*IOSR Journal of Computer Engineering (IOSR-JCE)*
*e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 27, Issue 3, Ser. 2 (May. – June. 2025), PP 70-74*
*www.iosrjournals.org*

# Evaluating Large Language Models on Mathematical Problem-Solving Tasks

Anshika Tiwari[1] , Gopindra Kumar[2]
*Scholar M.Tech(C.S.E) ABSSIT[1] , HOD Dept. of CSE, ABSSIT[2]*

***Abstract***
*The rapid advancement of natural language processing (NLP) systems and the growth of large language models (LLMs) have opened up new possibilities in education and teaching methodologies. These innovations enable personalized learning experiences and instant feedback while remaining accessible and cost-effective. A significant application of this progress is in mathematical problem-solving, which requires both an understanding of complex problem statements and accurate arithmetic computations throughout the process. However, the evaluation of LLMs' arithmetic abilities has not been extensively explored.*
*To address this gap, we introduce MathQuest, a comprehensive mathematics dataset derived from the 11th and 12th-grade NCERT Mathematics textbooks. This dataset includes mathematical problems of varying difficulty levels, covering a wide range of mathematical concepts. Using MathQuest, we fine-tune and assess the performance of three well-known LLMs: LLaMA2, WizardMath, and MAmmoTH. Our experiments indicate that MAmmoTH-13B outperforms the other models, exhibiting the highest proficiency in solving mathematical problems. As a result, MAmmoTH-13B establishes itself as a strong and reliable benchmark for tackling NCERT mathematics problems.*

---

Date of Submission: 01-05-2025 Date of Acceptance: 10-05-2025

---

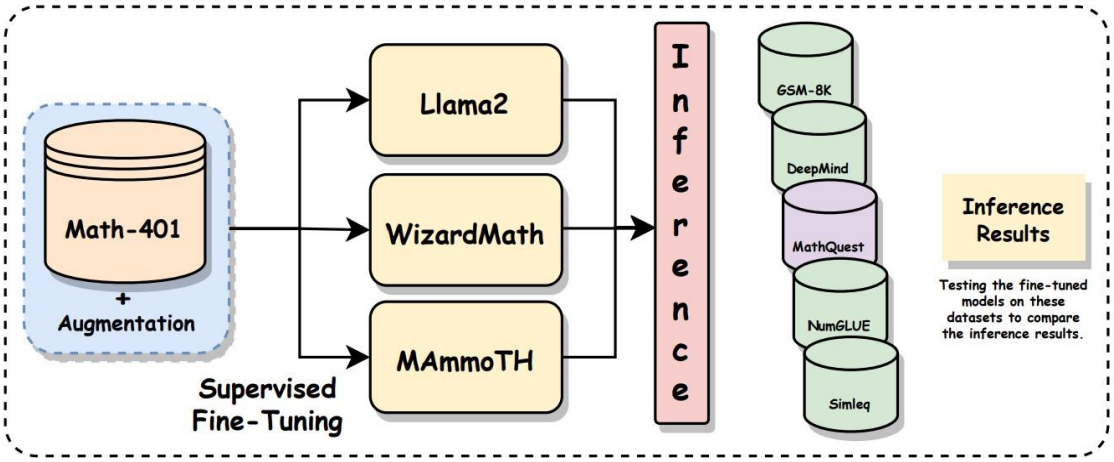


Figure 4.5: Fine-Tuning and Inference flow of LLMs

## II. Methodology

The process of solving mathematical problems involves a diverse range of cognitive skills. It includes understanding problem statements, identifying relevant concepts and formulas, applying

appropriate algorithms and strategies, performing precise calculations, and verifying the validity and reasonableness of solutions. Traditionally, the teaching and assessment of mathematical problem- solving have relied on conventional methods such as textbooks, worksheets, and examinations. However, these methods often provide limited feedback and guidance to learners. With advances in artificial intelligence and natural language processing, LLMs have grown as strong tools for producing natural language text across a broad spectrum of areas and applications. Existing LLMs face significant challenges in solving math word problems that require multi-step arithmetic calculations, complex reasoning, or domain-specific knowledge.

