Efficient Tool Use with Chain-of-Abstraction Reasoning

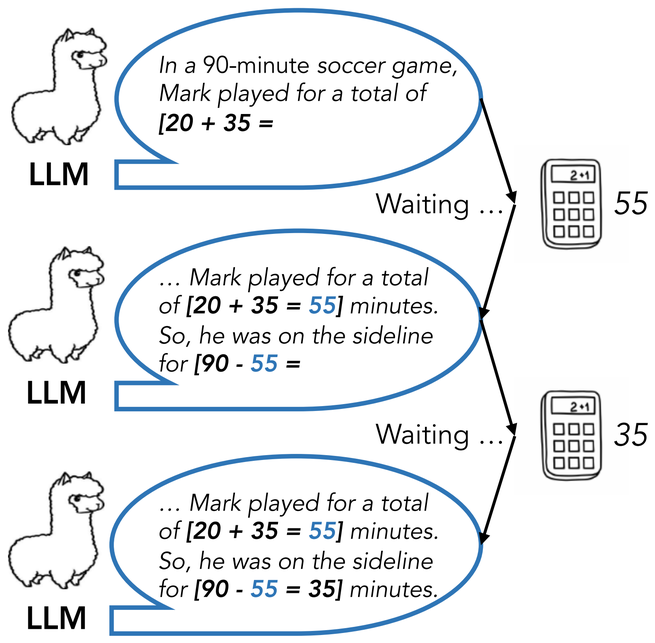
Silin Gao, Jane Dwivedi-Yu, Ping Yu, Xiaoqing Ellen Tan, Ramakanth Pasunuru, Olga Golovneva, Koustuv Sinha, Asli Celikyilmaz, Antoine Bosselut*, Tianlu Wang*

*Equal Supervision

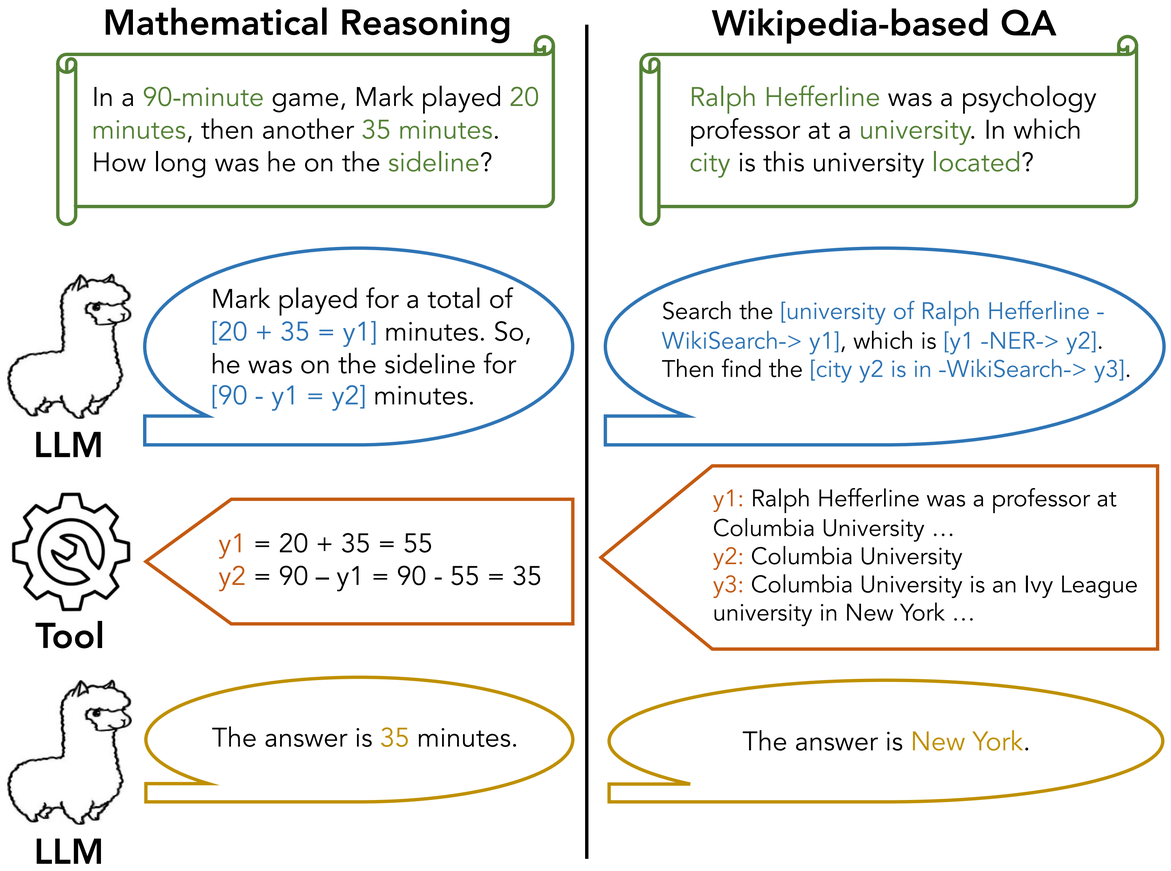
LLM agents use tools to mitigate errors in **multi-step** reasoning, however:

- **lack of coordination** among multiple tool calls
- **need to wait** for tool responses

How to achieve better **efficiency** and holistic **planning** of tool use?



Chain-of-Abstraction (CoA) Reasoning



LLMs plan reasoning chains with **abstract placeholders**:

- inter-connect multiple tool calls
- robust to variation of knowledge
- more **feasible** and **general** strategies

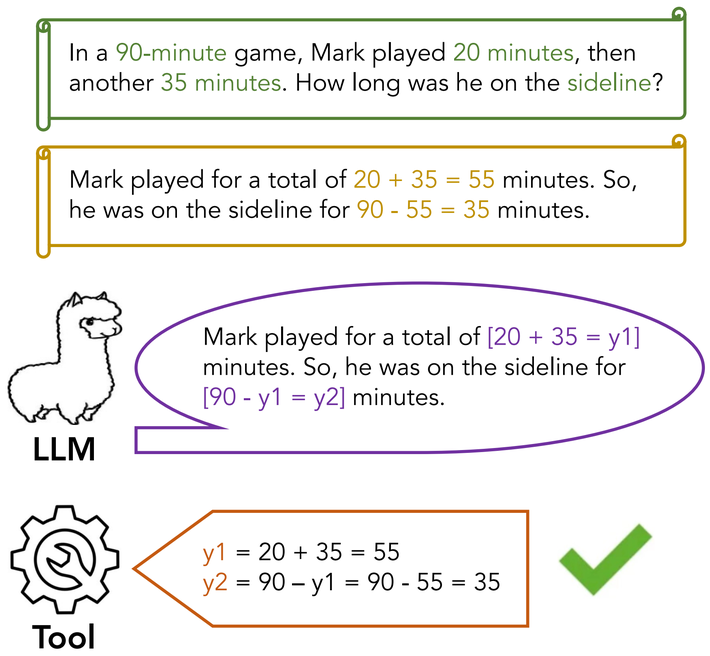
→ Domain tools **infill knowledge once** to the whole CoA:

- CoA decoding for next question can start while tools infill current CoA
- more **efficient** multiple inferences

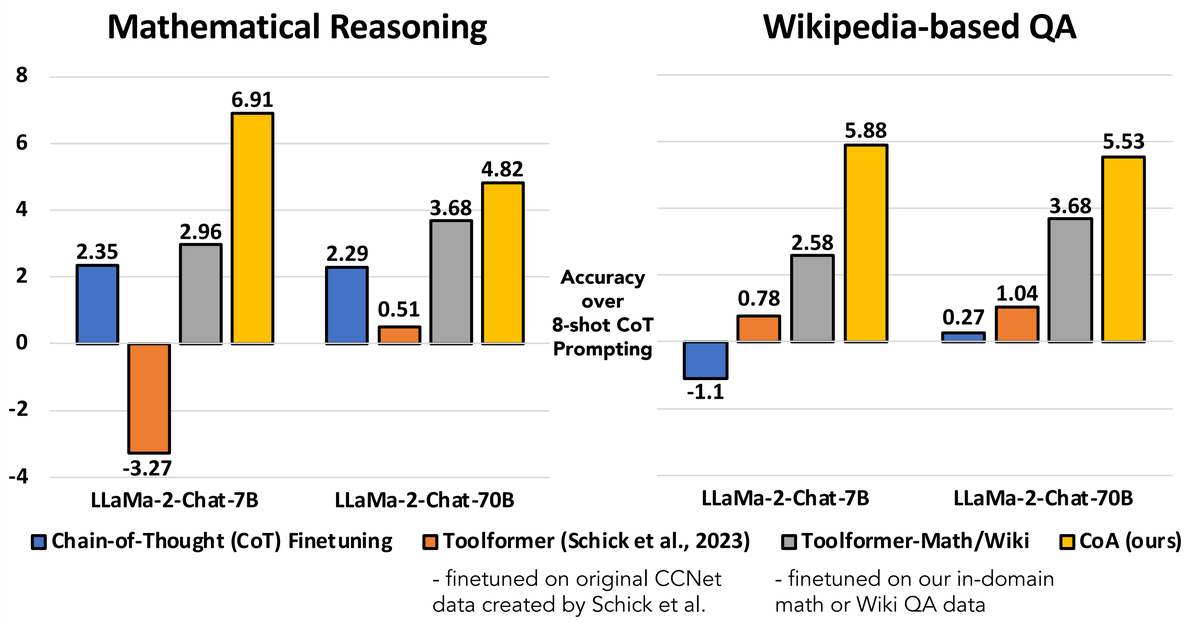
CoA Finetuning Data Construction

Simple method to build finetuning data for LLM agents to learn CoA:

- prompt LLMs to **re-write** existing gold answers to abstract chains
- domain tools **verify** the correctness of answer re-writing



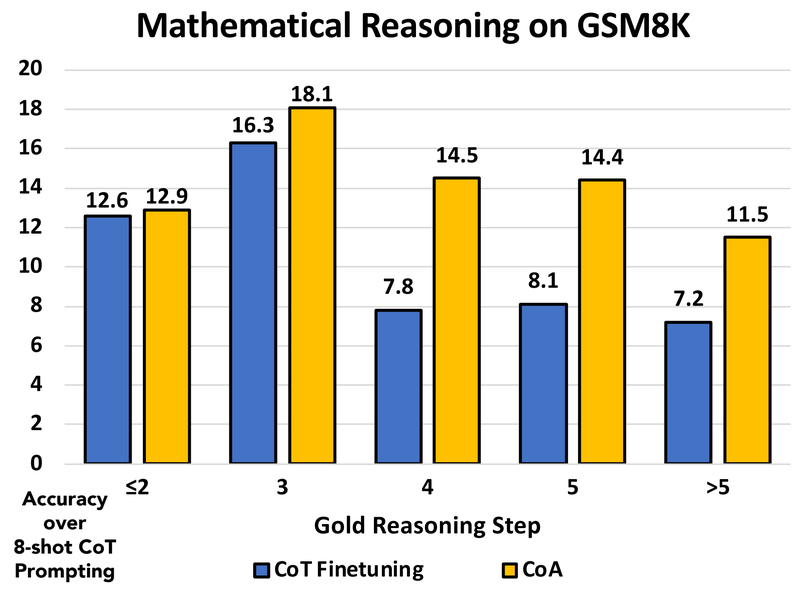
CoA improves Accuracy of reasoning



CoA outperforms CoT and Toolformer on **both in-distribution and OOD** test sets, average **~6%** absolute accuracy improvement over 8-shot CoT prompting.

Why?

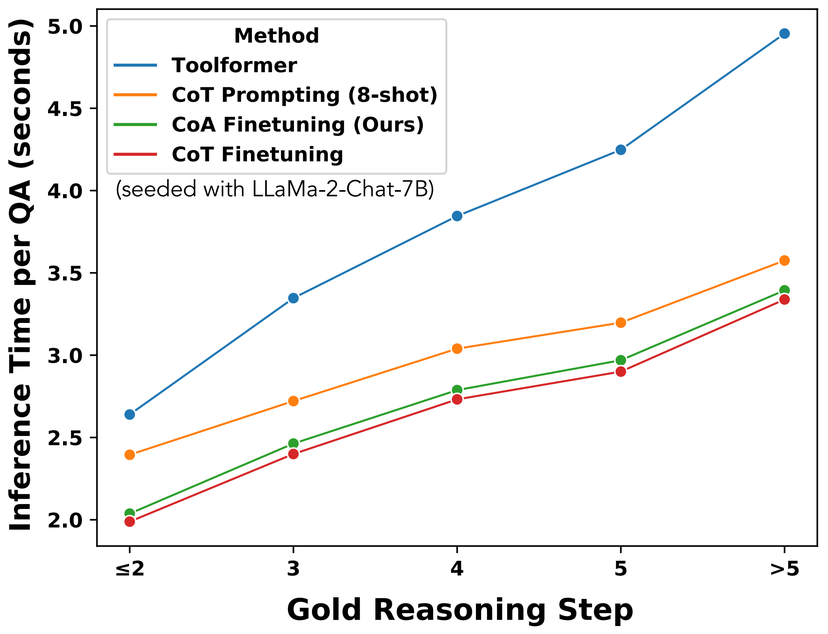
Model with CoA has **more robust** capability in **long chain** of reasoning, which benefits from **planning** abstract placeholders.



CoA improves reasoning Efficiency

CoA decoding does **not** need to **wait** for external tool responses.

Performance benefits do **not** come with **increased** computational costs.



Accuracy with Different Decoding Schemes

CoA **generalizes well** to more advanced decoding schemes.