

Teach Language Models to Reason



Denny Zhou
Google DeepMind

December, 2023

What do you expect from AI?

- Self-driving cars
- Digital assistant
- Solving hardest math problems
- ...



As a machine learning researcher, my dream is to achieve

Accuracy 100%

by **learning from only a few examples**, like what humans usually do

Does machine learning meet this expectation?

Semi-supervised learning

Manifold learning

Sparsity and low rank

Active learning

Transfer learning

Metalearning

Bayesian nonparametric

Kernel machines

...



What is missing in machine learning?

Reasoning

Humans can learn from only a few examples because humans can reason

We have found a simple way to solve reasoning:

**Teach language models to
reason, like teaching kids**

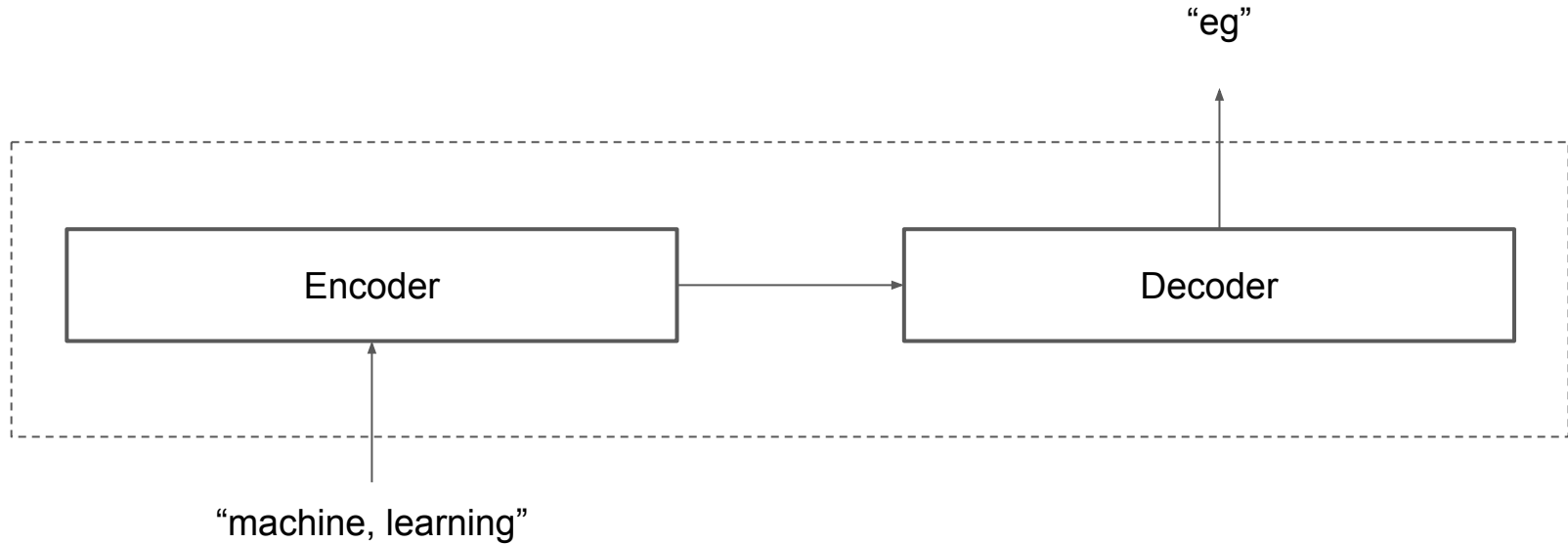
Let's start from a toy problem

Toy problem: last-letter-concatenation

Input	Output
“Elon Musk”	“nk”
“Bill Gates”	“ls”
“Barack Obama”	?

Rule: Take the last letter of each word, and then concatenate them

Solve it by machine learning? Tons of labels needed

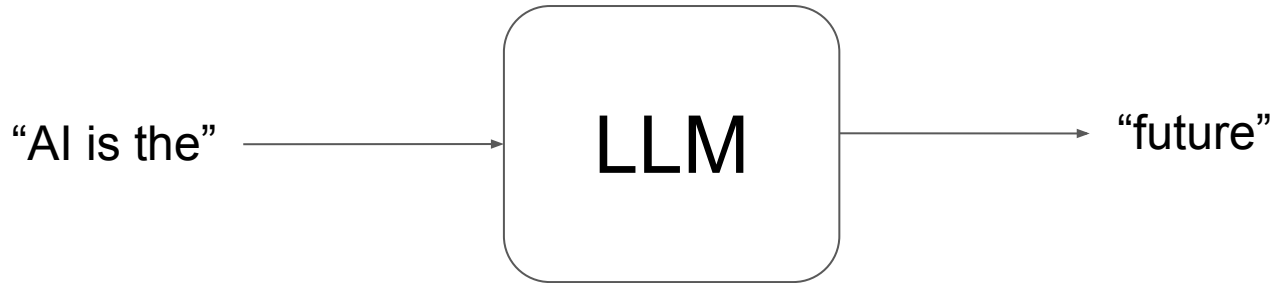


Would you like to call an ML method which needs tons of labels to learn a "trivial" task as AI?

How to solve this problem
with LLMs?

What are **L**arge **L**anguage **M**odels (LLMs)?

LLM is a “transformer” model trained to predict the next word

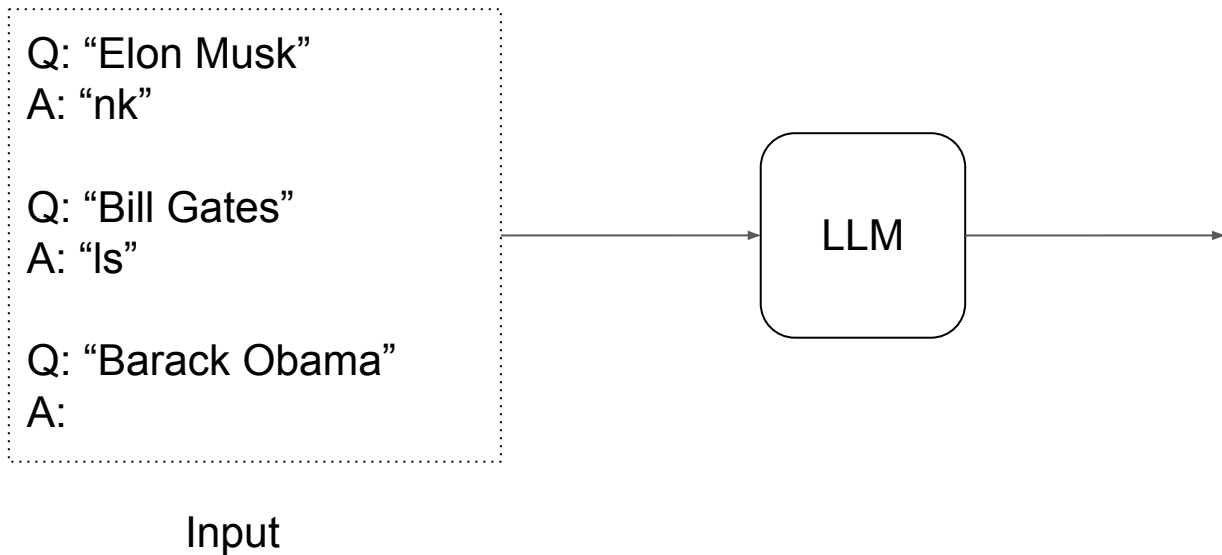


Trained with many sentences, e.g. all texts from the Internet

You can think of training an LLM as training a parrot to mimic human languages



Few-shot prompting for last-letter-concatenation



Playground

Load a preset... 

