Elementary Mathematics Solving with Deep Learning

Diep Duc Duong
Nguyen Dinh Le
Nguyen Hoang Nguyen

Abstract

Large Language Models are currently trending and hold significant potential for various applications in our daily lives. However, one of the challenges faced by these models and other AI systems is performing mathematical and reasoning tasks effectively. In this work, we tackled the Zalo AI Challenge 2023: Elementary Maths Solving and experimented with different deep learning models including CNN-1D, LLM barebone and LLM with Chain-of-Thought prompting and evaluated the performance of each method then drawing conclusions and comments on how to improve future results.

Keywords: Large Language Model, Mathematical Word Problems Solving, Elementary Mathematics Solving

1. Introduction

Recent years advancement in LLM models has empowered more applications of machine mathematical reasoning and math word problem solving abilities. The motivation for this work comes from the fact that parents can use a chatbot like ChatGPT to support checking their children's math homework or students can find their own solutions to difficult math problems.

In this work, we explored various deep learning methods to work on the Elementary Math Solving of Zalo AI Challenge 2023 [1] using its provided structure dataset in JSON format, with each sample including id, question, choices, answer and explanation fields in Vietnamese.

Our work includes three main steps. First we built a baseline CNN-1D QA to train with the train dataset from scratch. Second, we tried the GPT-3.5 model (using OpenAI API) with zero-shot, one-shot and few-show prompting. Finally, we applied Chain-of-Thought prompting on the same GPT-3.5 model and then compared the results.

2. Related work

Zhao, Xu, et al [2] pioneer an innovative approach to enhance language model performance through strategic model combination. Demonstrating its prowess on the GSM8K [7] and SVAMP [6] datasets, the proposed method achieves remarkable accuracies of 96.8% and 93.7%, respectively. Notably, it outperforms PAL [9, 10] by a significant margin, showcasing a 3.2% improvement on GSM8K [7], even with a lower

individual accuracy of 64.4% for CoT [8]. The authors provide both theoretical analysis and empirical results to substantiate the feasibility of their model combination strategy, emphasizing its potential for reducing computation costs while delivering notable performance improvements across various datasets with diverse backbone LLMs.

In their pioneering work, Zheng, Chuanyang, et al. [3] introduce a revolutionary strategy aimed at enhancing the efficiency of automatic interactions between users and language models (LLMs). Their approach, known as the Progressive-Hinting Paradigm (PHP), leverages previously generated answers as hints, guiding the language model towards more accurate responses over time. This innovative methodology demonstrates remarkable performance improvements, particularly excelling in mathematical reasoning tasks, leading to state-of-the-art outcomes across multiple reasoning benchmarks. Notably, when applied with text-davinci-003, PHP exhibited a 4.2% improvement on the GSM8K [7] dataset using greedy decoding compared to Complex CoT [8], along with a substantial 46.17% reduction in sample paths through self-consistency. Additionally, when employed in conjunction with GPT-4, PHP achieved state-of-the-art performances on the MATH benchmark, showcasing a notable improvement from 50.3% to an impressive 53.9%. The authors underscore PHP's adaptability to more potent models and prompts, highlighting its compatibility with other techniques like CoT [8] and self-consistency. Of particular significance is PHP's orthogonal nature, enabling seamless integration with existing state-of-the-art methods and, potentially, unlocking further performance enhancements in the realm of LLMs.

3. Approach

3.1. Baseline: CNN-1D QA

For the multiple choice question answering problem, we propose to model the problem as the distribution $P(y = i|q, o_1, o_2, o_3, o_4), i = 1, 2, 3, 4$, where q is the representation of the question, o_i is the representation of the i_{th} choice, $y = i$ is equivalent to the answer to the question being the i_{th} choice. Let $g(i) = P(y = i|q, o_1, o_2, o_3, \dots)$.