**Dataset**

In our research experiments, we utilized the Math-401 dataset [77], comprising 401 samples of mathematical problems. This dataset encompasses a diverse range of mathematical operations, including (+, , , /), exponentiation, trigonometric, logarithmic functions (sin, cos, tan, log, ln), and incorporates integers,

decimals, and irrational numbers ($\pi$, $e$). Acknowledging the restricted sample size of Math-401 for effective learning by large language models, we expanded it through augmentation, yielding a dataset size of 302,000 samples. To create our augmented dataset, we utilized the **SymPy**[2] Python library. This library enabled us to generate arithmetic mathematical equations along with their corresponding ground truth values. Table 4.7 offers a detailed breakdown of the question types employed in crafting our augmented dataset.

| Type | Range | Decimal Places (1 - 4) | Variables | Count |
|---|---|---|---|---|
| Small Integer | [-20, 20] | × | (x, y) | 65,000 |
| Small Decimal | [-20, 20] | ✓ | (x, y) | 35,000 |
| Small Decimal + Integer | [-20, 20] | ✓ | (x, y) | 39,000 |
| Large Integer | [-1000, 1000] | × | (x, y) | 39,000 |
| Large Decimal | [-1000, 1000] | ✓ | (x, y) | 25,000 |
| Large Decimal + Integer | [-1000, 1000] | ✓ | (x, y) | 25,000 |
| 3 Terms | [-100, 100] | ✓ | (x, y, z) | 25,000 |
| 4 Terms | [-100, 100] | ✓ | (w, x, y, z) | 49,000 |
| **Total** | - | - | - | 302,000 |

Table 4.7: The distribution of types of question in our augmented Math-401 dataset

**MathQuest:** In this research work, we have also carefully curated our proprietary dataset, known as MathQuest, by extracting problems from high school mathematics NCERT books. MathQuest serves as a diverse resource, incorporating word problems of various complexities and covering a wide range of mathematical concepts. Our dataset encompasses a total of 14 comprehensive mathematical domains, including sets, trigonometry, binomial theorem, and more. The distribution of samples across these concepts is visually illustrated in Figure 4.6. Our dataset comprises a total of 223 samples, with the ”Sequence and Series” category notably having the highest number of problems, as indicated in the charts.

This study aims to improve the problem-solving capacities of LLMs within the field of mathe- matics. Initially, we noted that established publicly available models, including LLaMA [63] and
Vicuna [13], struggled with basic mathematical tasks like subtraction and addition. This insight became the catalyst behind our study, motivating us to improve LLMs ability to grasp and solve mathematical problems.