Save

View code

Share



Q: "Elon Musk"

A: "nk"

Q: "Bill Gates"

A: "Is"

Q: "Barack Obama"

A: "ma"



 Complete 

Model

text-davinci-003 

Temperature 0



Maximum length 256



Stop sequences

Enter sequence and press Tab

Top P 1



Frequency penalty 0



Presence penalty 0



FAILED

Submit



54

Load a preset... 

Save

View code

Share



Q: "Elon Musk"

A: "nk"

Q: "Bill Gates"

A: "Is"

Q: "Barack Obama"

A: "ma"



 Complete 

Model

text-davinci-003 

Temperature 0



Maximum length 256



Stop sequences

Enter sequence and press Tab

Top P 1



Frequency penalty 0



Presence penalty 0



How about adding more examples?

Submit



54

Playground

Load a preset... ▾

Save

View code

Share



Q: "Elon Musk"

A: "nk"

Q: "Bill Gates"

A: "ls"

Q: "Steve Jobs"

A: "es"

Q: "Larry Page"

A: "ye"

Q: "Jeff Bezos"

A: "fs"

Q: "Barack Obama"

A: "ma"



☰ Complete ▾

Model

text-davinci-003 ▾

Temperature 0



Maximum length 256



Stop sequences

Enter sequence and press Tab

Top P 1



Frequency penalty 0



Presence penalty 0



FAILED

Submit



94

Why we created the last-letter-concatenation task?

- Machine learning fail
- Few-shot prompting fail
- Trivial for humans

Chain-of-Thought (CoT) Prompting

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou.
[Chain-of-thought prompting elicits reasoning in large language models](#). NeurIPS 2022.

CoT: Adding “thought” before “answer”

Q: “Elon Musk”

A: the last letter of "Elon" is "n". the last letter of "Musk" is "k". Concatenating "n", "k" leads to "nk". so the output is "nk".

thought

Q: “Bill Gates”

A: the last letter of "Bill" is "l". the last letter of "Gates" is "s". Concatenating "l", "s" leads to "ls". so the output is "ls".

Q: “Barack Obama”

A:

CoT: Adding “thought” before “answer”

Q: “Elon Musk”

A: the last letter of "Elon" is "n". the last letter of "Musk" is "k". Concatenating "n", "k" leads to "nk". so the output is "nk".

thought

Q: “Bill Gates”

A: the last letter of "Bill" is "l". the last letter of "Gates" is "s". Concatenating "l", "s" leads to "ls". so the output is "ls".

Q: “Barack Obama”

A: the last letter of "Barack" is "k". the last letter of "Obama" is "a". Concatenating "k", "a" leads to "ka". so the output is "ka".

One demonstration is enough, as humans

Q: "Elon Musk"

A: the last letter of "Elon" is "n". the last letter of "Musk" is "k". Concatenating "n", "k" leads to "nk". so the output is "nk".

Q: "Barack Obama"

A: the last letter of "Barack" is "k". the last letter of "Obama" is "a". Concatenating "k", "a" leads to "ka". so the output is "ka".

Brown et al. Language Models are Few-Shot Learners. May, **2020**

Standard few-shot prompting

<input, output>

Wei et al. Chain-of-thought prompting elicits reasoning in large language models. Jan, **2022**

Chain-of-thought prompting

<input, **thought**, output>

Chain-of-Thought Prompting

Google I/O 2022

PaLM: Scaling Language Modeling with Pathways

Ananksha Chowdhery¹ Sharan Narang¹ Jacob Devlin¹
Maarten Bosma¹ Gaurav Mishra¹ Adam Roberts¹ Paul Barham¹
Hyung Won Chung¹ Charles Sutton¹ Sebastian Gehrmann¹ Parker Schuh¹ Kensen Shi¹
Sasha Tsvyashchenko¹ Joshua Maynez¹ Abhishek Rao¹ Parker Barnes¹ Yi Tay¹
Noam Shazeer¹ Vinodkumar Prabhakarar¹ Emily Reif¹ Nian Du¹ Ben Hutchinson¹
Reiner Pope¹ James Bradbury¹ Jacob Austin¹ Michael Isard¹ Guy Gur-Ari¹
Pengcheng Yin¹ Teja Dike¹ Anselm Levskaya¹ Sanjay Ghemawat¹ Sunipa Dev¹
Henryk Michalewski¹ Xavier Garcia¹ Vedant Misra¹ Kevin Robinson¹ Liam Fedus¹
Denny Zhou¹ Daphne Ippolito¹ David Luan¹ Hyeontaek Lim¹ Barret Zoph¹
Alexander Spitsidov¹ Ryan Sepasat¹ David Duhan¹ Shivali Agrawal¹ Mark Duerick¹
Andrew M. Dai¹ Zhanmuyalaya Sankaranarayanan Pillai¹ Marie Pellat¹ Aitor Lewkowycz¹
Erica Moreira¹ Reyon Child¹ Oksana Polozov¹ Katherine Lee¹ Zongwei Zhou¹
Xuwu Wang¹ Benjamin Sotia¹ Mark Diaz¹ Oskan Firat¹ Michele Catasta¹ Jason Wei¹
Kathy Meier-Hellstern¹ Douglas Eck¹ Jeff Dean¹ Slav Petrov¹ Noah Fiedel¹

Google Research

Abstract

Large language models have been shown to achieve remarkable performance across a variety of natural language tasks using *few-shot* learning, which drastically reduces the number of task-specific training examples needed to adapt the model to a particular application. To further our understanding of the

Can LLMs solve math word problems?

Elsa has 3 apples. Anna has 2 more apples than Elsa. How many apples do they have together?

They have 5 apples together.

FAILED

☰ Complete ▾

Model

text-davinci-003 ▾

Temperature 1



Maximum length 256



Stop sequences

Enter sequence and press Tab

Let's see how CoT magically works

Demonstration example
with “thought”

There were 3 pizzas in total at the pizza shop. A customer bought one pizza. How many pizzas are left?

There are 3 pizzas in total and a customer bought one. So there are $3 - 1 = 2$ pizzas left. So the answer is 2.

Elsa has 3 apples. Anna has 2 more apples than Elsa. How many apples do they have together?

Elsa has 3 apples and Anna has 2 more apples than Elsa. So Anna has $3 + 2 = 5$ apples.

So Elsa and Anna have $3 + 5 = 8$ apples together. So the answer is 8.

CoT for any task

StrategyQA

Q: Yes or no: Would a pear sink in water?

A: The density of a pear is about 0.6 g/cm^3 , which is less than water. Thus, a pear would float. So the answer is no.

Date Understanding

Q: The concert was scheduled to be on 06/01/1943, but was delayed by one day to today. What is the date 10 days ago in MM/DD/YYYY?

A: One day after 06/01/1943 is 06/02/1943, so today is 06/02/1943. 10 days before today is 05/23/1943. So the answer is 05/23/1943.

Sports Understanding

Q: Is the following sentence plausible? "Joao Moutinho caught the screen pass in the NFC championship."

A: Joao Moutinho is a soccer player. The NFC championship is part of American football, not soccer. So the answer is no.

SayCan (Instructing a robot)

Human: How would you bring me something that isn't a fruit?

Explanation: the user wants something to eat that isn't a fruit. An energy bar is not a fruit, so I will bring the user an energy bar.

Plan: 1. find(energy bar) 2. pick(energy bar) 3. find(user) 4. put(energy bar) 5. done().