To learn the g function, we proposed using the CNN-1D architecture, illustrated in Figure 1. Wherein, the text representation of the question and choices will be tokenized into integers, then go through the embedding layer to get real-valued vector representations. The representation vectors of the question and choices then are fed into the $QuestionEncoder$ and $OptionEncoder$ components respectively, which are implemented by CNN-1D networks, in order to process the initial representation vectors and output representations that are complex enough to fit the goal of the problem. Finally, the CNN-1D QA network computes the dot product between the question representation vector and the representation vectors of i_{th} choice, denoted as $score_i$. The $score_i$ values

are fed into the softmax function, the output of the softmax function is considered as the values of the $g(i)$ function - or the distribution $P(y|q, o_1, o_2, o_3, \dots)$.

Because the number of choices for each question can be different, to ensure the correctness of the model and take advantage of GPU's parallel computing capabilities, we set a maximum number of choices that each sample can have and use the masking technique from masked attention **[4]**. The maximum number of choices is 4, for samples that do not have 4 choices, an o_{pad} vector is substituted for the missing choice's input vector. After that, the $QuestionEncoder(q) * OptionEncoder(o_{pad})$ value is set to negative infinity, and thus the predicted probability for the missing choice will be approximately 0.

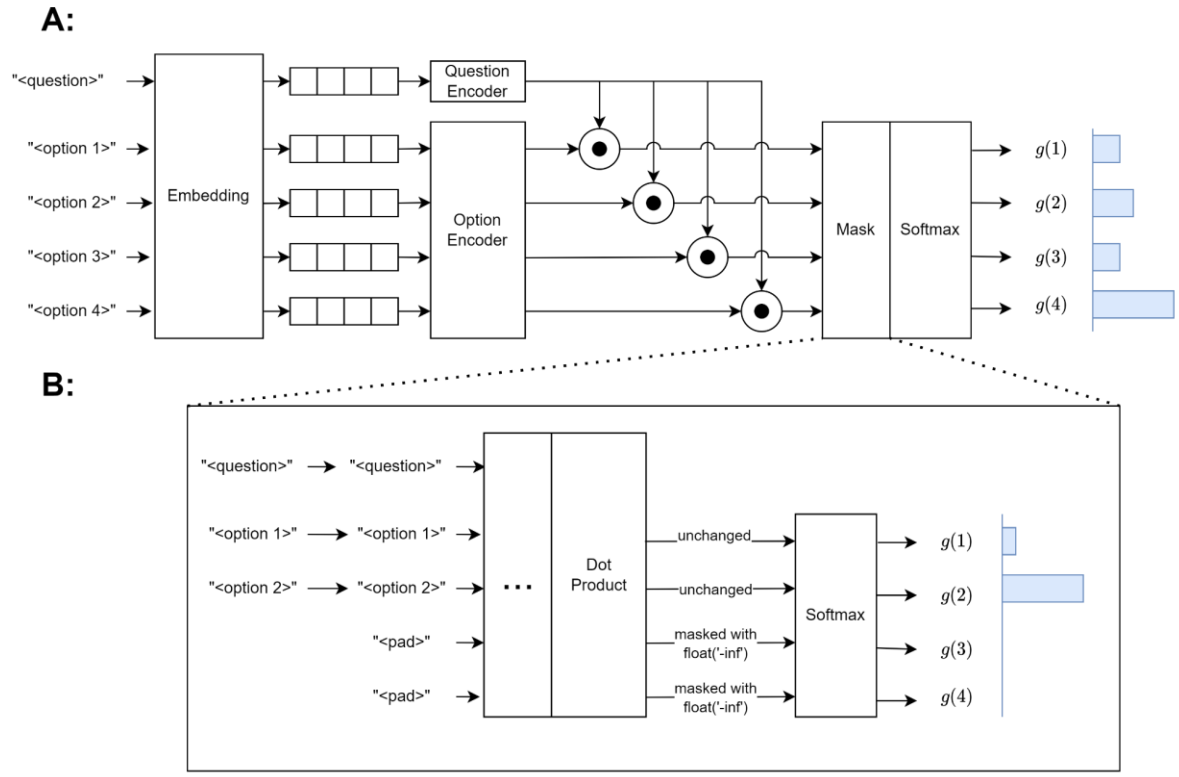

Figure 1: (A) Overview of the CNN-1D QA model. (B) Masking details

3.2. Large Language Models

In recent times, large language models have revolutionized the field of natural language processing. Many LLMs are trained on large amounts of multilingual data and have high generalization capabilities. With new training techniques, LLMs can even be used as multifunctional chatbots. In this research, we also experiment with using LLMs for the Vietnamese elementary math problem, specifically the GPT-3.5 model from OpenAI. We test prompting GPT-3.5 in many scenarios, including zero shot, one shot and few shot learning, with or without CoT [8].

