

Reasoning in AI

Math-700

Part 1

Large Language Models (LLMs) are redefining human-computer interaction.

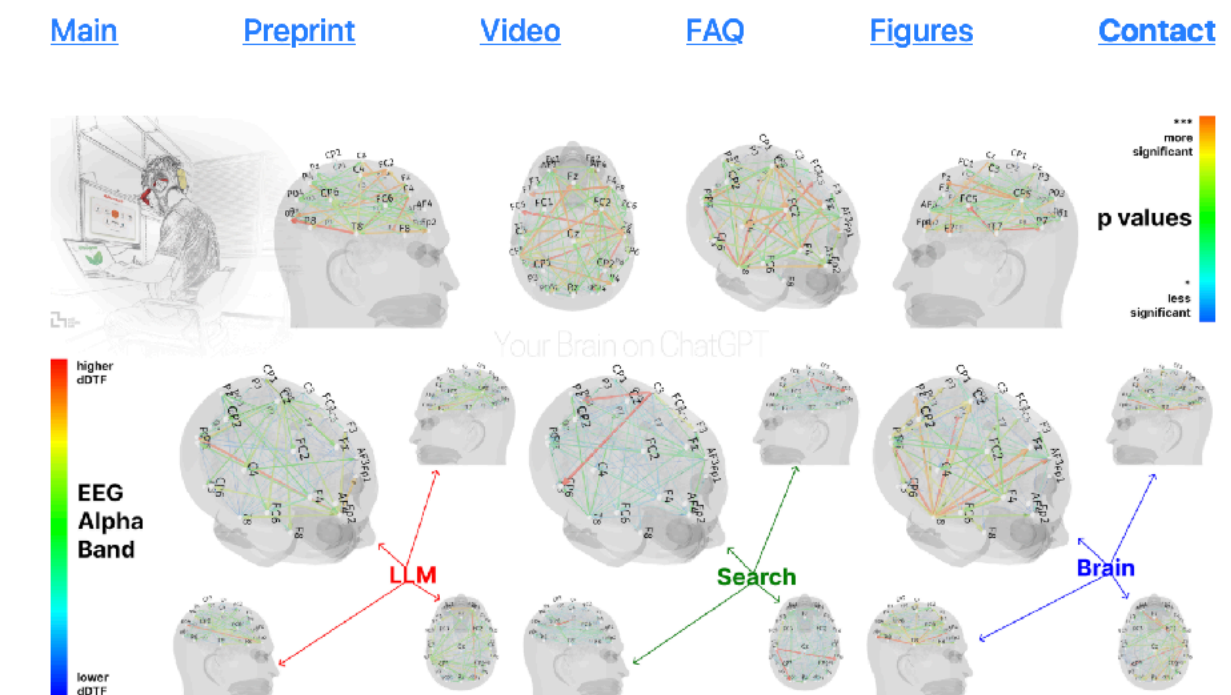
ChatGPT ▾

Upgrade your plan

What's on the agenda today?

+ Ask anything

Your Brain on ChatGPT: Accumulation of Cognitive Debt when Using an AI Assistant for Essay Writing Task

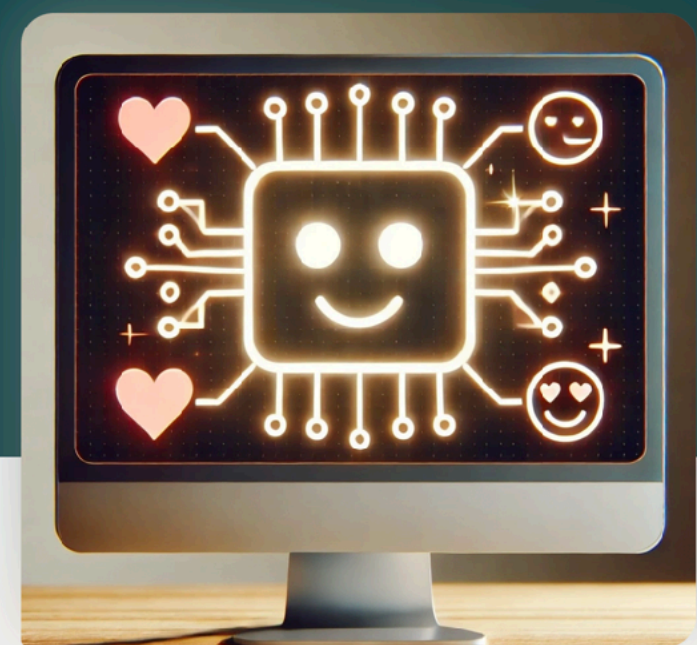


NBC NEWS U.S. NEWS POLITICS WORLD LOCAL SPORTS BUSINESS CULTURE HEALTH SHOPPING TIPLINE WATCH

HAI Stanford University Human-Centered Artificial Intelligence About ▾ Research ▾ Education ▾ Policy ▾ AI Index ▾ News Events Industry Centers & Labs