Last Letter Concatenation

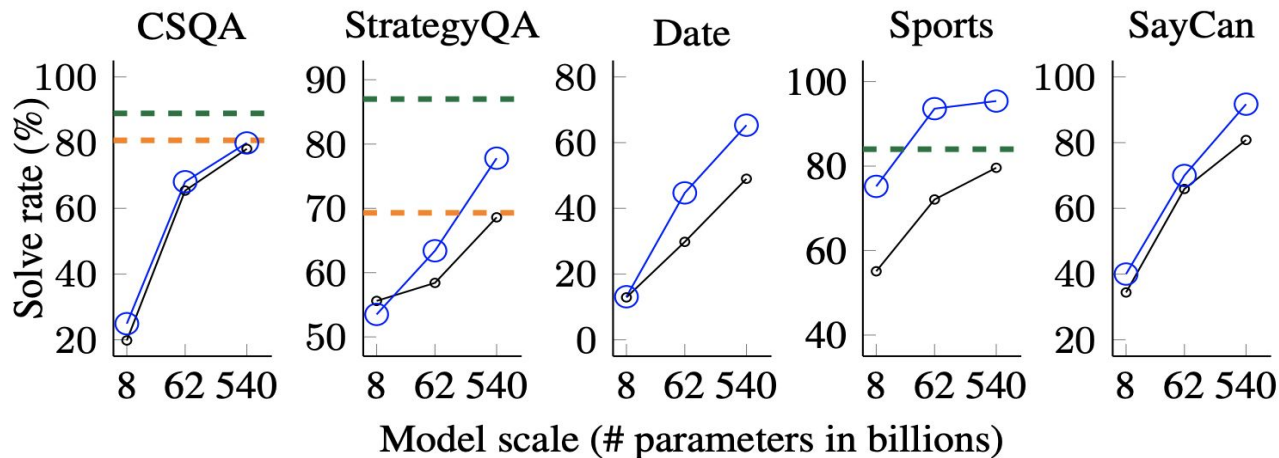
Q: Take the last letters of the words in "Lady Gaga" and concatenate them.

A: The last letter of "Lady" is "y". The last letter of "Gaga" is "a". Concatenating them is "ya". So the answer is ya.

Coin Flip (state tracking)

Q: A coin is heads up. Maybelle flips the coin. Shalonda does not flip the coin. Is the coin still heads up?

A: The coin was flipped by Maybelle. So the coin was flipped 1 time, which is an odd number. The coin started heads up, so after an odd number of flips, it will be tails up. So the answer is no.



- - - Prior supervised best —○ Standard prompting
 - - - Human —○ Chain of thought

100x - 1000x data efficient than supervised sota in the literature

CoT Stands on Giants (I)

Ling et al 2017 has proposed using **natural language rationale** to solve math word problems: “**derive the final answer through a series of small steps**”.
Trained a sequence-to-sequence model from scratch.

Problem 1:

Question: Two trains running in opposite directions cross a man standing on the platform in 27 seconds and 17 seconds respectively and they cross each other in 23 seconds. The ratio of their speeds is:

Options: A) $3/7$ B) $3/2$ C) $3/88$ D) $3/8$ E) $2/2$

Rationale: Let the speeds of the two trains be x m/sec and y m/sec respectively. Then, length of the first train = $27x$ meters, and length of the second train = $17y$ meters. $(27x + 17y) / (x + y) = 23 \rightarrow 27x + 17y = 23x + 23y \rightarrow 4x = 6y \rightarrow x/y = 3/2$.

Correct Option: B



CoT Stands on Giants (II)

Following the work by Ling et al 2017, Cobbe et al 2021 in OpenAI built a much larger math word problem dataset (**GSM8K**) with natural language rationales, and using the dataset to finetuned GPT3

Problem: Ali is a dean of a private school where he teaches one class. John is also a dean of a public school. John has two classes in his school. Each class has $\frac{1}{8}$ the capacity of Ali's class which has the capacity of 120 students. What is the combined capacity of both schools?

Solution: Ali's class has a capacity of 120 students. Each of John's classes has a capacity of $120/8 = 15$ students. The total capacity of John's two classes is $15 \text{ students} * 2 \text{ classes} = 30$ students. The combined capacity of the two schools is $120 \text{ students} + 30 \text{ students} = 150$ students.

Final answer: 150



Cobbe et al. Training Verifiers to Solve Math Word Problems. [arXiv:2110.14168](https://arxiv.org/abs/2110.14168) [cs.LG]. October 2021.

CoT Stands on Giants (III)

Nye et al 2021 propose **Scratchpad**: predicting the final output of a program by predicting its intermediate execution result from line to line

Input:

```
2 9 + 5 7
```

Target:

```
<scratch>
```

```
2 9 + 5 7 , C: 0
```

```
2 + 5 , 6 C: 1 # added 9 + 7 = 6 carry 1
```

```
, 8 6 C: 0 # added 2 + 5 + 1 = 8 carry 0
```

```
0 8 6
```

```
</scratch>
```

```
8 6
```



Nye et al. Show Your Work: Scratchpads for Intermediate Computation with Language Models. [arXiv:2112.00114](https://arxiv.org/abs/2112.00114) [cs.LG], 2021

Self-Consistency

Greatly improves chain-of-thought prompting

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, Denny Zhou.

[Self-Consistency Improves Chain of Thought Reasoning in Language Models](#). ICLR 2023.

Prompt with example chains of thought

Q: Shawn has five toys. He gets two more each from his mom and dad. How many toys does he have now?

A: Shawn started with 5 toys. 2 toys each from his mom and dad is 4 more toys. The final answer is $5+4=9$. The answer is 9.

Q: Janet's ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder for \$2 per egg. How much does she make every day?

A:

Language model

Sample decode with diverse reasoning paths

She has $16 - 3 - 4 = 9$ eggs left. So she makes $\$2 * 9 = \18 per day. **The answer is \$18.**

This means she uses $3 + 4 = 7$ eggs every day. So in total she sells $7 * \$2 = \14 per day. **The answer is \$14.**

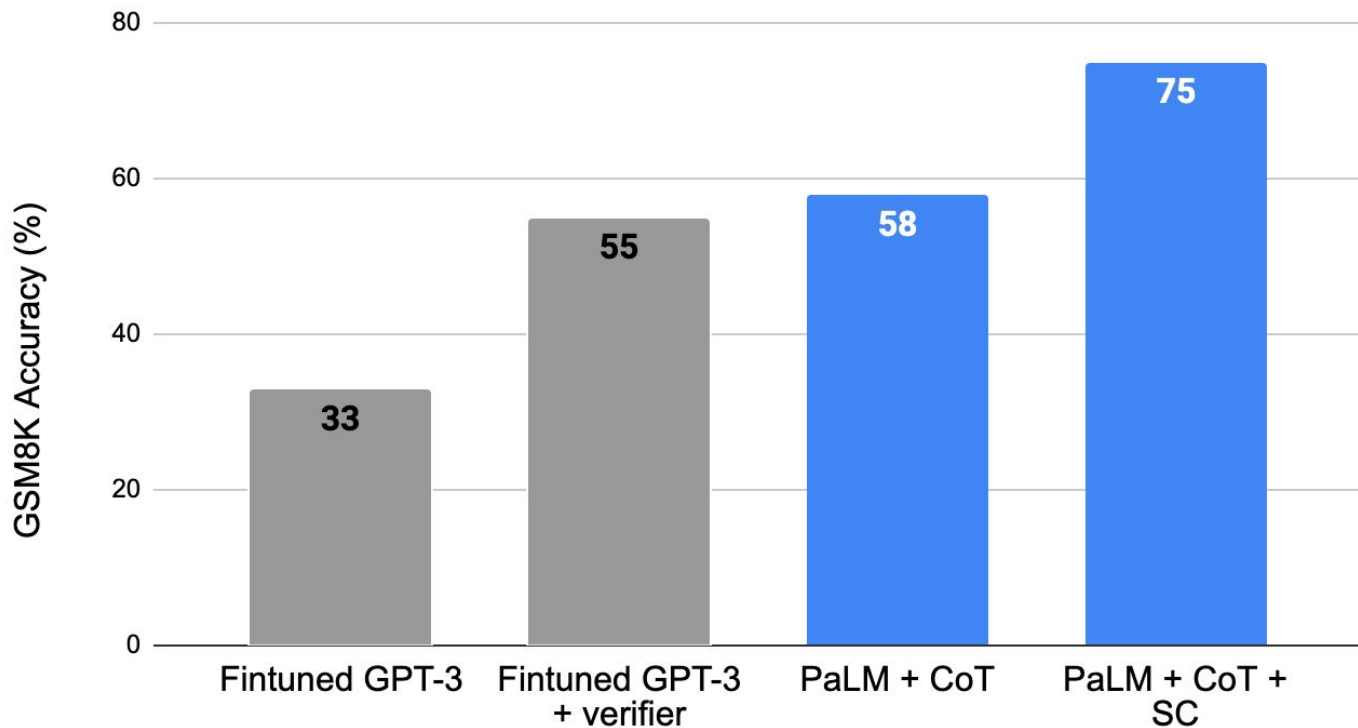
She eats 3 for breakfast, so she has $16 - 3 = 13$ left. Then she bakes muffins, so she has $13 - 4 = 9$ eggs left. So she has $9 * \$2 = \18 . **The answer is \$18.**