3.3. Chain-of-Thought

The effectiveness of LLMs depends not only on the model weights itself, but also on the quality of the input prompts [5]. LLMs can make better predictions if the prompts require the LLM to provide step-by-step reasoning rather than just the desired result; this prompting technique is called Chain-of-Thought [5]. We further investigate the effectiveness of LLMs when applying the CoT [8] technique, with zero shot, one shot and few shot learning scenarios.

4. Experiments

4.1. Dataset

We collected the Elementary Math dataset from the Zalo AI Challenge 2023 [1]. The dataset includes the train set, with 1200 samples and the public test set, with 189 samples. The data samples are math questions, covering many topics like arithmetic, geometry, logic, combinations, sorting, etc. Each question has multiple choices and only one correct answer. Each sample may have 3 to 4 options. In the training set, some samples also include the explanation field, which is the solution for the answer. We split this dataset into 3 sets: train, validation and test. The number of samples in each set is shown in Table 1.

Train (77.75%)	**Validation (8.64%)**	**Public Test (13.61%)**
1080 samples = 664 without explanations and 416 have explanations	120 samples with explanations	189 samples without answer

Table 1: Dataset

4.2. Metrics

The metric used in the competition is accuracy.
accuracy = {number of correct answers} / {total number of questions}.

4.3. Baseline: CNN-1D QA

4.3.1. Experiment Details

In this experiment, we use the CNN-1D QA model presented in the previous section. The model is trained with the Cross-Entropy loss function; techniques, like skip connections, dropout, L2-norm, and early stopping, are applied to increase model accuracy. A more

detailed description of the encoder components is shown in Figure 2. The hyperparameters in this experiment are shown in Table 2.

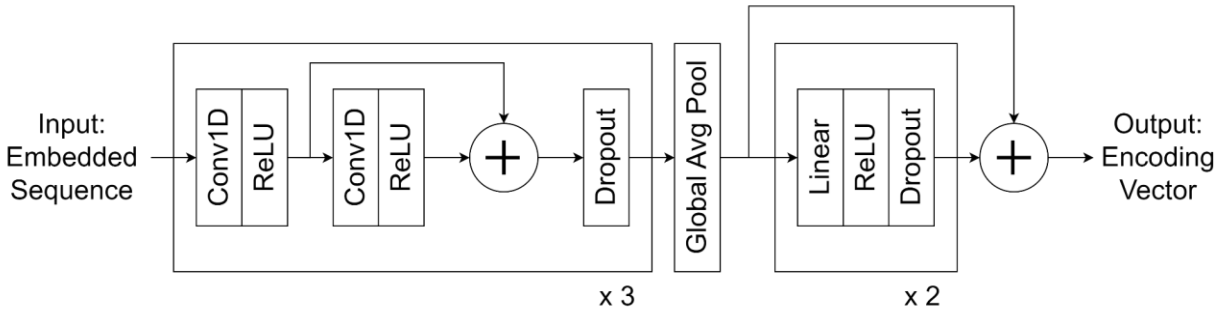

Figure 2: Encoder component

Architectural details	See the model's notebook source code
Optimizer and learning rate scheduler	AdamW and Cosine Annealing
Learning rate	3.10^{-3}
Weight decay	5.10^{-3}
Batch size	32
Number of epochs	1000 with early stopping

Table 2: Hyperparameters in experimenting with CNN-1D QA.