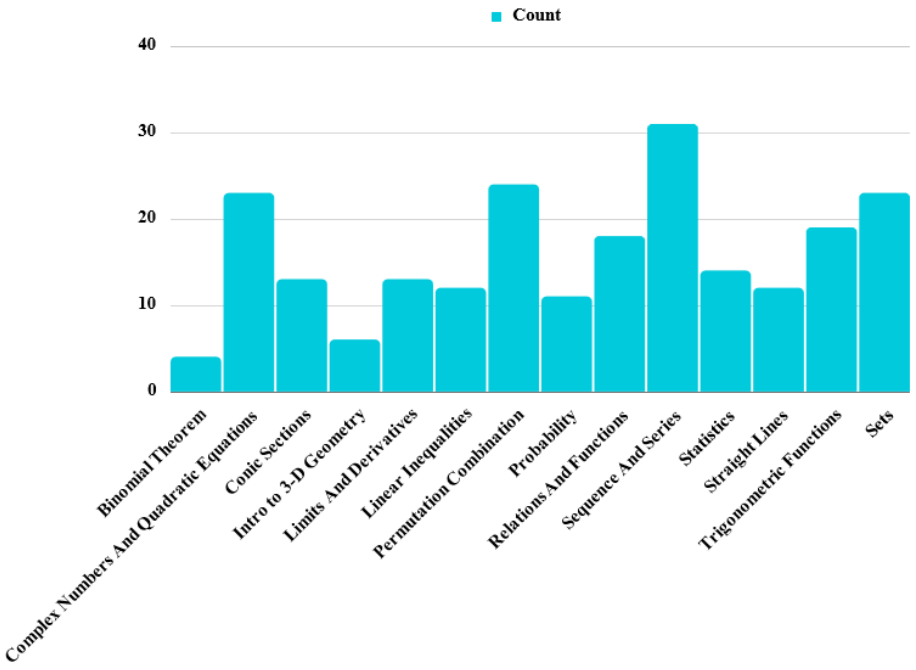


Figure 4.6: Distribution of the number of questions for each mathematical topic.

To achieve this goal, we used an instructional strategy similar to how children are taught mathematics. We started with basic operators like +, , and / before moving on to more sophisticated operators and expressions. In a same spirit, we wanted to familiarize LLMs with the meanings of numerous mathematical operators and expressions. To help with this procedure, we used the Math-401 dataset [77], which is a good resource that contains 401 data samples that include fundamental mathematics questions and their solutions. Given the dataset's limited size, we expanded it to add more diversity and complexity, guaranteeing that the model could learn and grasp advanced mathematical ideas throughout training. To fine-tune, we used three popular LLMs: MAmmoTH [78], WizardMath [45], and LLaMA-2 [64].

## III. Results & Analysis

In this section, we delve into the specifics of our conducted experiments, providing an overview of the experimental setup. We ran trials using three popular large language models: MAmmoTH, WizardMath, and LLaMA2. We tested both the 7B and 13B versions of these LLMs. Our ex- periments were conducted in two stages. In the first stage, we loaded the original model weights and performed inference on our test set. In the second stage, we fine-tuned the LLMs using the Math-401, which we have augmented in this research work.

The dataset was partitioned into 2.41K train, 30K test and 30K validation samples. We used QLora for fine-tuning, which optimizes memory and reduces computing costs in a pretrained language model by 4-bit quantization. Each model is fine-tuned for #10 epochs at a step size of 3e-4.

To evaluate performance, we measured accuracy by assessing the match between generated answers and the actual solutions for five open-source datasets: GSM-8K, DeepMind, SimulEq,

NumGLUE, and Math-401. These datasets offer ground truth answers, enabling the calculation of exact match accuracy.

Table. 4.8 shows the exact match accuracy of three models (7B and 13B variants) before fine-

| Model | # of Params | Accuracy | | | | | |
|---|---|---|---|---|---|---|---|
| | | GSM-8K | DeepMind | NumGLUE | SimulEq | Math-401* | MathQuest |
| **LLaMA-2** | 7B | 16.0 | 46.0 | 37.0 | 11.0 | 10.0 | 10.4 |
| **LLaMA-2** | 13B | 22.0 | 50.0 | 42.0 | 15.0 | 10.0 | 14.1 |
| **WizardMath** | 7B | 61.0 | 51.0 | 54.0 | 27.0 | 6.0 | 14.6 |
| **WizardMath** | 13B | 65.0 | 55.0 | 70.0 | 36.0 | 8.0 | 14.3 |
| **MAmmoTH** | 7B | 43.0 | 49.0 | 54.0 | 23.0 | 11.0 | 12.2 |
| **MAmmoTH** | 13B | 44.0 | 48.0 | 56.0 | 26.0 | 14.0 | 18.1 |

Table 4.8: Before fine-tuning, results for 100 samples from five datasets and our MathQuest dataset. (*) denotes the augmented subset of Math-401.

tuning on five datasets and our MathQuest dataset. Overall, performance is notably lower on the SimulEq dataset and our augmented Math-401 dataset. This is likely due to the presence of complex problems that require additional knowledge, such as questions like "Number of red color cards in a deck of 52 cards."