Majority vote

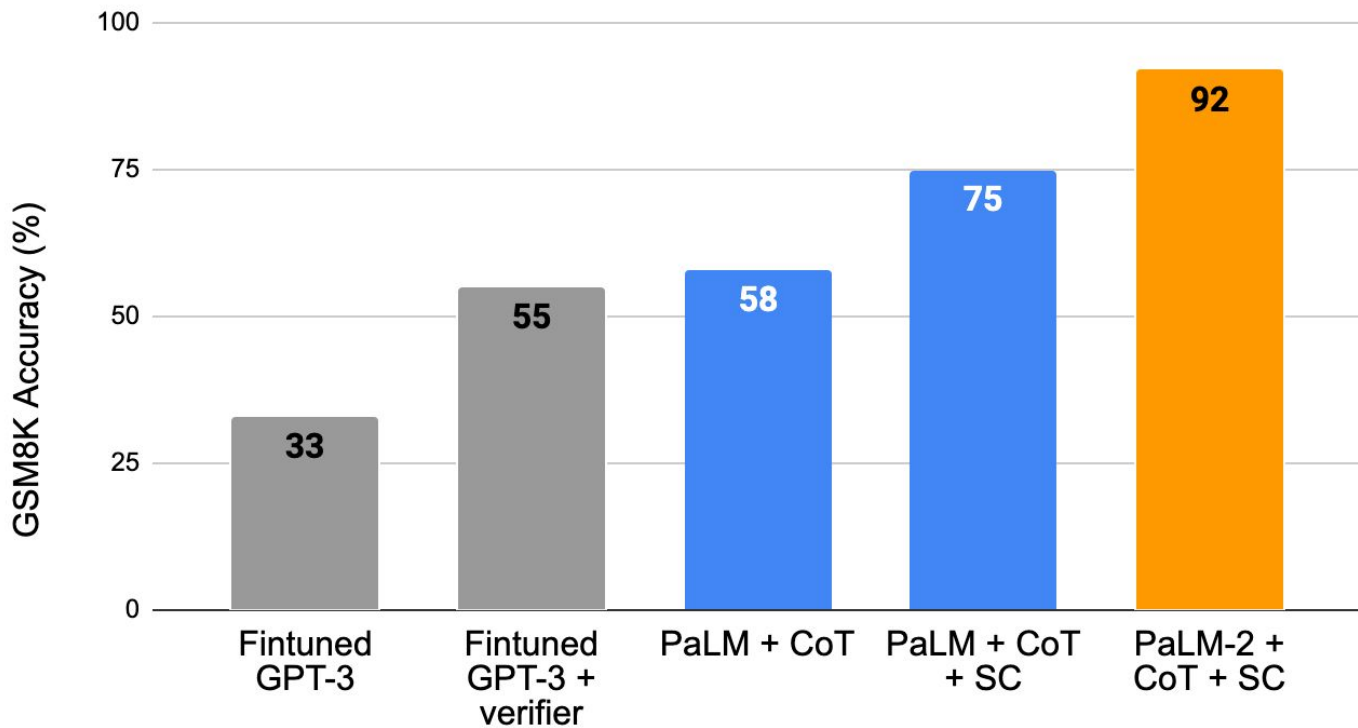
The answer is \$18.

Figure 1: The self-consistency method contains three steps: (1) prompt a language model using example chains of thought; (2) sample from the language model's decoder to generate a diverse set of reasoning paths; and (3) choose the most consistent answer using the majority/plurality vote.

Crushed GSM8K SOTA with only 8 examples



Crushed GSM8K SOTA with only 8 examples



How many more examples are needed for finetuning to be comparable to CoT + SC?

“We perform finetuning... GSM8K ... it appears likely that the 175B model would require **at least two additional orders of magnitude** of training data to reach an **80% solve rate**.”

Cobbe et al. Training Verifiers to Solve Math Word Problems. arXiv:2110.14168 [cs.LG], 2021.

Solve high school math problems

- Finetuning PaLM with math data
- SC + CoT solves **50%**!
- Non-math graduate students solve: 40%

Lewkowycz et al., 2022. Solving Quantitative Reasoning Problems With Language Models.

Question: A particle moves so that it is at $(3 \sin \frac{t}{4}, 3 \cos \frac{t}{4})$ at time t . Find the speed of the particle, measured in unit of distance per unit of time.

Target: The parametric curve $(x, y) = (3 \sin \frac{t}{4}, 3 \cos \frac{t}{4})$ describes a circle with radius 3. Furthermore, it makes a full revolution at time $t = 8\pi$. (figure) Therefore, the speed of the particle is $\frac{6\pi}{8\pi} = \boxed{\frac{3}{4}}$.

Model output: The speed is the magnitude of the velocity vector. Therefore, we must first find the velocity vector, which is the derivative of the position vector. We have

$$\mathbf{v} = \frac{d}{dt} \left(3 \sin \frac{t}{4}, 3 \cos \frac{t}{4} \right) = 3 \left(\cos \frac{t}{4}, -\sin \frac{t}{4} \right) \frac{1}{4} = \left(\frac{3}{4} \cos \frac{t}{4}, -\frac{3}{4} \sin \frac{t}{4} \right).$$

Therefore, the magnitude of the velocity vector is

$$\begin{aligned} |\mathbf{v}| &= \sqrt{\left(\frac{3}{4} \cos \frac{t}{4} \right)^2 + \left(-\frac{3}{4} \sin \frac{t}{4} \right)^2} = \sqrt{\frac{9}{16} \cos^2 \frac{t}{4} + \frac{9}{16} \sin^2 \frac{t}{4}} \\ &= \frac{3}{4} \sqrt{\cos^2 \frac{t}{4} + \sin^2 \frac{t}{4}} = \frac{3}{4}. \end{aligned}$$

Therefore, the speed of the particle is $\boxed{\frac{3}{4}}$.