4.3.2. Results

The model achieved an $accuracy = 0.29947$ on the public test set. This result is better than random choice ($accuracy = 0.23529$), proving that the model has learned useful patterns for elementary math problem-solving.

4.4. Large Language Model (LLM)

4.4.1. Experiment Details

We evaluated the effectiveness of GPT-3.5 on the public test set in three scenarios: zero-shot, one-shot and few-shot. In the few-shot scenario, 3 question-answer examples were used. We used the API's API to generate prompts, the gpt-3.5-turbo-16k model and set the temperature parameter to 0 to maximize reproducibility. The prompt template that we used is shown in Table 3. Examples were taken from the train and validate sets.

System	You are a helpful assistant that solves Vietnamese word math problems in the form of multi-choice questions, specifically, you have to choose the one correct option among multiple options and your reply must contain no more information other than the chosen option itself.	
	Example user	**Example assistant**
Example #1	Một xưởng may trong tuần thứ nhất thực hiện được 3/8 kế hoạch tháng, tuần thứ hai thực hiện được 3/16 kế hoạch, trong tuần thứ ba thực hiện được 1/3 kế hoạch. Để hoàn thành kế hoạch của tháng thì trong tuần cuối xưởng phải thực hiện bao nhiêu phần kế hoạch?\nA. 5/48\nB. 43/48\nC. 11/48\nD. 27/48	A. 5/48
Example #2	Mệnh đề nào sau đây là phủ định của mệnh đề 'Mọi động vật đều di chuyển'?A. Mọi động vật đều không di chuyển\nB. Mọi động vật đều đứng yên\nC. Có ít nhất một động vật không di chuyển\nD. Có ít nhất một động vật di chuyển	C. Có ít nhất một động vật không di chuyển
Example #3	Một bài trắc nghiệm khách quan có 10 câu hỏi. Mỗi câu có 4 phương án trả lời. Có bao nhiêu phương án trả lời?A. 40\nB. 10^4\nC. 4^10\nD. 4	C. 4^10

Table 3: Prompt template. Experimenting zero-shot uses only the system part, one-shot uses example number 1, few-shot uses all 3 examples.

4.4.2. Results

The accuracy of zero-shot, one-shot and few-shot were 0.41711, 0.41176, 0.4492 respectively; much higher than the from-scratch trained CNN-1D QA model and the few-shot prompting achieved the highest accuracy.

4.5. Chain-of-Thought

4.5.1. Experiment Details

We combined the CoT technique with the GPT-3.5 model on the public test set in three scenarios: zero-shot, one-shot and few-shot. In the few-shot scenario, 3 question-answer examples were used. We used the OpenAPI API to generate prompts, the gpt-3.5-turbo-16k model and set the temperature parameter to 0 to maximize reproducibility. The prompt template that we used is shown in Table 4. Examples were taken from the train and validation sets.

