**Parameter-efficient fine-tuning of Large Language Models in Math Education**

**Grant Proposal**

Ashwin Sivagaminathan

Massachusetts Academy of Mathematics and Science at Worcester Polytechnic Institute

85 Prescott St., Worcester, MA, 01605

**Executive Summary**

Artificial intelligence is used widely in professional and educational domains but still has many shortcomings. This paper addresses the limitations of large language models in solving math problems at a grade school level. Prior studies have shown that people are receptive towards using AI, but they have also outlined specific failures of AI, such as simple calculation mistakes which lead to overall lower accuracy, or confusing and verbose solutions that prevent students from using the full potential of AI. In the initial stages, huggingface.co was used as it is the premier resource for free AI models and its python library is very user-friendly. An open-source solution was preferred to keep costs down. Simple and small datasets were chosen and the results showed that the model used was capable of learning how to solve that specific type of problem. However, the preliminary phase was only effective because it was not a diverse set of data and as a result the model overfitted to the data. In the next phase, a larger, more diverse dataset will be trained, which will increase computational costs beyond what is feasible on most personal computers. To offset higher costs, parameter efficient fine-tuning methods will be employed, and cloud hosting will be used for the model to ensure there are no limitations for testing. Expected results will be that this model will be capable of solving a diverse set of math problems. The application of this work will be that the model will be useful in the educational sphere as a tool to help students study math in a world where education is more competitive than ever.

## Parameter-efficient fine-tuning of Large Language Models in Math Education

The use of artificial intelligence is becoming more prevalent in everyday life. Professionals in many fields, such as finance, healthcare, and business, are looking forward to utilizing its capabilities to increase their own productivity (GiniMachine, 2024). At the same time, image generation and chat features are being enjoyed by many people for personal and recreational purposes. The latter has seen a huge growth in popularity, with large language models like ChatGPT fascinating the public.

As a whole, AI has entered the mainstream and is very widely discussed but is often misunderstood or misused as a term. To clarify this issue, we specifically define artificial intelligence as the theory behind the development of computer systems to perform tasks that traditionally require human-like intelligence. AI is a very wide discipline that covers text generation, image recognition, predictive analytics, sentiment analysis, and other types of models. Of these, natural language processing, a form of text generation, is one of the most well-known. Natural language processing is the ability of computers to interpret and output human language (Google Cloud Tech, 2023).

Natural language processing is carried out by large language models. Large language models are made with very high numbers of parameters and are trained on lots of data. They are often fine-tuned for specific purposes. Large language models are immensely powerful, but still have shortcomings, such as inheriting biases from the training data. Also, more general models are expected to be competent in many domains due to the vast amounts of data they are trained on, but this is not always the case. One example of such a gap is mathematical reasoning, which is to be investigated in this project. For example, in a study on ChatGPT's ability to solve high school level math problems, they found that it performed poorly on problems that required applications of skills, as seen in Figure 1.

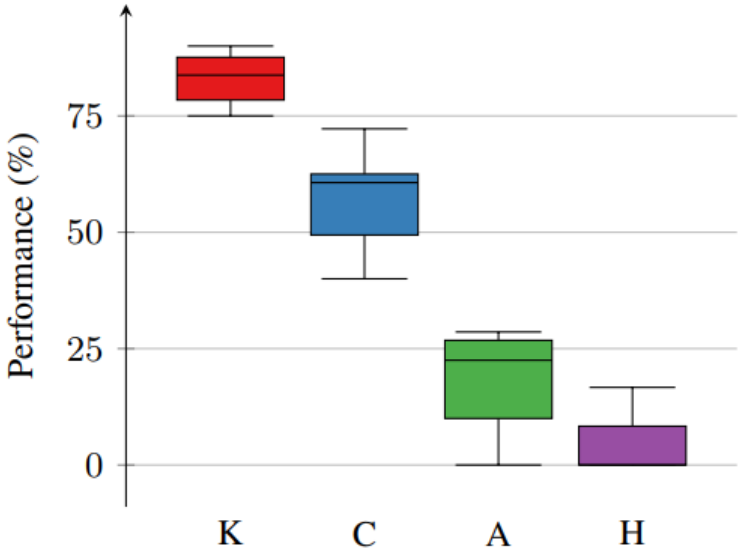


*Figure 1: ChatGPT's accuracy on various levels of difficulty in Vietnamese High School math problems. In ascending order of difficulty, K, C, A, and H represent Knowledge, Comprehension, Application, and High Application. The dataset that was used was published by the Vietnamese Ministry of Education and Training (Dao & Le, 2023).*