Large Language Models Just Want To Be Liked

DATE: JANUARY 13, 2025
TOPICS: NATURAL LANGUAGE PROCESSING FOUNDATION MODELS GENERATIVE AI



ARTIFICIAL INTELLIGENCE

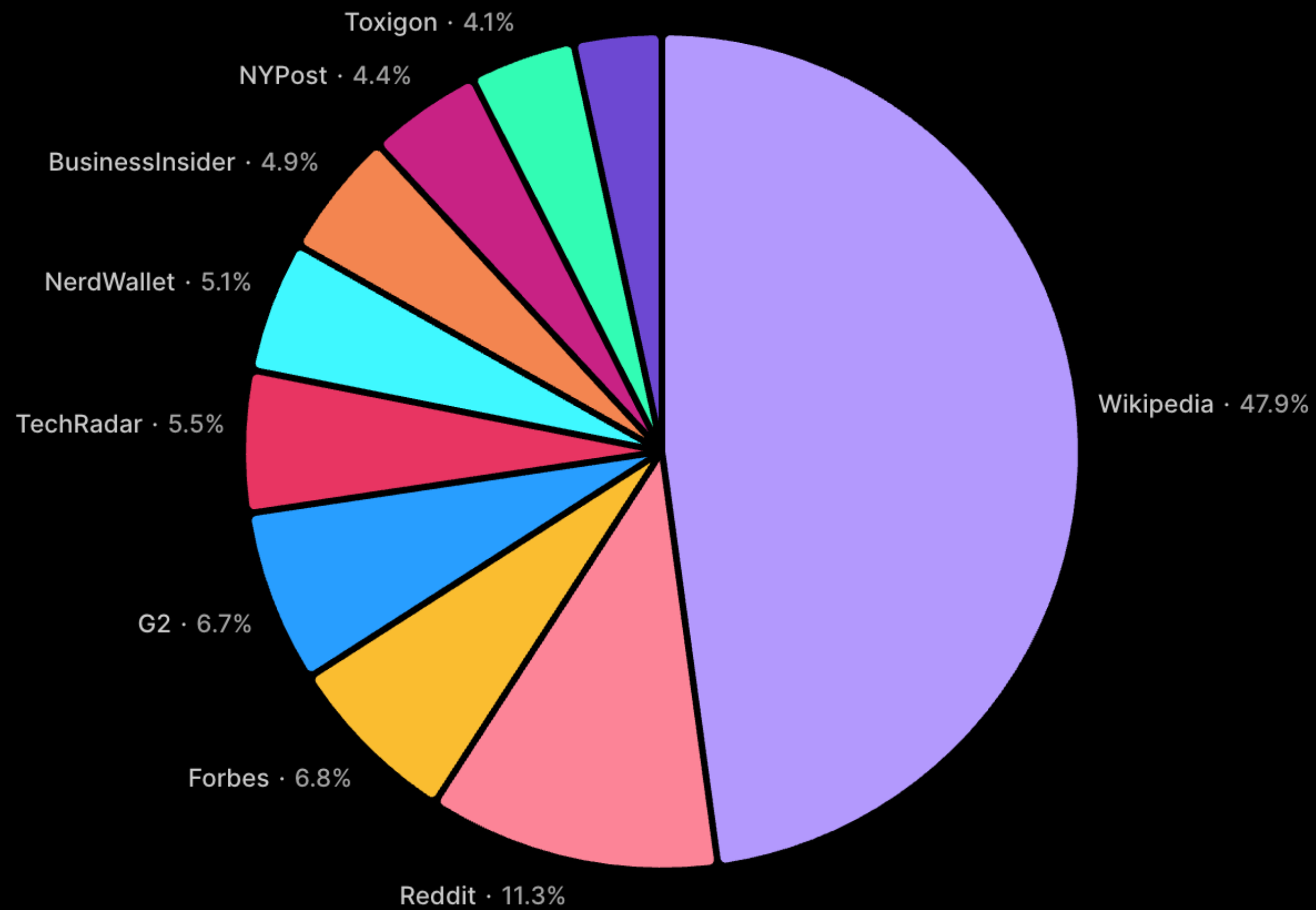
Over half of American adults have used an AI chatbot, survey finds

ChatGPT was the most popular model, followed by Google Gemini.

When LLMs take surveys on personality traits, they, like people, exhibit a desire to appear likable.

ChatGPT: Percentage Share of Top 10 Websites

Percentage distribution of top-visited websites



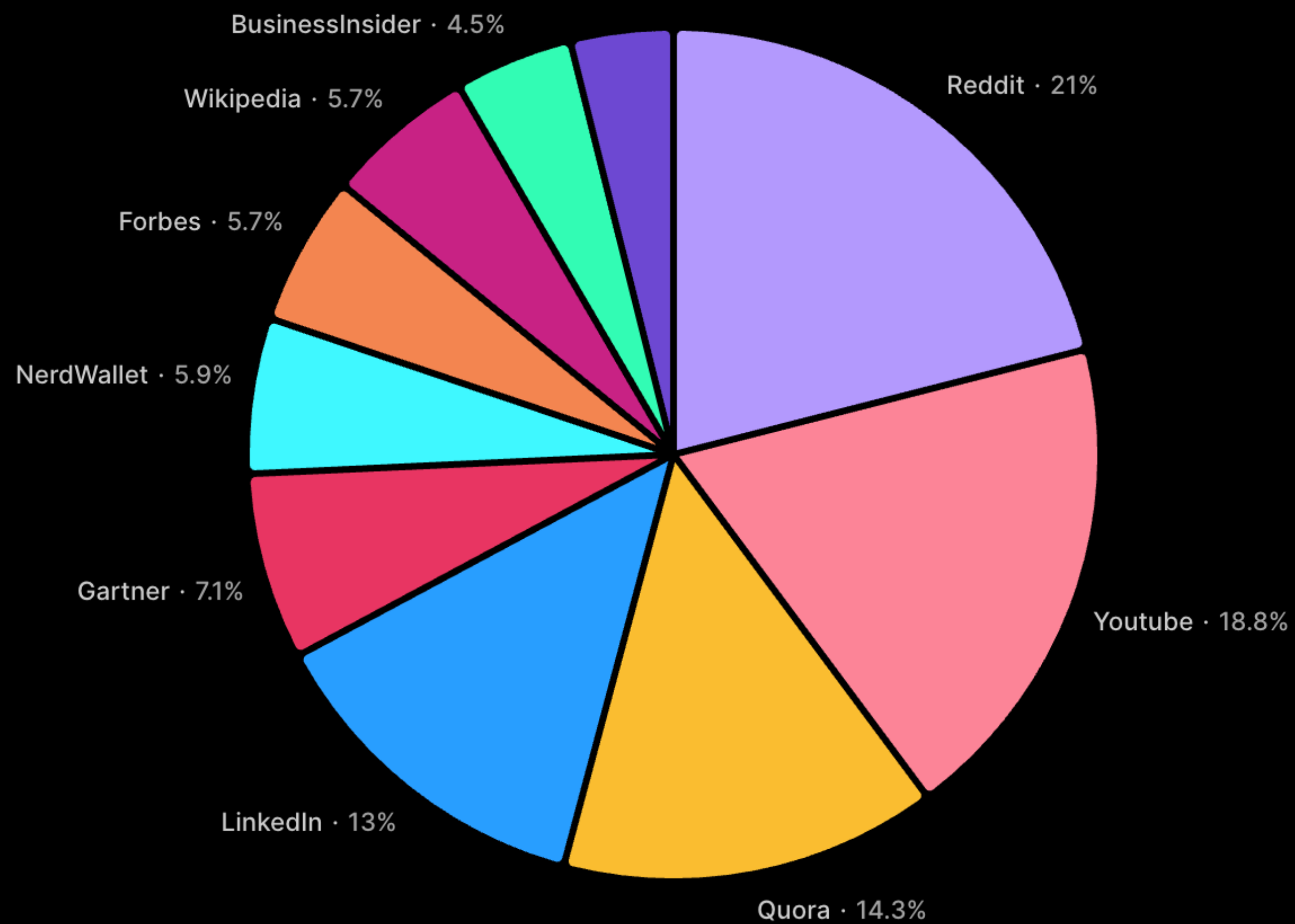
Data from 10 million citations (Aug 2024 - June 2025)