System	You are a helpful assistant that solves Vietnamese word math problems in the form of multi-choice questions, specifically, you have to choose the one correct option among multiple options. You solve the question step by step and explain along with the answer. Your reply must be in the form of \"Reason: {your_reasoning_step_by_step}\n\nAnswer: {your_chosen_option}\".	
	Example user	**Example assistant**
Example #1	Một xưởng may trong tuần thứ nhất thực hiện được 3/8 kế hoạch tháng, tuần thứ hai thực hiện được 3/16 kế hoạch, trong tuần thứ ba thực hiện được 1/3 kế hoạch. Để hoàn thành kế hoạch của tháng thì trong tuần cuối xưởng phải thực hiện bao nhiêu phần kế hoạch?\nA. 5/48\nB. 43/48\nC. 11/48\nD. 27/48	Reason: Để hoàn thành kế hoạch của tháng thì trong tuần cuối xưởng phải thực hiện:\n1 − (3/8 + 3/16 + 1/3) = 5/48 (kế hoạch)\n\nAnswer: A. 5/48
Example #2	Mệnh đề nào sau đây là phủ định của mệnh đề 'Mọi động vật đều di chuyển'?A. Mọi động vật đều không di chuyển\nB. Mọi động vật đều đứng yên\nC. Có ít nhất một động vật không di chuyển\nD. Có ít nhất một động vật di chuyển	Reason: Mệnh đề \"Có ít nhất một động vật không di chuyển\" là phủ định của mệnh đề \"Mọi động vật đều di chuyển\"\n\nAnswer: C. Có ít nhất một động vật không di chuyển
Example #3	Cho các đoạn thẳng AB = 4cm; MN = 5cm; EF = 3cm; PQ = 8cm; IK = 7cm . Sắp xếp độ dài các đoạn thẳng theo thứ tự tăng dần?\nA. EF, AB, MN, IK, PQ\nB. PQ, IK, MN, AB, EF\nC. EF, AB, IK, PQ, MN\nD. EF, MN, IK, PQ, AB	Reason: Độ dài các đoạn thẳng theo thứ tự tăng dần là: EF, AB, MN, IK, PQ\n\nAnswer: A. EF, AB, MN, IK, PQ

Table 4: Prompt template with CoT. Experimenting with zero-shot uses only the system part, one-shot uses example number 1, few-shot uses all 3 examples.

4.5.2. Results

The accuracy of zero-shot, one-shot and few-shot were 0.5508, 0.6738, 0.65241 respectively; much higher than the from-scratch trained CNN-1D QA model and LLM without CoT. The one-shot Chain-of-Thought scenario achieved the highest accuracy. As of 18:00 on November 26, 2023, our one-shot CoT method ranked 46th (equal to the result and later submission than rank 45) out of a total of 136 participating teams.

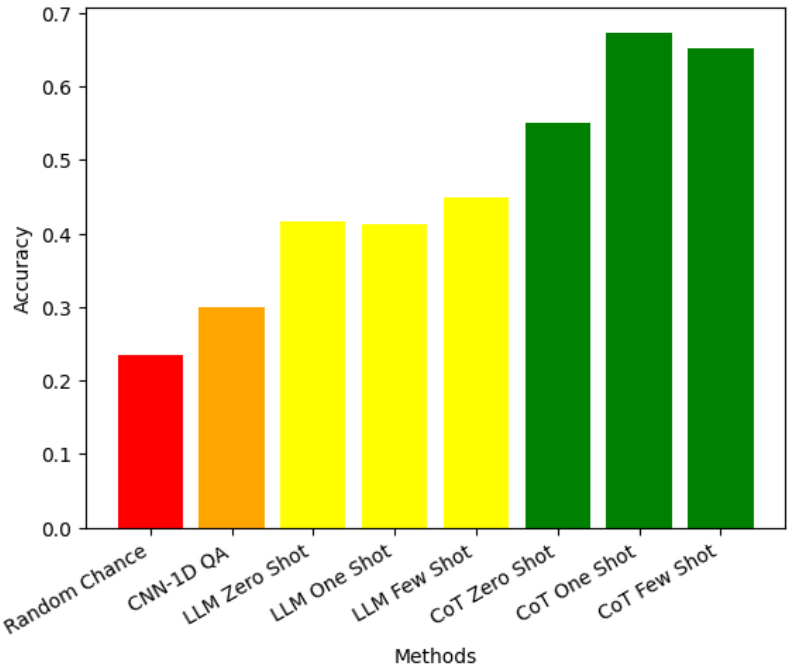

Figure 3: Results of the methods on the test set

5. Analysis

Since the public test set labels are not publicly available and there is a time limit, we did not collect the true labels of the test set and only compared based on the sole metric provided by the competition which is accuracy.