**Past Research**

Previous work has researched the impact of artificial intelligence in math education, and many shortcomings have been found. One of these is that educators struggle to use artificial intelligence due to a lack of communication between educators and artificial intelligence developers (Xie et al., 2021). As far as the models themselves are concerned, even advanced large language models like GPT-4 struggle with certain aspects of solving math problems. In previous studies of the effectiveness of artificial intelligence in assisting individuals with undergraduate level math problems, GPT-4 was often found to make mistakes in algebra, rely too much on memorized solutions (rather than creating a solution to help a student solve, for example), and use an amount of verbosity that was quite high, making it unpleasant to read, and difficult for students to understand the solution. (Collins et al., 2024). Ideally, these models

would be able to not only solve math problems effectively but also explain their solutions in an efficient manner. Unfortunately, in the current state, neither of these goals are being met to the level that they should be. For example, another study looked at a specific method of increasing the effectiveness of large language models in solving high school level math problems. In a novel approach, they trained separate verifier models to check a model's solution and thus increase its accuracy, in theory. While it was effective to an extent, a large language model with 175 billion parameters using the verifier was not even able to achieve 60% accuracy on these math problems, as seen in Figure 2. With such a high budget and resource intensive solution, higher quality outputs should be expected.

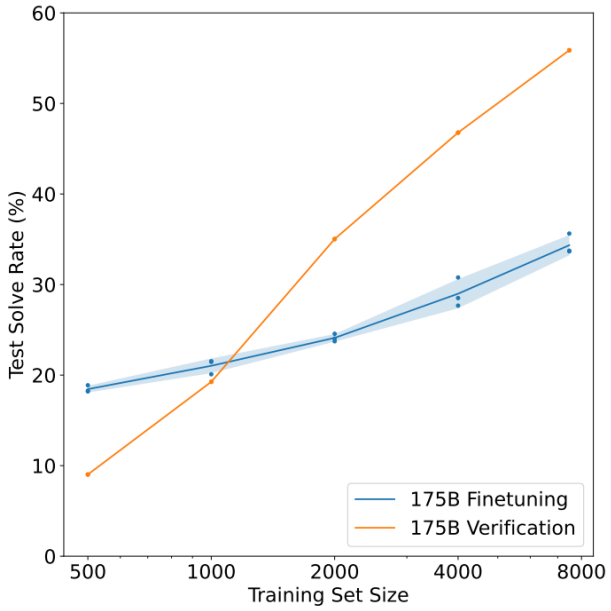


*Figure 2: Comparison between finetuning and verification of a model with 175 billion parameters. Total fine-tuning is not necessary to achieve high accuracy and may even be outclassed by novel methods. Still, there is lots of room for improvement as the accuracy did not exceed 60% for any model in the study (Cobbe et al., 2021).*

The premier language model in the educational space is Khanmigo. Khanmigo was created by Khan Academy and its base model is GPT-4. Despite the fact that Khan Academy is a nonprofit,

philanthropic company, Khanmigo still requires a subscription to use for people who are not educators. Therefore, it is clear that Khanmigo is very resource intensive. Additionally, Khanmigo was made with teaching in mind, so it emphasizes the process of solving a given problem and does not solve a problem on its own. While Khanmigo is helpful for educators, it means that it cannot be used to test the limits of artificial intelligence in terms of raw math problem solving ability (Ofgang, 2024).

## Section II: Specific Aims

This proposal's objective is to find and utilize novel and efficient methods in the field of fine-tuning large language models in order to improve their effectiveness at solving high school level math problems. Because state-of-the-art models are usually locked behind paywalls, it may seem that achieving mathematical reasoning is unfeasible to do in a cost-effective manner. However, by coming to a balance between cost and model power, adequate results can be expected.