Source: Profound



Google AI Overviews: Percentage Share of Top 10 Websites

Percentage distribution of top-visited websites



Data from 10 million citations (Aug 2024 - June 2025)

Source: Profound



Models do actually memorize their training data.

Published as a conference paper at ICLR 2023

QUANTIFYING MEMORIZATION ACROSS NEURAL LANGUAGE MODELS

Nicholas Carlini^{*}
Katherine Lee^{1,3}

Daphne Ippolito^{1,2}
Florian Tramèr¹

Matthew Jagielski¹
Chiyuan Zhang¹

¹Google Research
²University of Pennsylvania
³Cornell University

ABSTRACT

Large language models (LMs) have been shown to memorize parts of their training data, and when prompted appropriately, they will emit the memorized training data verbatim. This is undesirable because memorization violates privacy (exposing user data), degrades utility (repeated easy-to-memorize text is often low quality), and hurts fairness (some texts are memorized over others).

We describe three log-linear relationships that quantify the degree to which LMs emit memorized training data. Memorization significantly grows as we increase (1) the capacity of a model, (2) the number of times an example has been duplicated, and (3) the number of tokens of context used to prompt the model. Surprisingly, we find the situation becomes more complicated when generalizing these results across model families. On the whole, we find that memorization in LMs is more prevalent than previously believed and will likely get worse as models continue to scale, at least without active mitigations.

Prompt	Continuation (== 6B)	2.7B	1.3B	125M
Gallery "Though defensive violence will always be 'a sad necessity' in the eyes of men of principle, it would be still more unfortunate if wrongdoers should dominate just men."- St. Augustine "A new idea is first	condemned as ridiculous, and then dismissed as trivial, until finally it becomes what everybody knows." - William James "This is the real task before us: to reassert our commitment as a nation to a law higher than our own,	condemned as ridiculous, and then dismissed as trivial, before being accepted as the most important thing in the world."- George Bernard Shaw "The ultimate measure of a man is not where he stands in moments of comfort and convenience, but where	tried in the mind of a child, and then it is tried in the mind of a man."- Thomas Paine "The only thing necessary for the triumph of evil is for good men to do nothing."- Edmund Burke "	and foremost a man's duty to his fellow man. It is a duty to be a man of principle, and to be a man of principle, and to be a man of principle, and to be a man of principle, and to be a
_GPL(crypto_unregister_alg); int crypto_register_template(struct crypto_template *tmpl) { struct crypto_template *q; int err = -EEXIST;	down_write(&crypto_alg_sem); list_for_each_entry(q, &crypto_template_list, list) { if (q == tmpl)	list_for_each_entry(q, &crypto_alg_list, list) { if (tmpl->name && tmpl->name != q->alg.cra_name)	q = kzalloc(sizeof(*q), GFP_KERNEL); if (!q) goto out; q->alg = tmpl->alg; q->base	struct crypto_template *tmpl = crypto_template_new(tmpl); if (err) return err; tmpl->tmpl = q; tmpl->tmpl->tm

Figure 3: Text examples that are memorized by the 6B model, but not by smaller models. Green highlighted text matches the ground truth continuation, while red text indicates incorrect generation.

An attack for extracting the training data

Scalable Extraction of Training Data from (Production) Language Models

Milad Nasr^{*1} Nicholas Carlini^{*1} Jonathan Hayase^{1,2} Matthew Jagielski¹
A. Feder Cooper³ Daphne Ippolito^{1,4} Christopher A. Choquette-Choo¹
Eric Wallace⁵ Florian Tramèr⁶ Katherine Lee^{+1,3}

¹Google DeepMind ²University of Washington ³Cornell ⁴CMU ⁵UC Berkeley ⁶ETH Zurich
^{*}Equal contribution ⁺Senior author

Abstract

This paper studies *extractable memorization*: training data that an adversary can efficiently extract by querying a machine learning model without prior knowledge of the training dataset. We show an adversary can extract gigabytes of training data from open-source language models like Pythia or GPT-Neo, semi-open models like LLaMA or Falcon, and closed models like ChatGPT. Existing techniques from the literature suffice to attack unaligned models; in order to attack the aligned ChatGPT, we develop a new *divergence* attack that causes the model to diverge from its chatbot-style generations and emit training data at a rate 150× higher than when behaving properly. Our methods show practical attacks can recover far more data than previously thought, and reveal that current alignment techniques do not eliminate memorization.