| Model | # of Params | Accuracy | | | | | |
|---|---|---|---|---|---|---|---|
| | | GSM-8K | DeepMind | NumGLUE | SimulEq | Math-401* | MathQuest |
| **LLaMA-2** | 7B | 30.0 | 46.0 | 45.0 | 15.0 | 17.0 | 10.6 |
| **LLaMA-2** | 13B | 42.0 | 51.0 | 54.0 | 16.0 | 24.0 | 20.3 |
| **WizardMath** | 7B | 64.0 | 55.0 | 52.0 | 29.0 | 15.0 | 16.01 |
| **WizardMath** | 13B | 68.0 | 56.0 | 70.0 | 38.0 | 10.0 | 20.1 |
| **MAmmoTH** | 7B | 56.0 | 50.0 | 62.0 | 24.0 | 16.0 | 18.5 |
| **MAmmoTH** | 13B | 67.0 | 51.0 | 64.0 | 34.0 | 18.0 | **24.0** |

Table 4.9: After fine-tuning, results for 100 samples from five datasets and our MathQuest dataset. (*) denotes the augmented subset of Math-401.

Table. 4.9 provides an in-depth analysis of the accuracy outcomes following the fine-tuning process. In summary, all models improved significantly in accuracy post fine-tuning on our heterogeneous question-and-answer dataset. We can also see that, models with 13B parameters were more accurate than those with 7B parameters.
The major findings from Tables 4.8 and 4.9 show that MAmmoTH-13B is the best-performing model for our MathQuest dataset, with the highest accuracy after fine-tuning (24.0%). It's worth noting that both MAmmoTH-7B and 13B produced results with precision up to two decimal places, demonstrating their accuracy. Table. 4.9 shows that MathQuest is a tougher difficulty due to its complexity and diversity, resulting in lesser accuracy when compared to other datasets.

**Paper Conclusion**

In conclusion, this research work provides Large Language Models (LLMs) with critical reasoning abilities for exact mathematical problem solution. The MathQuest dataset includes customizable question-and-answer pairs that address one or more mathematical operators as well as expressions. These challenges direct the model's approach to incremental problem resolution, with the goal of improving solution clarity and precision. Our findings show considerable gains in solution correctness and comprehensibility, which will be useful for educators and students looking to improve their problem-solving mathematical ability.

While this study provides a solid foundation for using Generative LLMs to advance mathematical problem-solving, further adjustments and optimizations are required to broaden its application to a wider range of contexts. Finally, our research helps to increase conceptual comprehension and numerical problem-solving abilities in high school-level mathematical question-answering, providing essential aid to pupils as well as professionals dealing with challenging questions via LLMs.

## IV. Limitations & Future Scope

While the proposed solution effectively handles simple mathematical problems, it occasionally faces difficulties when confronted with complex mathematical scenarios that require retaining variable values for subsequent equations. Additionally, our work exhibits a limitation concerning the partial enhancement of LLMs' reasoning abilities in solving mathematical problems. However, it struggles to address complex expressions containing nested brackets within equations.

In our future work endeavors, we target to address these drawbacks by expanding our training dataset. Given the rapid pace of advancements in LLM research, with new techniques, models, and prompting strategies emerging daily, we plan to integrate more advanced techniques to enhance LLM reasoning capabilities. This includes leveraging prompting techniques such as Recall, CoT, and Self-Consistency CoT, as well as advanced techniques like RLHF. By incorporating these methodologies, we seek to further refine LLMs' reasoning abilities and effectively address the challenges posed by complex mathematical problems.

## References:


[1]. Riaz Ahmad, Muhammad Tanvir Afzal, and Muhammad Abdul Qadir. Information extraction from pdf sources based on rule-based system using integrated formats. In Semantic Web Challenges: Third SemWebEval Challenge at ESWC 2016, Heraklion, Crete, Greece, May 29-June 2, 2016, Revised Selected Papers 3, pages 293–308. Springer, 2016.