The CNN-1D QA model can learn from old data and make predictions for unseen data, with better accuracy than random guessing. However, the difference from random guessing is not high. The architecture of CNN-1D QA is still naive for a task with diverse, heterogeneous and complex relationships between data components like elementary math. Furthermore, CNN-1D QA was only trained on a small amount of data - around 1000 samples, so it could not leverage other knowledge, including explanation data. In contrast, GPT-3.5 has a much larger capacity, trained on much more data, including Vietnamese data. GPT-3.5 significantly outperforms the from-scratch trained model like CNN-1D QA, and performs better with more examples.

When combining GPT-3.5 and the Chain-of-Thought technique, the model's accuracy is further enhanced. Even in the zero-shot scenario, the chain-of-thought method still performs better than any scenario using GPT-3.5 without Chain-of-Thought. In elementary math problems, Chain-of-Thought generally performs better when accompanied by one or more examples.

6. Conclusion

In this study, we have shown the potential of applying deep learning techniques to elementary math problem-solving tasks, from small trained-from-scratch models to large language models. In addition to advantages, deep learning models also have disadvantages that need to be addressed. CNN-1D QA did not fully leverage all possible knowledge. GPT-3.5, along with Chain-of-Thought, achieved good results but could be better if the model is fine-tuned for the Vietnamese language and for the task of solving Vietnamese math word problems. Therefore, we propose the following future research directions:

- Use data augmentation methods to enrich training data, as well as give the model the ability to handle invariance to some input changes, for example: "A has 5 apples, then is given 3 more apples. Ask how many apples A has" should give the same result as "A has 3 apples, then is given 5 more apples. Ask how many apples A has", "sort the numbers 3, 5, 4 in ascending order" should give the same result as "sort the numbers 4, 5, 3 in ascending order".

- Leverage other knowledge sources, including (1) training along with Vietnamese math corpora, for example, exam questions, and math textbooks (2) leveraging Vietnamese language models and performing transfer learning.

- Better problem modelling, one of which is modelling reasoning ability. For this issue, we have imputed the Zalo AI Challenge 2023 [1] dataset to have full explanations for the train set and will experiment with this improvement in the future. We also used GPT-3.5 with suitable prompts to automatically generate explanations for questions and their corresponding answers.

7. References

[1] Zalo AI Challenge 2023. *https://challenge.zalo.ai*. (2023)

[2] Zhao, Xu, et al. "Automatic Model Selection with Large Language Models for Reasoning." arXiv preprint arXiv:2305.14333 (2023).

[3] Zheng, Chuanyang, et al. "Progressive-hint prompting improves reasoning in large language models." arXiv preprint arXiv:2304.09797 (2023).

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, (2017)

[5] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *Advances in Neural Information Processing Systems.* (2022)

[6] Patel, Arkil, Satwik Bhattamishra, and Navin Goyal. "Are NLP models really able to solve simple math word problems?." arXiv preprint arXiv:2103.07191 (2021). [SVAMP]

[7] Cobbe, Karl, et al. "Training verifiers to solve math word problems." arXiv preprint arXiv:2110.14168 (2021). [GSM8K]

[8] Wei, Jason, et al. "Chain-of-thought prompting elicits reasoning in large language models." Advances in Neural Information Processing Systems 35 (2022): 24824-24837.[CoT [8]]

[9] Gao, Luyu, et al. "Pal: Program-aided language models." International Conference on Machine Learning. PMLR, 2023. [PAL]

[10] Chen, Wenhu, et al. "Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks." arXiv preprint arXiv:2211.12588 (2022). [PAL]

[11] Hendrycks, Dan, et al. "Measuring mathematical problem solving with the math dataset." arXiv preprint arXiv:2103.03874 (2021). (MATH)