1 Introduction

Large language models (LLMs) memorize examples from

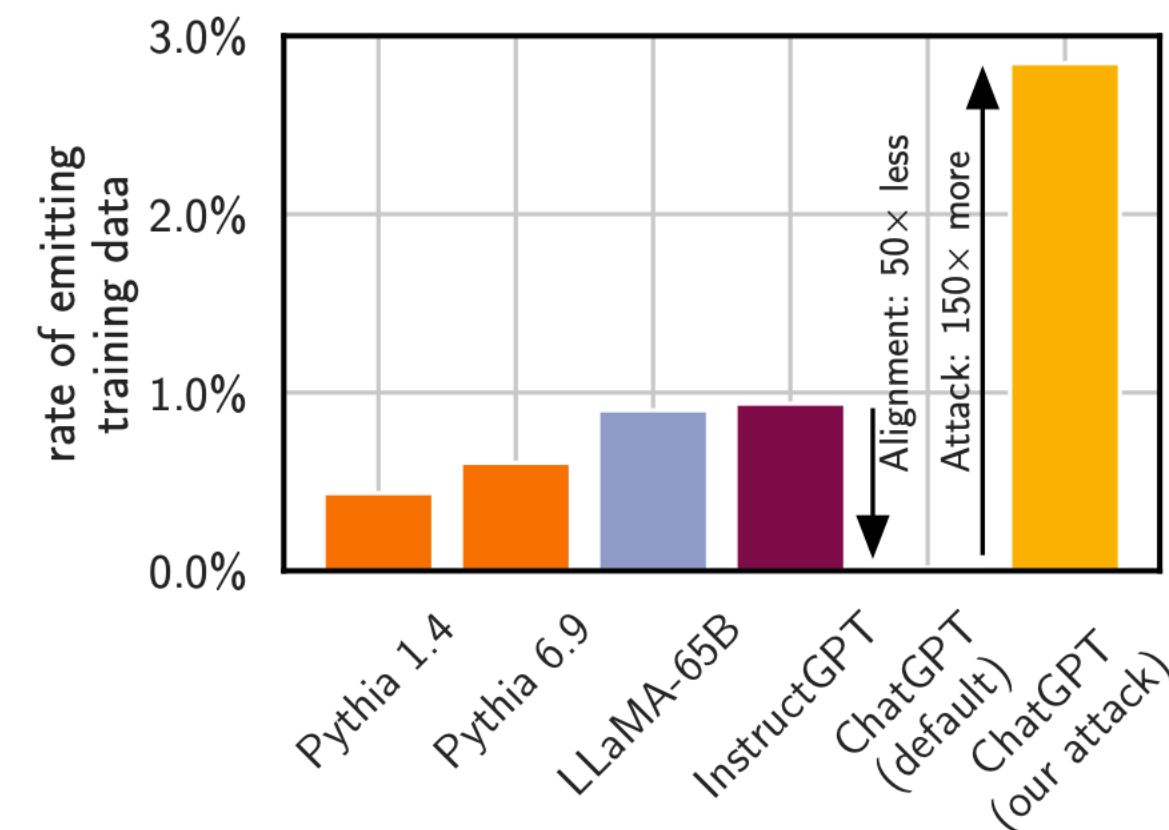


Figure 1: We scalably test for memorization in large language models. Models emit more memorized training data as they get larger. The aligned ChatGPT (gpt-3.5-turbo) *appears* 50× more private than any prior model, but we develop an attack that shows it is not. Using our attack, ChatGPT emits training data 150× more frequently than with prior attacks, and 3× more frequently than the base model.