[2]. Alammar, J (2018). The Illustrated Transformer, https://jalammar.github.io/illustratedtransformer/.

[3]. Avinash Anand, Krishnasai Addala, Kabir Baghel, Arnav Goel, Medha Hira, Rushali Gupta, and Rajiv Ratn Shah. Revolutionizing high school physics education: A novel dataset. In Big Data and Artificial Intelligence: 11th International Conference, BDA 2023, Delhi, India, December 7–9, 2023, Proceedings, page 64–79, Berlin, Heidelberg, 2023. Springer-Verlag.

[4]. Avinash Anand, Arnav Goel, Medha Hira, Snehal Buldeo, Jatin Kumar, Astha Verma, Rushali Gupta, and Rajiv Ratn Shah. Sciphyrag - retrieval augmentation to improve llms on physics q&a. In Big Data and Artificial Intelligence: 11th International Conference, BDA 2023, Delhi, India, December 7–9, 2023, Proceedings, page 50–63, Berlin, Heidelberg, 2023. Springer-Verlag.

[5]. Avinash Anand, Mohit Gupta, Kritarth Prasad, Navya Singla, Sanjana Sanjeev, Jatin Kumar, Adarsh Raj Shivam, and Rajiv Ratn Shah. Mathify: Evaluating large language models on mathematical problem solving tasks, 2023.

[6]. Avinash Anand, Raj Jaiswal, Pijush Bhuyan, Mohit Gupta, Siddhesh Bangar, Md. Modassir Imam, Rajiv Ratn Shah, and Shin'ichi Satoh. Tc-ocr: Tablecraft ocr for efficient detection & recognition of table structure & content. In Proceedings of the 1st International Workshop on Deep Multimodal Learning for Information Retrieval, page 11–18, New York, NY, USA, 2023. Association for Computing Machinery.

[7]. Avinash Anand, Raj Jaiswal, Mohit Gupta, Siddhesh S Bangar, Pijush Bhuyan, Naman Lal, Rajeev Singh, Ritika Jha, Rajiv Ratn Shah, and Shin'Ichi Satoh. Ranlaynet: A dataset for document layout detection used for domain adaptation and generalization. In Proceedings of the 5th ACM International Conference on Multimedia in Asia, New York, NY, USA, 2024. Association for Computing Machinery.

[8]. Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character region awareness for text detection. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 9357–9366, 2019.

[9]. Maciej Besta1, Nils Blach, Ales Kubicek, Robert Gerstenberger, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Michał Podstawski, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. Graph of thoughts: Solving elaborate problems with large language models, 2023.

[10]. Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler,

[11]. Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers, 2020.
[12]. Raghavendra Chalapathy, Ehsan Zare Borzeshi, and Massimo Piccardi. Bidirectional lstm-crf for clinical concept extraction, 2016.
[13]. Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023.
[14]. Xiyang Dai, Yinpeng Chen, Jianwei Yang, Pengchuan Zhang, Lu Yuan, and Lei Zhang. Dynamic detr: End-to-end object detection with dynamic attention. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 2988–2997, 2021.
[15]. Yonghao Dang, Fuxing Yang, Baiquan Su, Jianqin Yin, and Jun Liu. Dbnet: A new generalized structure efficient for classification. In 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO), pages 1–6, 2019.
[16]. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.
[17]. Yuning Du, Chenxia Li, Ruoyu Guo, Xiaoting Yin, Weiwei Liu, Jun Zhou, Yifan Bai, Zilin Yu, Yehua Yang, Qingqing Dang, and Haoshuang Wang. Pp-ocr: A practical ultra lightweight ocr system, 2020.
[18]. Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation, 2015.
[19]. Liangcai Gao, Yilun Huang, Herve Dejean, Jean-Luc Meunier, Qinqin Yan, Yu Fang, Florian Kleber, and Eva Lang. Icdar 2019 competition on table detection and recognition (ctdar). In ICDAR 2019, pages 1510–1515, 09 2019.
[20]. Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: Program-aided language models, 2023