	Method	AddSub	MultiArith	ASDiv	AQuA	SVAMP	GSM8K
	Previous SoTA	94.9^a	60.5 ^a	75.3 ^b	37.9 ^c	57.4 ^d	35 ^e / 55 ^g
UL2-20B	CoT-prompting	18.2	10.7	16.9	23.6	12.6	4.1
	Self-consistency	24.8 (+6.6)	15.0 (+4.3)	21.5 (+4.6)	26.9 (+3.3)	19.4 (+6.8)	7.3 (+3.2)
LaMDA-137B	CoT-prompting	52.9	51.8	49.0	17.7	38.9	17.1
	Self-consistency	63.5 (+10.6)	75.7 (+23.9)	58.2 (+9.2)	26.8 (+9.1)	53.3 (+14.4)	27.7 (+10.6)
PaLM-540B	CoT-prompting	91.9	94.7	74.0	35.8	79.0	56.5
	Self-consistency	93.7 (+1.8)	99.3 (+4.6)	81.9 (+7.9)	48.3 (+12.5)	86.6 (+7.6)	74.4 (+17.9)
GPT-3 Code-davinci-001	CoT-prompting	57.2	59.5	52.7	18.9	39.8	14.6
	Self-consistency	67.8 (+10.6)	82.7 (+23.2)	61.9 (+9.2)	25.6 (+6.7)	54.5 (+14.7)	23.4 (+8.8)
GPT-3 Code-davinci-002	CoT-prompting	89.4	96.2	80.1	39.8	75.8	60.1
	Self-consistency	91.6 (+2.2)	100.0 (+3.8)	87.8 (+7.6)	52.0 (+12.2)	86.8 (+11.0)	78.0 (+17.9)

“Self-consistency + chain-of-thought” crushed SOTA by large margin

Why does self-consistency work?

Marginalization!

$\arg \max \mathbb{P}(\text{answer}|\text{problem})$ (find the answer with the maximum probability)

$$= \sum_{\text{rationale}} \mathbb{P}(\text{answer, rationale}|\text{problem}) \quad (\text{sum over all latent reasoning paths})$$

$$\approx \frac{\text{frequency of the answer}}{\text{total number of sampled responses}} \quad (\text{approximate the sum by sampling})$$

$$\propto \text{frequency of the answer} \quad (\text{ignore the common denominator})$$

Thus, $\arg \max \mathbb{P}(\text{answer}|\text{problem}) \approx$ selecting the most frequent answer.

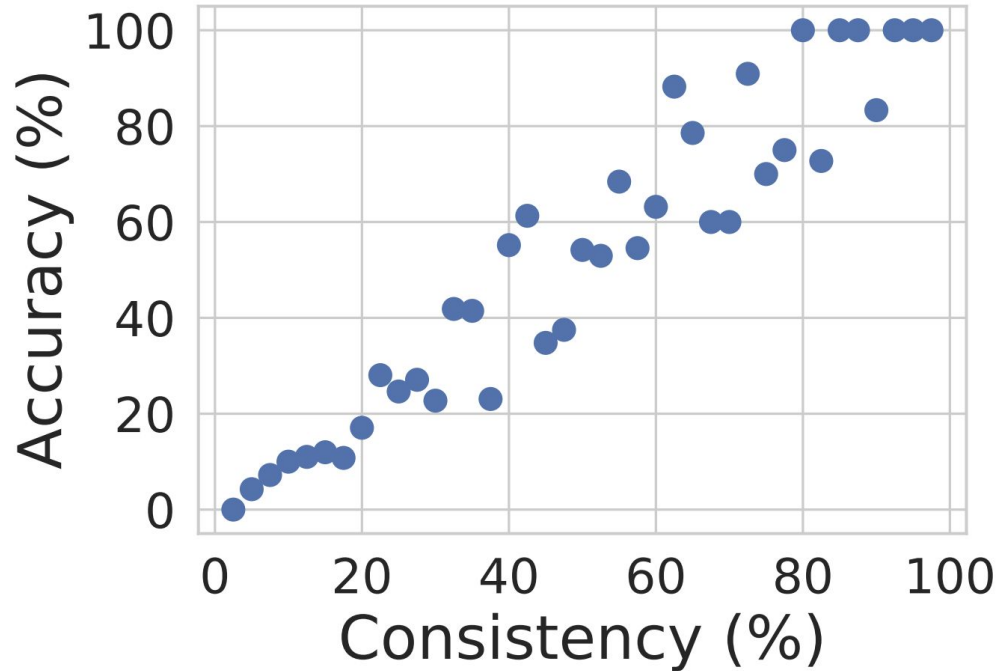
Self-consistency is the empirical implementation of marginalization. Don't be superficial to interpret it as majority voting!



When there is no reasoning path, we don't need self-consistency, since we can directly choose the most likely answer using $P(Y|X)$!

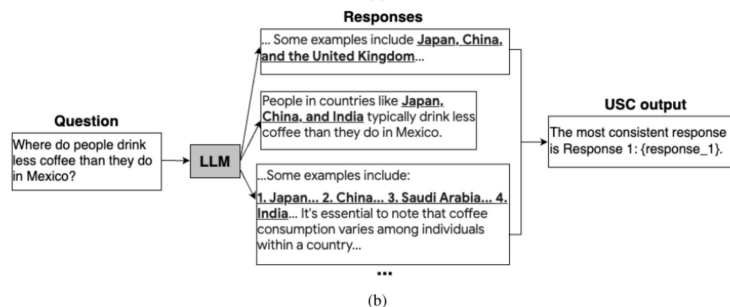
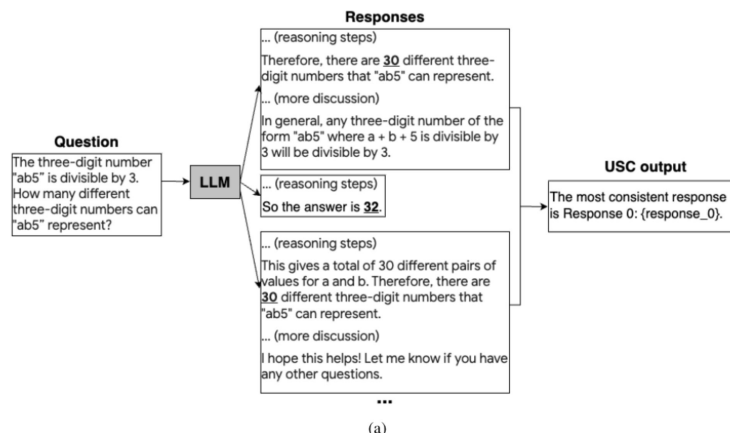


More consistent, more likely to be correct!



Universal Self-Consistency (USC) for Any Task

Just ask LLMs to select the most consistent response based on majority consensus!



Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, Xuezhi Wang, Denny Zhou. [Universal Self-Consistency for Large Language Model Generation](#). arXiv:2311.17311 [cs.CL], 2023.

Universal Self-Consistency (USC) for Any Task

- USC consistently improves the performance on free-form generation tasks, like summarization, where SC is inapplicable
- For code generation, USC matches the performance of execution-based SC that selects the code with the most consistent execution outputs, while USC does not require code execution
- For math reasoning, USC matches the performance of SC while eliminating the need of parsing answers to aggregate.

Least-to-Most Prompting

Enable easy-to-hard generalization by decomposition

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, Ed Chi. [Least-to-Most Prompting Enables Complex Reasoning in Large Language Models](#). ICLR 2023.

CoT fails to generalize to harder problems

Last-letter-concatenation task with list longer than demonstration which is limited to 3 words:

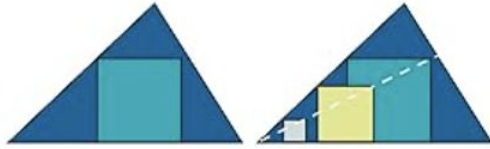
Method	L = 4	L = 6	L = 8	L = 10	L = 12
Standard prompting	0.0	0.0	0.0	0.0	0.0
Chain-of-Thought	89.4	75.0	51.8	39.8	33.6

CoT's performance greatly drops as length increases

Key Idea of Least-to-Most Prompting

1. **Decompose** a complex problem into a list of easier subproblems
2. Sequentially **solve these subproblems** (from least to most complex)

Least-to-Most Prompting = Planning/Sketch + Reasoning



How to Solve It

a new aspect of
mathematical method

*With a new foreword
by John H. Conway*

G. POLYA

Decomposing and recombining are important operations of the mind.

differently. You decompose the whole into its parts, and you recombine the parts into a more or less different whole.