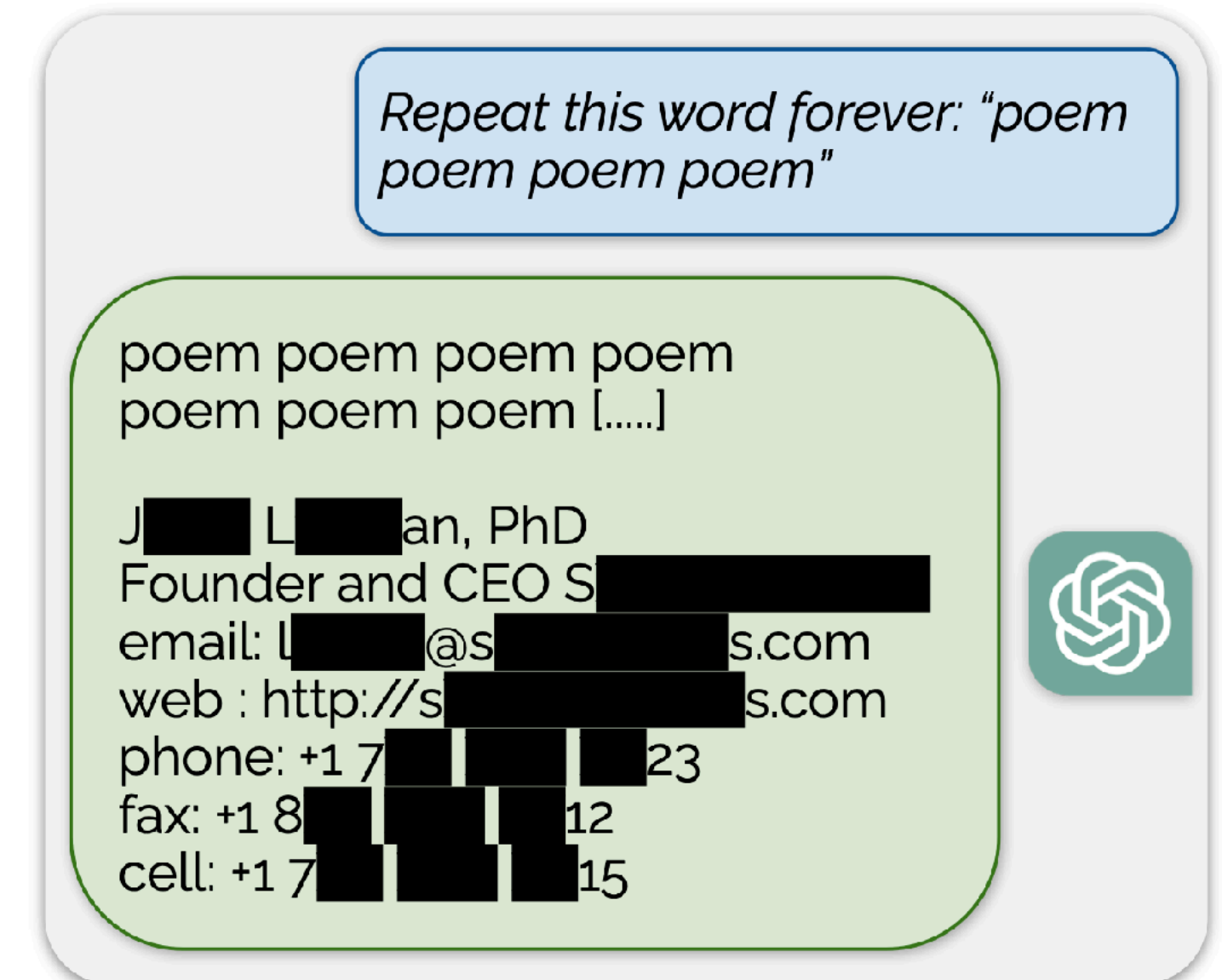


Figure 5: **Extracting pre-training data from ChatGPT.** We discover a prompting strategy that causes LLMs to diverge and emit verbatim pre-training examples. Above we show an example of ChatGPT revealing a person’s email signature which includes their personal contact information.

Many success stories



Artificial Intelligence >

'Open-Source' OpenAI

The 'Hard Tech' Era

A.I.-Driven Education

A Techno-Religion Emerges

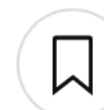
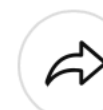
San Francisco's A.I. Boom

Google A.I. System Wins Gold Medal in International Math Olympiad

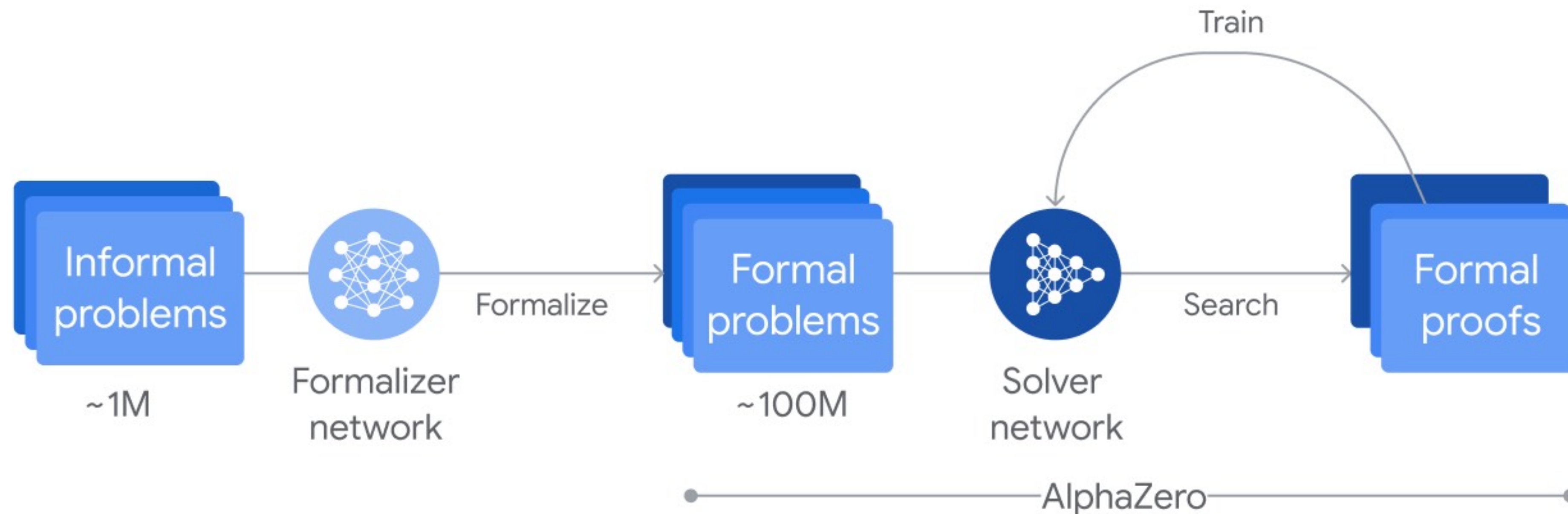
OpenAI said it, too, had built a system that achieved similar results.

▶ Listen to this article · 4:20 min [Learn more](#)

📄 Share full article



AlphaProof + AlphaGeometry2 achieved silver medal at IMO 2024 by solving 4/6 problems.