Large language models are very resource intensive. They usually have billions of parameters, or weights learned from the training data, and often require petabytes of data to be pre-trained. Pre-training is the process of creating the model itself that can then be deployed to various tasks with only a little bit of fine-tuning by comparison. Access to development of large language models themselves, therefore, is only open to a select few very large companies, and is out of the scope of this project (Google Cloud Tech, 2023). Thankfully, access to smaller models exists for free for preliminary testing, while state-of-the-art models can be worked with for a price. Because fine-tuning requires fewer resources than pre-training, it will be the focus. However, it still stands that fine-tuning is inaccessible to many. In general, fine-tuning involves adjusting the model parameters using domain-specific data to achieve higher accuracy in a relevant task. Total fine-tuning involves adjusting every single one of those billions of parameters.

The computation required to fine-tune every parameter in a model depends on the model and data used. GPU renting depends on the model itself, so one may be able to save if they use a smaller

model. In general, the smaller the model, the cheaper it is to train. As seen in Figure 3A, the maximum batch size a model can be trained with depends mainly on the memory of the GPU and the size of the model. Batch size of a model in training is how much data is passed through the model at once. Maximum batch size is important because if more data can be passed through the model at once, it can be trained faster. As most GPU renting services charge over time, saving time saves money. The throughput of the model – tokens per unit of time – is how fast a model can produce outputs. A higher throughput may mean a higher cost per minute but ends up in a lower overall cost because the model spends less time creating an output. Figure 3B shows that throughput and maximum batch size are positively correlated, meaning that a smaller model will be cheaper to train and use. Researchers found that training the model Mixtral on the dataset OpenOrca could cost as low as $3460 depending on the GPU used (Xia et al., 2024).

$$Max\_BSZ = \lfloor C_0 * \frac{GPU\_mem - model\_mem}{seq\_len * ((1 - C_1) + C_1 * sparsity)} \rfloor$$

*Figure 3A: Maximum batch size depends on GPU memory and model size. A larger GPU will allow for a large batch size, while a smaller model will also as it takes up less of the GPU's memory (Xia et al., 2024).*

$$Throughput = C_2 * log(\frac{batch\_size}{sparsity * C3}) + C_4$$

*Figure 3B: Throughput depends on maximum batch size of the model. A higher throughput means the model is more efficient in training and needs less overall cost to train (Xia et al., 2024).*

While not as expensive as pre-training, total fine-tuning is still out of reach for most individuals and may even cause problems for small startups looking for a preliminary product. However, cheap methods of fine-tuning exist, which will be one of the specific aims of the project. The general term for

this is parameter-efficient fine-tuning, which means that not all parameters are modified during fine-tuning. Only some of the parameters need to be tweaked to capture the main nuances of the training data. One specific subcategory of parameter-efficient fine-tuning that will be studied is Low Rank Adaptation. Low Rank Adaptation involves adding new parameters to the model to be tuned. It freezes the original parameters which are much higher in count. The matrices of the new parameters multiply to a matrix that is the same size as the original. Thus, it takes far fewer parameters but minimizes loss of complexity (Shaw Talebi, 2023).

The long-term goal is to demonstrate that success with large language models can be achieved without reaching overwhelming costs. The rationale is that efficient fine-tuning methods do exist, and that open-source platforms for artificial development also exist. The work we propose here will look at raw mathematical problem-solving ability of large language models using parameter-efficient fine-tuning methods.

**Specific Aim 1: Save on computational costs when fine-tuning language models**

**Specific Aim 2: Test models trained on various methods at accuracy on math problems**

The expected outcome of this work is that methods such as Low-Rank Adaptation take much less time to perform than other methods. Additionally, these methods may see a slight compromise in performance. However, this decrease is expected to be low as the objective of such a method is to cut corners on computational costs without sacrificing much accuracy.

