

# Literature Review: Solving Math Word Problems via Cooperative Reasoning induced Language Models

Frederik Gerard de Vries | 369939 | rik.devries@epfl.ch  
REAL

## 1 Summary

The authors start off by stating the problem at hand (Zhu et al., 2023). The recent surge in the popularity of LLMs brings many new opportunities, such as using these models to solve math problems. However, the accuracy on such problems is still not good enough. The models seem to struggle especially when it comes to math word problems. These are math problems defined with words and usually told through a story. An example of such a problem from the paper is given in the appendix, section 5.

To improve upon these problems, the authors propose a new cooperative reasoning-induced framework, called CoRe. Taking inspiration from human biology, they mention how in our brain, different tasks are performed by different brain systems. System 1 is responsible for immediate actions, it comprises our human instinct. System 2 on the other hand can perform complicated reasoning. They project these two systems onto the world of LLMs in terms of a generator as an equivalent for system 1 and a verifier for system 2. For the system 1 functionality, the authors use a pre-trained large language model (PLM) and make it output its reasoning process and its final answer.

They then train two verifiers: one to score token-level steps and the other to score path-level steps. The path-level steps are scored by the final output of the generator: whether the predicted answer is the same as the ground truth. The token-level steps are scored by DeBERTa-large through the classification mechanism. Then, during inference, the authors treat the output of the LLM as a Monte Carlo Tree. They use a modified version of the Monte Carlo Tree Search (MCTS) algorithm to reach their final choice of answer.

The results of this technique speak for themselves.

The authors managed to achieve an increase of 9.6% over SOTA on similar benchmarks. The 540B version of PaLM with a zero-shot-CoT prompting approach and self-consistency achieves an 89% accuracy on the MultiArith benchmark. This new technique achieves a score of 97.5%. It also improves upon other methods on ASDiv-A, SingleOp, SingleEq, and the GSM8K benchmarks.

## 2 Strengths

The approach of the authors is good in many ways. The idea of modeling human biology is one of the foundational principles behind neural networks (Hardesty, 2017). Trying to model the functionality of the system 1 and system 2 parts of the brain therefore makes a lot of sense.

The way CoRe is designed, as a framework built on top of other LLMs and classifiers, makes it so that it can make use of future improvements in the underlying technology. In other words, as LLMs continue to develop and continue to get better, CoRe itself will get better as well. Another very large benefit of CoRe is that it rivals models that have far larger model sizes. As mentioned in the summary section, it has improved upon the 540B parameter PaLM model on the MultiArith benchmark. The authors themselves used a fine-tuned instance of GPT-J as their generator, another instance of GPT-J as their path-level verifier, and DeBERTa as their token-level verifier. GPT-J has a size of a bit over 6B parameters and DeBERTa consists of a little over 300 million parameters. The whole system, therefore, consists of 12.4B parameters. This is a tiny fraction of the amount PaLM has, and it achieves a higher score on these math-based benchmarks. This reduction in parameters allows the framework to run on GPUs with much less VRAM than these larger models.

### 3 Weaknesses

Besides all the benefits of CoRe, there are also a lot of problems with it and the paper presenting it. To start off, the idea of modeling system 1 and system 2 functionality in machine learning is great. However, there is no reason it has to be done through a combination of generators and verifiers. The argumentation the authors provide for this choice is that it makes a lot of intuitive sense. While this may be true, it is not a solid scientific argument for the choice.

The authors then go on to assume that PLMs can provide the system 1 functionality, and output a lot of possible options very quickly. However, this is not the case with the typical sizes of SOTA models, as inference may take a significant amount of time. This is likely the reason the authors chose GPT-J as their LLM here. This assumption makes the system incompatible with many newer, larger, and more performant LLMs. Also, while CoRe can indeed rival much larger LLMs on this specific problem set with a fraction of the parameters (thus needing less VRAM), it still requires a very long execution time and significant computing resources. In the paper, the authors discuss their hardware setup, which includes 8 Nvidia A100 GPUs. During inference, they let these GPUs run the MCTS algorithm for 320 seconds before selecting a final option. These high hardware and time requirements heavily limit the practical use of this algorithm for now.

All the critiques discussed so far have been related to the CoRe framework itself. However, the paper and its description also raise some questions. The description of the path-level verifier is two sentences long and lacks a lot of details. The authors just mention they use a BERT-like model with the classifier token. To replicate this system or fully understand what is going on, more details are necessary.

The final problem with the paper comes from the benchmarking of CoRe. As discussed in the summary, the authors employ a wide variety of benchmarks which is very good. However, for each benchmark, they choose different models or frameworks to compare themselves to. For ASDiv-A, SingleOp, SingleEq, and MultiArith they compare themselves to mostly very old fine-tuned methods

and to the base GPT-J zero-shot inference. Then for GSM8K they exclude zero-shot testing and instead replace it with few-shot testing without providing reasoning for this choice. Here they compare themselves to LaMDA, GPT-3 and PaLM. Some more consistency or at least an explanation on why these choices were made seems like a large shortfall of the paper.

### 4 Conclusion

In conclusion, the paper presents CoRe, a novel framework that is shown to have significant potential to improve upon currently existing methods in solving math word problems. Built on ideas coming from human biology, it can rival modern LLMs which have far larger parameter counts. The framework is also flexible and can be used with different underlying models. However, the current state of CoRe requires large compute and time allowances. The paper outlining CoRe is too concise at some points and is not fully consistent with respect to its benchmarking methods.

### References

- Larry Hardesty. 2017. [Explained: Neural networks](#).
- Xinyu Zhu, Junjie Wang, Lin Zhang, Yuxiang Zhang, Yongfeng Huang, Ruyi Gan, Jiaxing Zhang, and Yujie Yang. 2023. [Solving math word problems via cooperative reasoning induced language models](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4471–4485, Toronto, Canada. Association for Computational Linguistics.

### 5 Appendix

#### Example of a word math problem

##### Question

Sophia and Rose went together to the market to buy onions and potatoes. Rose bought 4 times the number of onions and potatoes Sophia bought. If Rose bought 12 onions and 4 potatoes, how many onions and potatoes in total did Sophia buy at the market?

##### CoRe solution

Since Rose bought 12 onions, this means there were  $12 / 4 = 3$  onions that Sophia bought. The number of potatoes that Sophia bought is 4, meaning there were  $4 / 4 = 1$  potato that Sophia

bought. The total number of onions and potatoes that Sophia bought is  $3 + 1 = 4$ . [ANS] 4.