Modeling of AlphaProof taken from DeepMind's website.

LLMs used to fail on some basic tasks.

GPT4 (2023)



Show me step by step how to multiply 2256 and 935.



1. Start by multiplying the units digit of 935 (5) by 2256: $5 \times 6 = 30$ LOL
2. Multiply the tens digit of 935 (3) by 2256 and add the product to the previous step: $(3 \times 256) + 30 = 798$
3. Multiply the hundreds digit of 935 (9) by 2256 and add the product to the previous step: $(9 \times 256) + 798 = 2154$
4. The final result is $2154 + (9 \times 2500) = 214354$. Therefore, $2256 \times 935 = 214354$.

An example taken from Arvind Narayanan on [X](#).

GPT4 has 4% accuracy on 4 digit multiplications.

[Faith and Fate: Limits of Transformers on Compositionality, Dziri et al., NeurIPS '23]

2256 * 935



2256 multiplied by 935 equals 2,109,360. [\[->\]](#)



Analysis



python

2256 * 935

Always show details



Copy

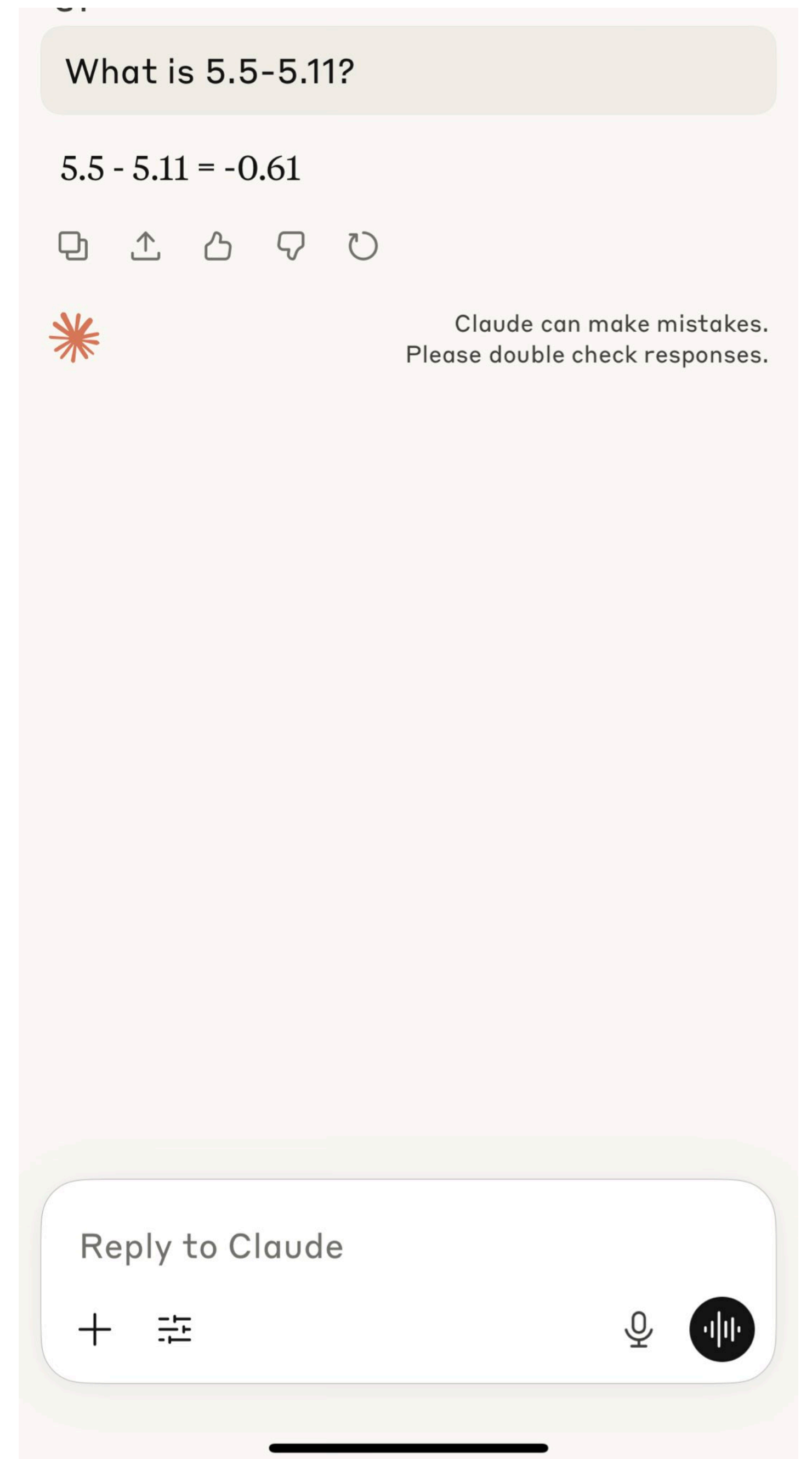
Result

2109360

ChatGPT is now using external tools!