1. If you go into detail you may lose yourself in details. Too many or too minute particulars are a burden on the mind. They may prevent you from giving sufficient attention to the main point, or even from seeing the main point at all. Think of the man who cannot see the forest for the trees.

Solve math problems by decomposition

Q: Elsa has 3 apples. Anna has 2 more apples than Elsa. How many apples do they have together?

A: Let's break down this problem: 1. How many apples does Anna have? 2. How many apples do Elsa and Anna have together?

1. Anna has 2 more apples than Elsa. So Anna has $2 + 3 = 5$ apples.

2. Anna has 5 apples. Elsa and Anna have $3 + 5 = 8$ apples together.

The answer is 8.

Q: Eliza's rate per hour for the first 40 hours she works each week is \$10. She also receives an overtime pay of 1.2 times her regular hourly rate. If Eliza worked for 45 hours this week, how much are her earnings for this week?

A:

Solve math problems by decomposition (Cont')

...

Q: Eliza's rate per hour for the first 40 hours she works each week is \$10. She also receives an overtime pay of 1.2 times her regular hourly rate. If Eliza worked for 45 hours this week, how much are her earnings for this week?

A: Let's break down this problem: 1. How much does Eliza earn for the first 40 hours she works each week? 2. How much does Eliza earn in overtime pay? 3. How much does Eliza earn in total for the week?

1. Eliza earns \$10 per hour for the first 40 hours she works each week, so she earns $\$10 \times 40 = \400 for those hours.

2. Eliza earns 1.2 times her regular hourly rate for overtime hours, so she earns $1.2 \times \$10 = \12 per overtime hour. Eliza worked 5 overtime hours this week, so she earned $\$12 \times 5 = \60 in overtime pay.

3. Eliza's total earnings for the week are $\$400 + \$60 = \$460$.

The answer is: \$460

Commonsense reasoning by decomposition ([StrategyQA](#))

Q: Can sunlight travel to the deepest part of the Black Sea?

A: Let's breakdown this problem: "What is the maximum depth of the Black Sea?", "How deep can sunlight penetrate a sea?", "Can sunlight travel to the deepest part of the Black Sea?"

1. The Black Sea has a maximum depth of 2,212 meters.
2. Sunlight does not penetrate water below 1000 meters.
3. 1000 meters is less than 2,212 meters. So sunlight cannot travel to the deepest part of the Black Sea

Then answer is: false.

Q: Did Aristotle use a laptop?

Commonsense reasoning by decomposition (Cont')

...

A: Let's breakdown this problem: "When was the laptop invented?", "When did Aristotle live?", "Did Aristotle Use a Laptop?"

1. The laptop was invented in 1981.
 2. Aristotle lived from 384 BC to 322 BC.
 3. 1981 is greater than 322 BC. So Aristotle did not use a laptop.
- The answer is: false.

Last-letter task (length generalization)

Method	L = 4	L = 6	L = 8	L = 10	L = 12
Standard prompting	0.0	0.0	0.0	0.0	0.0
Chain-of-Thought	89.4	75.0	51.8	39.8	33.6
Least-to-Most	94.0	88.4	83.0	76.4	74.0

Table 3: Accuracies of different prompting methods with `code-davinci-002` on the last-letter-concatenation task with the length of lists increasing from 4 to 12. All the methods are 2-shot.

Q: “think, machine, learning”

A: “think”, “think, machine”, “think, machine, learning”

Decomposition

Table 1: Least-to-most prompt context (decomposition) for the last-letter-concatenation task. It can decompose arbitrary long lists into sequential sublists with an accuracy of 100%.

Q: “think, machine”

A: The last letter of “think” is “k”. The last letter of “machine” is “e”. Concatenating “k”, “e” leads to “ke”. So, “think, machine” outputs “ke”.

Q: “think, machine, learning”

A: “think, machine” outputs “ke”. The last letter of “learning” is “g”. Concatenating “ke”, “g” leads to “keg”. So, “think, machine, learning” outputs “keg”.

Teach LLMs the trick from n to $n+1$

Table 2: Least-to-most prompt context (solution) for the last-letter-concatenation task. The two exemplars in this prompt actually demonstrate a base case and a recursive step.

SCAN (compositional generalization): text-to-actions

Method	Standard prompting	Chain-of-Thought	Least-to-Most
code-davinci-002	16.7	16.2	99.7
text-davinci-002	6.0	0.0	76.0
code-davinci-001	0.4	0.0	60.7

Table 8: Accuracies (%) of different prompting methods on the test set of SCAN under length split. The results of `text-davinci-002` are based on a random subset of 100 commands.

CFQ (compositional generalization): text-to-code

	MCD1	MCD2	MCD3	Ave.
Fully Supervised				
T5-base (Herzig et al., 2021)	58.5	27.0	18.4	34.6
T5-large (Herzig et al., 2021)	65.1	32.3	25.4	40.9
T5-3B (Herzig et al., 2021)	65.0	41.0	42.6	49.5
HPD (Guo et al., 2020)	79.6	59.6	67.8	69.0
T5-base + IR (Herzig et al., 2021)	85.8	64.0	53.6	67.8
T5-large + IR (Herzig et al., 2021)	88.6	79.2	72.7	80.2
T5-3B + IR (Herzig et al., 2021)	88.4	85.3	77.9	83.9
LeAR (Liu et al., 2021)	91.7	89.2	91.7	90.9
Prompting				
(Ours) Dynamic Least-to-Most	94.3	95.3	95.5	95.0

Using only 1% data!

Table 1: Test accuracy across the MCD splits for the CFQ dataset.

Andrew Drozdov, Nathanael Schärli, Ekin Akyürek, Nathan Scales, Xinying Song, Xinyun Chen, Olivier Bousquet, Denny Zhou. [Compositional Semantic Parsing with Large Language Models](#). ICLR 2023.

Is it possible to make one common prompt for all tasks?

Yes!

Key Idea

Making a big prompt by combining prompts from different tasks, and then using it for **any task**

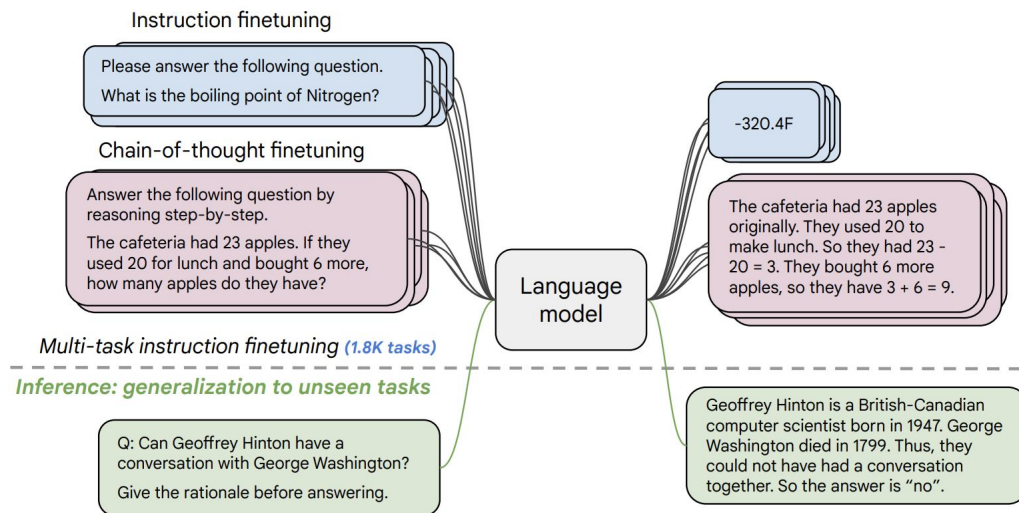
Magic

Any task: including tasks which are no even seen

Implementation

Too big to load? **“Store” them in “weights” ! (Instruction tuning!)**

FLAN2: Finetune PaLM with 1800+ tasks



Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V Le, Jason Wei. [Scaling Instruction-Finetuned Language Models](#). 2022.