## Section III: Project Goals and Methodology

***Specific Aim #1:*** Use lower computational costs

The objective is to demonstrate that it is possible to work with large language models in an efficient and cost-effective manner. As seen in Figure 4, the costs of training these models are incredibly far away from the budget of the average person or small company. Even just fine-tuning these models

would be very expensive, even if only taking into account the sheer number of parameters. Therefore, it is important to look into cheaper methods of fine-tuning in order to cut down on costs and increase accessibility for individuals. Cheaper fine-tuning also has the added benefit of being environmentally friendly, as it would lead to less spending on equipment and less resource consumption, which leads to fewer carbon emissions.

| | GPT-3 large | LLaMa |
|---|---|---|
| Vocabulary size | 50,257 | 32,000 |
| Sequence length | 2048 | 2048 |
| Parameters in the largest model trained | 175B | 65B |
| Tokens in the training dataset | 300B | 1 – 3T |
| Number of GPUs | 10,000 V100 GPUs | 2048 A100GPUs |
| Training time | One month | 21 days |

*Figure 4: Comparison of hardware and software of two large language models. Both models reach billions in terms of parameter count and used thousands of GPUs. The amount of time taken to train each model was also very high, as expected of state-of-the-art models (Yeluri, 2023).*

**Methodology**

The methodology to investigate resource-efficient fine-tuning is as follows. Multiple methods of fine-tuning will be chosen. Prompt tuning and Low Rank Adaptation will be used. Prompt tuning involves adding prefixes to model inputs and has shown to be very parameter-efficient, while Low Rank Adaptation involves taking the weight matrix that makes up a model and creating two smaller matrices that then multiply to the size of the original one. The original weights remain unaffected (Lialin et al., 2024). In order to test the hardware requirements of these models, the time taken to run all of the fine-tuning will be measured and compared, between total fine-tuning, prompt tuning, and Low Rank Adaptation. This has the added benefit of setting up the testing of the next specific aims, because it will

mean the models are already ready to work with. The justification for using time as a measure of hardware requirements is that it is a simple value to measure, and that intuitively, processing time should correlate with memory usage. In order to prevent any memory limitations of a local device from impeding testing, especially as total fine-tuning of models may not be feasible with a personal computer, cloud services will be utilized.

***Specific Aim #2:*** Fine-tune language models to solve math problems

The above knowledge will be applied to the domain of mathematics. The Hugging Face API will be used to train models on grade-school level math problems. Hugging Face is an open-source AI platform that allows users to train models and access models and datasets, as well as offering many python libraries for training AI models (Hugging Face, 2024).

**Methodology**

Hugging Face hosts models from many top AI competitors, such as OpenAI and Google. For this project, a Google-BERT (Bidirectional encoder representations from transformers) model will be used. The Hugging Face transformers model will be used to train this model. Transformers allows users to easily fine-tune pretrained models using PyTorch, Tensorflow, which are machine learning python libraries (Hugging Face, 2024). Transformers supports parameter-efficient fine-tuning methods and total fine-tuning. Both will be used and accuracy will be compared.

**Summary of Preliminary Data.**

For preliminary testing purposes, the inverse-scaling/redefine-math dataset on Hugging Face was used (Shen, 2022). This dataset consists of 900 problems that require the model to differentiate between computation and direct reading of characters. Although the dataset is very small, it is simple enough and

lacks diversity of questions, so the model is able to learn the problems to a fairly high accuracy as seen in Figure 5.

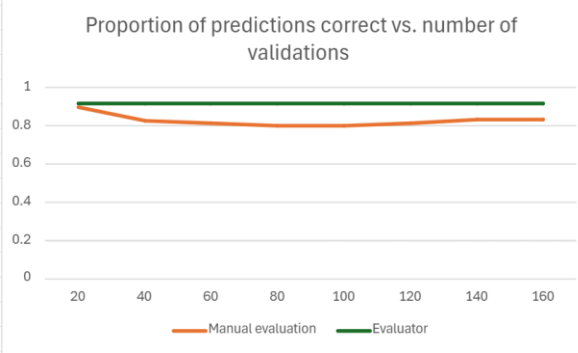


*Figure 5: Hugging Face evaluate class after training along with manual question answering. The model consistently scored above an 80% accuracy.*

**Expected Outcomes.** The overall outcome of this aim is to achieve high accuracy with both parameter-efficient fine-tuning and total fine-tuning. However, it is expected that using LoRA will result in a slightly lower accuracy than total fine-tuning. In the best-case scenario, there will not be a statistically significant difference between accuracy of LoRa and accuracy of total fine-tuning, which will suggest that both methods are viable for training large language models. Furthermore, if the accuracy can meet the standards of the preliminary data, it will show promise for using these methods in the domain of math problem solving.