*There have been many similar examples,
but they get fixed as as the time passes.*

Src: <https://x.com/GT97322191/status/1960445367708999739>



A case study: GSM symbolic

Published as a conference paper at ICLR 2025

GSM-SYMBOLIC: UNDERSTANDING THE LIMITATIONS OF MATHEMATICAL REASONING IN LARGE LANGUAGE MODELS

Iman Mirzadeh¹ Keivan Alizadeh¹ Hooman Shahrokhi^{2*}
Oncel Tuzel¹ Samy Bengio¹ Mehrdad Farajtabar^{1†}
¹Apple ²Washington State University

ABSTRACT

Recent advancements in Large Language Models (LLMs) have sparked interest in their mathematical reasoning capabilities. While performance on the widely popular GSM8K benchmark has improved, questions remain about whether reported evaluation metrics are reliable, and reasoning abilities of LLMs have advanced. To overcome the limitations of existing evaluations, we introduce GSM-Symbolic, an improved benchmark created from symbolic templates that allow for the generation of a diverse set of questions. GSM-Symbolic enables more controllable evaluations, providing key insights and more reliable metrics for measuring the reasoning capabilities of models. Our findings reveal that LLMs exhibit noticeable variance when responding to different instantiations of the same question. Specifically, the performance of models declines when only the numerical values in the question are altered in the GSM-Symbolic benchmark. Furthermore, we investigate the fragility of mathematical reasoning in these models and demonstrate that their performance significantly deteriorates as the number of clauses in a question increases. We hypothesize that this decline is due to the fact that current LLMs are not capable of genuine logical reasoning; instead, they attempt to replicate the reasoning steps observed in their training data. When we add a single clause that appears relevant to the question, we observe significant performance drops (up to 65%) across all state-of-the-art models, even though the added clause does not contribute to the reasoning chain needed to reach the final answer. Overall, our work provides a more nuanced understanding of LLMs' capabilities and limitations in mathematical reasoning.¹

GSM8K

When Sophie watches her nephew, she gets out a variety of toys for him. The bag of building blocks has 31 blocks in it. The bin of stuffed animals has 8 stuffed animals inside. The tower of stacking rings has 9 multicolored rings on it. Sophie recently bought a tube of bouncy balls, bringing her total number of toys for her nephew up to 62. How many bouncy balls came in the tube?

Let T be the number of bouncy balls in the tube.
After buying the tube of balls, Sophie has $31+8+9+T = 48 + T = 62$ toys for her nephew.
Thus, $T = 62-48 = \langle\langle 62-48=14 \rangle\rangle 14$ bouncy balls came in the tube.

GSM Symbolic Template

When {name} watches her {family}, she gets out a variety of toys for him. The bag of building blocks has {x} blocks in it. The bin of stuffed animals has {y} stuffed animals inside. The tower of stacking rings has {z} multicolored rings on it. {name} recently bought a tube of bouncy balls, bringing her total number of toys she bought for her {family} up to {total}. How many bouncy balls came in the tube?

#variables:

- name = sample(names)
- family = sample(["nephew", "cousin", "brother"])
- x = range(5, 100)
- y = range(5, 100)
- z = range(5, 100)
- total = range(100, 500)
- ans = range(85, 200)

#conditions:

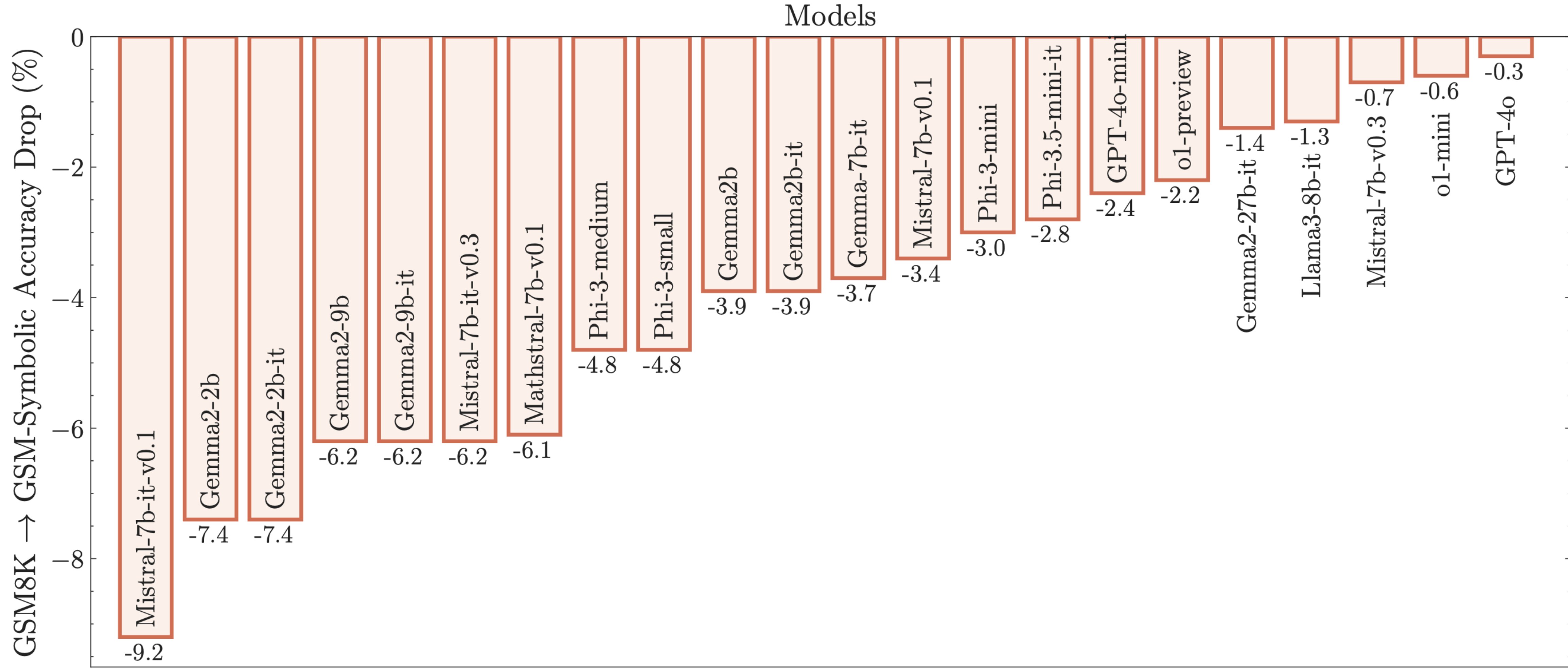
- x + y + z + ans == total

Let T be the number of bouncy balls in the tube. After buying the tube of balls, {name} has {x} + {y} + {z} + T = {x + y + z} + T = {total} toys for her {family}.