Input

```
from: Jason Wei <jasonwei@google.com>
to: Denny Zhou <dennyzhou@google.com>,
Xuezhi Wang <xuezhiw@google.com>,
Yi Tay <yitay@google.com>,
Hyung Won Chung <hwchung@google.com>
date: Oct 25, 2022, 10:41 AM
subject: Fwd: Hi from <redacted>
mailed-by: google.com
security: Standard encryption (TLS) Learn more
```

Task description

Parse out all of the names from this message.

Then, sort them in alphabetical order by first name. Also add "Quoc Le" to the list.

Output

Input

```
from: Jason Wei <jasonwei@google.com>
to: Denny Zhou <dennyzhou@google.com>,
Xuezhi Wang <xuezhiw@google.com>,
Yi Tay <yitay@google.com>,
Hyung Won Chung <hwchung@google.com>
date: Oct 25, 2022, 10:41 AM
subject: Fwd: Hi from <redacted>
mailed-by: google.com
security: Standard encryption (TLS) Learn more
```

This is zero-shot!

Parse out all of the names from this message.

Then, sort them in alphabetical order by first name. Also add "Quoc Le" to the list.

Output

Denny Zhou, Hyung Won Chung, Jason Wei, Quoc Le, Xuezhi Wang, Yi Tay

Pretraining (next-token-prediction) +
Instruction tuning → ChatLLM

Chat is nothing but zero-shot prompting!

Zero-Shot

LLMs as Optimizers

Yang, Chengrun, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. [Large Language Models as Optimizers](#). arXiv preprint arXiv:2309.03409 (2023).

Can LLMs solve math word problems?

Elsa has 3 apples. Anna has 2 more apples than Elsa. How many apples do they have together?

They have 5 apples together.

FAILED

☰ Complete ▾

Model

text-davinci-003 ▾

Temperature 1



Maximum length 256



Stop sequences

Enter sequence and press Tab

Let's think step by step: zero-shot CoT

Elsa has 3 apples. Anna has 2 more apples than Elsa. How many apples do they have together? 🗣️

Let's think step by step.

Step 1: Elsa has 3 apples.

Step 2: Anna has 2 more apples than Elsa.

Step 3: To find out how many apples they have together, we need to add the number of apples that Elsa has (3) to the number of apples that Anna has ($2 + 3 = 5$).

They have 8 apples together.

Mode



Complete

Legacy



Model

text-davinci-003



Temperature

1



Maximum length

256



Stop sequences

Enter sequence and press Tab

Anything better than “Let’s think step by step”?

Let LLMs optimize it!

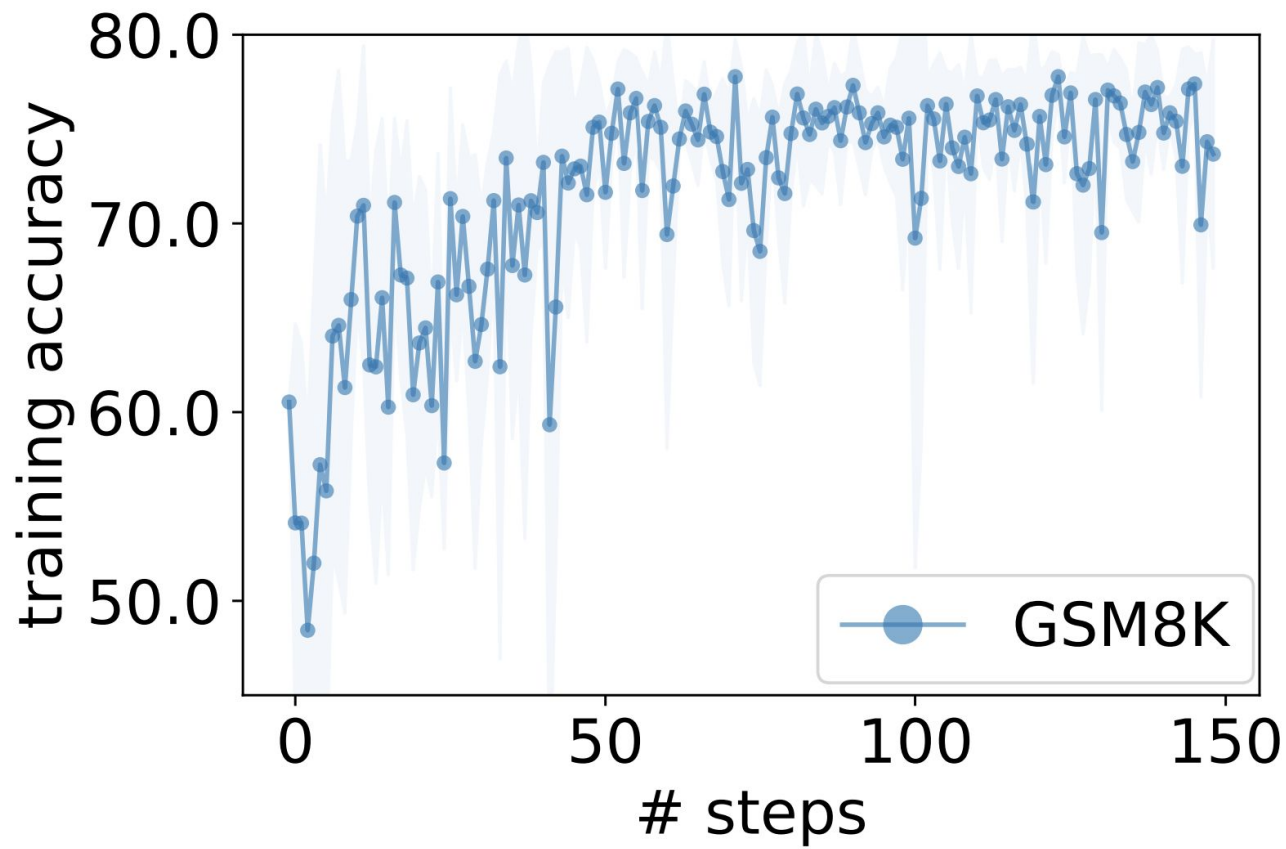
Text: Let's figure it out! **Score:** 61

Text: Let's solve the problem. **Score:** 63

(... more texts and scores ...)

Write your new text that is different from the old ones and has a score as high as possible.





Results on GSM8K w/ PaLM

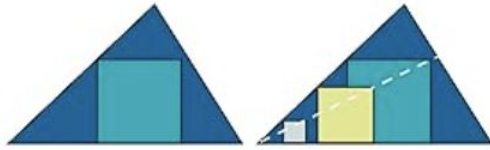
“Let’s think step by step” 71.8%

“Take a deep breath and work on this problem step-by-step” 80.2%
(found by LLM optimization)

LLMs as Analogical Reasoners

Michihiro Yasunaga, Xinyun Chen, Yujia Li, Panupong Pasupat, Jure Leskovec, Percy Liang, Ed H. Chi, Denny Zhou. [Large Language Models as Analogical Reasoners](https://arxiv.org/abs/2310.01714). arXiv:2310.01714 [cs.LG]

Instead of manually making few-shot CoT prompts, why not let LLMs automatically generate related examples?