## Section IV: Resources/Equipment

This project only requires the use of a personal computer. All resources, including cloud hosting, can be accessed using it. The specific laptop used is a Yoga Pro 9i with an NVIDIA GeForce RTX 4050 graphics card.

## Section VI: Timeline

August 2024 – September 2024

- **Brainstormed initial project ideas**
- **Started reading scientific literature**

September 2024 – December 2024

- **Majority of background research**
- **Locally trained preliminary model**

December 2024 – February 2024

- **Trained final model using parameter efficient methods**
- **Data collection**

**Section VIII: References**

Cobbe, K., Kosaraju, V., Bavarain, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., Schulman, J. (2021). Training Verifiers to Solve Math Word Problems. https://arxiv.org/abs/2110.14168.

Collins, K. M., Jiang, A. Q., Frieder, S., Wong, L., Zilka, M., Bhatt, U., Lukasiewicz, T., Wu, Y., Tenenbaum, J. B., Hart, W., Gowers, T., Li, W., Weller, A., & Jamnik, M. (2024). Evaluating language models for mathematics through interactions. *Proceedings of the National Academy of Sciences of the United States of America*, *121*(24). https://doi.org/10.1073/pnas.2318124121

Dao, X.-Q., & Le, N.-B. (2023, October 10). *Investigating the Effectiveness of ChatGPT in Mathematical Reasoning and Problem Solving: Evidence from the Vietnamese National High School Graduation Examination*. ArXiv. https://arxiv.org/abs/2306.06331

GiniMachine. (2024, July 17). *10 fields that use artificial intelligence and machine learning*. https://ginimachine.com/blog/fields-that-use-artificial-intelligence/

Google Cloud Tech. (2023, May 8). *Introduction to Generative AI* [Video]. YouTube. https://youtu.be/G2fqAlgmoPo

Google Cloud Tech. (2023, May 8). *Introduction to large language models* [Video]. YouTube. https://youtu.be/zizonToFXDs

Hugging Face. (2024). *Hugging Face – On a mission to solve NLP, one commit at a time.* Huggingface.co. https://huggingface.co/

Lialin, V., Desphande, V., Xiaowei, Y., Rumshisky, A. (2024). Scaling Down to Scale Up: A Guide to Parameter-Efficient Fine-Tuning. arxiv.org/abs/2303.15647.

Ofgang, E. (2024, April 17). *What is Khanmigo? The GPT-4 Learning Tool explained by Sal Khan*. TechLearningMagazine. https://www.techlearning.com/news/what-is-khanmigo-the-gpt-4-learning-tool-explained-by-sal-khan

Shaw Talebi. (2023, Oct 1). *Fine-tuning Large Language Models (LLMs) | w/ Example Code* [Video]. YouTube. https://youtu.be/eC6Hd1hFvos

Shen, X. (2022, October 8). *inverse-scaling/redefine-math* [Review of *inverse-scaling/redefine-math*]. Huggingface.co. https://huggingface.co/datasets/inverse-scaling/redefine-math/tree/main

Xie, H., Hwang, G. J., & Wong, T. L. (2021). Editorial Note: From Conventional AI to Modern AI in Education: Re-examining AI and Analytic Techniques for Teaching and Learning. *Educational Technology & Society*, *24*(3), 85-88. https://doi.org/10.30191/ETS.202107_24(3).0006

Yeluri, S. (2023, August 10). *Large Language Models — the hardware connection*. APNIC Blog. https://blog.apnic.net/2023/08/10/large-language-models-the-hardware-connection/

Xia, Y., Kim, J., Chen, Y., Ye, H., Kundu, S., Hao, C., C., & Talati, N. (2024, August 8). Understanding the Performance and Estimating the Cost of LLM Fine-Tuning. https://arxiv.org/abs/2408.04963