266. CHAIN-OF-THOUGHT REASONING IN LANGUAGE MODELS

Nemykin N.B.

Belarusian State University of Informatics and Radioelectronics
Minsk, Republic of Belarus

Kaspiarovich N.G. – Senior Lecturer

This work looks at whether chain-of-thought prompting actually helps language models think through multi-step logic problems, or if it just makes their mistakes look more convincing. The key takeaway is that while walking through the reasoning step-by-step does improve accuracy, the models are still prone to subtle logic slips and the explanations they generate do not always reflect how they really got to the answer.

This paper looks into the use of Chain-of-Thought (CoT) prompting with language models when those models are asked to solve complicated logic problems that need many steps. The study goes over the normal CoT method and a change to it called Self-Consistency. The results from the PaLM-540B model suggest that using CoT makes the models more correct by 17–39 %, and Self-Consistency can improve things by as much as 17.9 %. Even so, there are problems, like how easily the method can be messed up if the problem is worded in a different way.

Modern language models are showing that they can do many things well. One thing that people are curious about is how well they can handle problems that need many steps. In many cases, just asking a question straight up does not work well for logic problems. A group of researchers at Google came up with Chain-of-Thought (CoT) in 2022. This method tells the models to explain how they got to the answer before giving the final answer. After that, Self-Consistency was created. With Self-Consistency, the models create many different ways of thinking about the problem and then pick the answer that shows up the most. This paper tries to see how well these methods work and what their limits are when it comes to helping models do better on logic problems.

The CoT method asks questions in a way that guides the model's thinking. Instead of directly asking What is 15+27?, the question is changed to What is 15+27? Let us think about it step by step. The first paper about CoT tested this method on the PaLM-540B model. The test showed that on the GSM8K dataset (which has simple math problems), CoT made the models get more correct answers, going from 17.9 % to 56.9 %. On the SVAMP dataset (which has problems that change a lot), scores went from 69.4 % to 79.0 %, and on the AQuA dataset (algebra word problems), scores went from 25.2 % to 35.8 %.

Still, counting on just one way of thinking might not be the best idea. Self-Consistency tries to fix this by making many different ways of figuring out the problem (using a temperature of 0.7) and then picking the answer that comes up most often. For example, on GSM8K, the PaLM-540B model got better, going from 56.5 % to 74.4 % (+17.9 %). SVAMP went from 79.0 % to 86.6 % (+7.6 %), and AQuA went from 35.8 % to 48.3 % (+12.5 %). It looks like making 40 different ways of thinking is the best number, since making more than that does not help much.

It is important to know that there are some problems with this method. First of all, CoT needs a model that is big enough. If the model has fewer than 100 billion parameters, it might not work as well. Second, CoT can be very sensitive to how the question is worded. Ye and Durrett pointed out that CoT can make things worse on tasks like ANLI and e-SNLI. Self-Consistency can be useful in these situations. On ANLI-R1, CoT was correct 68.8 % of the time, but Self-Consistency raised that to 78.5 %.

Third, CoT does not promise that the thinking will be correct. When researchers looked at answers from LaMDA-137B, they found that 46 % of the wrong answers had mostly correct thinking but small mistakes (like math errors or missing steps). The other 54 % had mistakes in how the model understood the problem.

The chain-of-thought method may be a good way to handle logic problems that have many steps. But things have to be just right for it to work best. You need a big model and problems that are not too hard. Self-Consistency improves the method by looking at many different ways of thinking. But the method has its limits. It takes a lot of computer power, it can be changed by small changes in the wording, and it does not make sure that every step of the thinking is correct. In the future, people might try to make things better by finding the best ways of thinking and helping the model understand the problem better.

References:

1. Rae J. W., Borgeaud S., Cai T. [et al.] Scaling Language Models: Methods, Analysis & Insights from Training Gopher // arXiv preprint arXiv:2112.11446. – 2021. – URL: https://arxiv.org/abs/2112.11446 (date of access: 16.03.2026).

2. Wei J., Wang X., Schuurmans D. [et al.] Chain-of-Thought Prompting Elicits Reasoning in Large Language Models // Conference on Neural Information Processing Systems (NeurIPS). – 2022. – P. 1–43. – URL: https://proceedings.neurips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html (date of access: 16.03.2026).

3. Wang X., Wei J., Schuurmans D. [et al.] Self-Consistency Improves Chain of Thought Reasoning in Language Models // arXiv preprint arXiv:2203.11171. – 2022. – P. 1–26. – URL: https://arxiv.org/abs/2203.11171 (date of access: 16.03.2026).