How to Solve It

a new aspect of
mathematical method

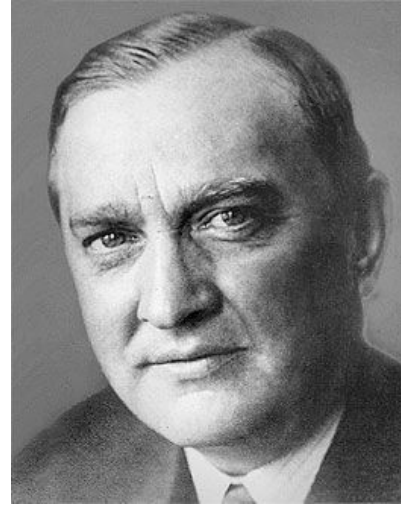
*With a new foreword
by John H. Conway*

G. POLYA

Do you know a related problem? We can scarcely imagine a problem absolutely new, unlike and unrelated to any formerly solved problem; but, if such a problem could exist, it would be insoluble. In fact, when solving a problem, we always profit from previously solved problems, using their result, or their method, or the experience we acquired solving them. And, of course, the problems from which we profit must be in some way related to our present problem. Hence the question: *Do you know a related problem?*

There is usually no difficulty at all in recalling formerly solved problems which are more or less related

A mathematician is a person who can find analogies between theorems; a better mathematician is one who can see analogies between proofs and the best mathematician can notice analogies between theories. One can imagine that the ultimate mathematician is one who can see analogies between analogies.



Stefan Banach

Model Input

Q: What is the area of the square with the four vertices at $(-2, 2)$, $(2, -2)$, $(-2, -6)$, and $(-6, -2)$?

Instruction:

Recall relevant exemplars:

Solve the initial problem:

Model Output

Relevant exemplars:

Q: What is the area of the square with a side length of 5?

A: The area of a square is found by squaring the length of its side. So, the area of this square is $5^2 = 25$

Solve the initial problem:

To find the area of the square, we need to find the side length. The length is ... $\sqrt{(2 - (-2))^2 + (-2 - 2)^2} = \sqrt{32}$. So, the area of the square is $(\sqrt{32})^2 = 32$.

Prompting Method	GSM8K Accuracy			MATH Accuracy	
	GPT3.5-turbo	text-davinci-003	PaLM2	GPT3.5-turbo	PaLM2
0-shot	75.0%	14.8%	60.8%	33.0%	27.1%
0-shot CoT	75.8%	50.3%	78.2%	33.9%	29.8%
5-shot CoT	76.7%	54.0%	80.7%	34.9%	34.3%
Ours: Self-generated Exemplars	77.8%	61.0% [†]	81.7%	37.3%	34.8%

Your goal is to write Python3 code to solve competitive programming problems. Given a problem, explain the core concepts in it and provide other relevant problems. Then solve the original problem.

Problem: You are given a sequence a_1, a_2, \dots, a_n . Each element of a is 1 or 2. Find out if an integer k exists so that the following conditions are met. $1 \leq k \leq n-1$, and $a_1 \cdot a_2 \dots a_k = a_{k+1} \dots a_n$. If there exist multiple k that satisfy the given condition, print the smallest.

Instruction:

Algorithms: Identify the core concepts or algorithms used to solve the problem.

Tutorial: Write a tutorial about these algorithms.

Example Problems: Provide three examples of relevant competitive programming problems that involve these algorithms. For each problem, describe the problem, explain the solution in detail, and then write the correct Python3 code.

Python3 code to solve the original problem: - Explanation of the solution: - Python3 code to solve the problem:

Prompting Method	GPT3.5-turbo-16k		GPT4	
	Acc@1	Acc@10	Acc@1	Acc@10
0-shot	8%	24%	16%	30%
0-shot CoT	9%	27%	16%	29%
3-shot CoT	11%	27%	17%	31%
Ours: Self-generated Exemplars	13%	25%	17%	32%
Ours: Self-generated Knowledge + Exemplars	15%	29%	19%	37%

Surprising findings

- The automatically generated problems are usually easier than the problem to solve
- The automatically generated problems may contain mistakes

Despite these, the analogical method still works so well!

Why LLMs can reason?

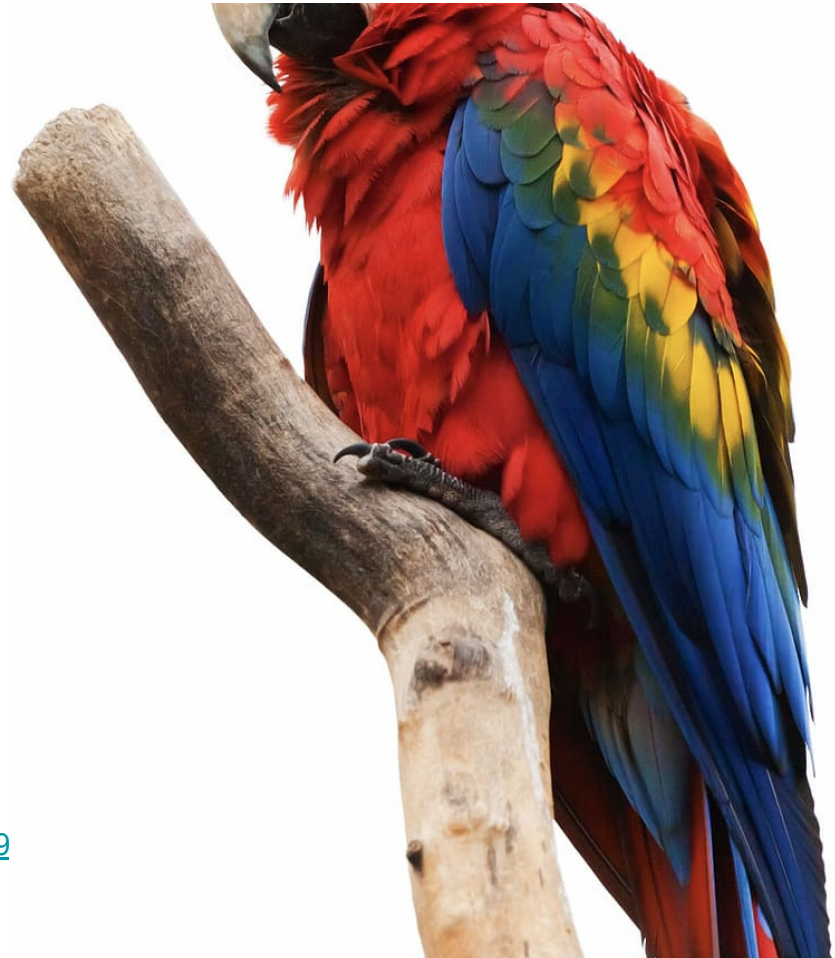
Reasoning as an emergent behavior

- Reasoning emerges from next-token-prediction pretraining
 - CoT is discovered, not manually designed
 - Sharp reasoning performance transition vs (#parameters, #tokens)
- Distill big models to small ones? Check performance on reasoning tasks!

“How to make parrots
intelligent?”

“Scaling up!”

https://twitter.com/denny_zhou/status/159145184723932544?s=20



Toward understanding in-context learning

- Transformer models are meta-learners: implicitly learned training algorithms (like gradient descent) from pretraining
- In the inference time, transformer models implicitly built the prediction model from the inputs and then predict

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. [What learning algorithm is in-context learning? Investigations with linear models.](#) ICLR 2023.

Summary

- **Chain-of-thought:** add “thought” before final “answer”
- **Self-consistency:** sample repeatedly, and select the most frequent answer
- **Least-to-most:** decompose to subproblems and solve them one by one
- **Instruction finetuning:** enable zero-shot / chat
- Auto prompting (LLMs as optimizers, LLMs as analogical reasoners)
- Emergence of LLM reasoning and theory

What is next?

A model with language understanding and reasoning opens a door to infinite possibilities

<https://colmweb.org>

Conference on Language Modeling (COLM)



Thank You



https://twitter.com/denny_zhou



<https://dennyzhou.github.io/>



<https://scholar.google.com/citations?user=UwLsYw8AAAAJ&hl=en>