

10. Scaling Language Models

Guest lecture by [Jason Wei](#)

Scaling language models

Scaling measures: model size (# parameters), training data (# tokens), training compute (FLOPs)

Why is scaling challenging?

- Compute is expensive
- Model parallelism & hardware challenges
- Loss divergences & hardware failures
- Other reasons: inductive bias, different from human learning, incentive mismatch

Scaling laws: Language modeling performance improves smoothly as we scale up the model

What can't language models learn from next-word prediction?

Current world knowledge, arbitrarily long arithmetic, many-step reasoning

Emergent abilities

Emergent abilities are abilities that are not present in small models, but are present in large models.

There are a lot of emergent abilities.

There are a lot of unknown unknowns. We don't know the full range of emergent abilities.

One model, any task.

Prompt engineering

Multi-step reasoning is hard for language models.

[Chain-of-Thought Prompting Elicits Reasoning in Large Language Models.](#)

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

Self-Consistency Improves Chain of Thought Reasoning in Language Models: a new decoding strategy, self-consistency, to replace the naive greedy decoding used in chain-of-thought prompting. It first samples a diverse set of reasoning paths instead of only taking the greedy one, and then selects the most consistent answer by marginalizing out the sampled reasoning paths. Self-consistency leverages the intuition that a complex reasoning problem typically admits multiple different ways of thinking leading to its unique correct answer.