[4] A. Rattan, C. Good, and C. S. Dweck, “‘It’s Ok — Not Everyone Can Be Good at math’: Instructors with an Entity Theory Comfort (and demotivate) Students,” *Journal of Experimental Social Psychology*, vol. 48, no. 3, pp. 731–737, May 2012, doi: https://doi.org/10.1016/j.jesp.2011.12.012.

[5] “CHARTING A COURSE FOR SUCCESS: AMERICA’S STRATEGY FOR STEM EDUCATION,” 2018. Available: https://www.energy.gov/sites/prod/files/2019/05/f62/STEM-Education-Strategic-Plan-2018.pdf

[6] R. Jiang *et al.*, “How mathematics anxiety affects students’ inflexible perseverance in mathematics problem-solving: Examining the mediating role of cognitive reflection,” *British Journal of Educational Psychology*, vol. 91, no. 1, Jun. 2020, doi: https://doi.org/10.1111/bjep.12364.

[7] T. T. Nguyen, K. Chang, and S. C. Hui, “A math-aware search engine for math question answering system,” pp. 724–733, Oct. 2012, doi: https://doi.org/10.1145/2396761.2396854.

[8] P. Hanrahan, “The Math Behind the Movies,” *Frontiers for Young Minds*, vol. 11, Jan. 2024, doi: https://doi.org/10.3389/frym.2023.1166415.

[9] A. Lewkowycz *et al.*, “Solving Quantitative Reasoning Problems with Language Models,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 3843–3857, Dec. 2022, Available: https://proceedings.neurips.cc/paper_files/paper/2022/hash/18abbeef8cfe9203fdf9053c9c4fe191-Abstract-Conference.html

[10] Z. Yuan *et al.*, “Scaling Relationship on Learning Mathematical Reasoning with Large Language Models,” *arXiv.org*, 2023. https://arxiv.org/abs/2308.01825

[11] T. Q. Luong, X. Zhang, Z. Jie, P. Sun, X. Jin, and H. Li, “ReFT: Reasoning with Reinforced Fine-Tuning,” *arXiv.org*, 2024. https://arxiv.org/abs/2401.08967

[12] Z. Levonian *et al.*, “Retrieval-augmented Generation to Improve Math Question-Answering: Trade-offs Between Groundedness and Human Preference,” *arXiv.org*, Oct. 04, 2023. https://arxiv.org/abs/2310.03184

[13] R. Stahnke, S. Schueler, and B. Roesken-Winter, “Teachers’ Perception, Interpretation, and Decision-Making: A Systematic Review of Empirical Mathematics Education Research.,” *ZDM: The International Journal on Mathematics Education*, vol. 48, pp. 1–27, 2016, Accessed: Dec. 23, 2024. [Online]. Available: https://eric.ed.gov/?id=EJ1096508

[14] Unpublished manuscript: “Using Unification-Based Techniques for More Accurate Equation Matching in Math Q&A Systems,” 2024.