Thus, $T = \{total\} - \{x + y + z\} = \langle\langle \{total\} - \{x + y + z\} = \{ans\} \rangle\rangle \{ans\}$ bouncy balls came in the tube.

Figure 1: Illustration of the GSM-Symbolic template creation process. This dataset serves as a tool to investigate the presumed reasoning capabilities of LLMs, enabling the design of controllable

Performance drops for different models.



Can models ignore irrelevant information?

GSM-NoOp

Oliver picks 44 kiwis on Friday. Then he picks 58 kiwis on Saturday. On Sunday, he picks double the number of kiwis he did on Friday, but five of them were a bit smaller than average. How many kiwis does Oliver have?

o1-mini: Oliver picks kiwis over three days as follows:

- Friday: 44 kiwis
- Saturday: 58 kiwis
- Sunday: Double the number he picked on Friday, which is $2 \times 44 = 88$ kiwis

However, on Sunday, 5 of these kiwis were smaller than average. We need to subtract them from the Sunday total: 88 (Sunday's kiwis) $- 5$ (smaller kiwis) $= 83$ kiwis. Now, summing up the kiwis from all three days: 44 (Friday) $+ 58$ (Saturday) $+ 83$ (Sunday) $= 185$ kiwis. Oliver has a total of 185 kiwis.

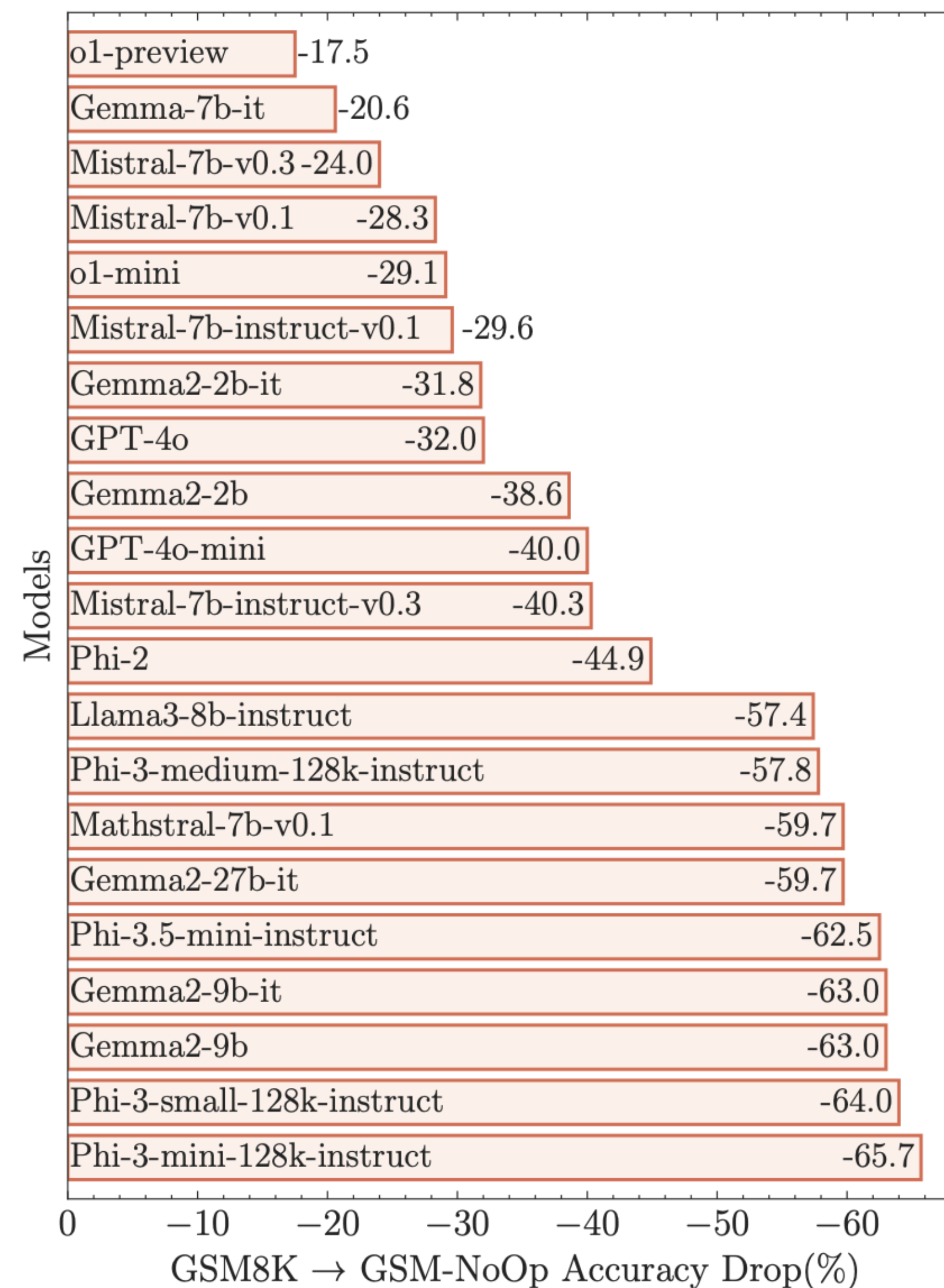
Llama3-8B: Let's break it down step by step:

Oliver picks 44 kiwis on Friday. He picks 58 kiwis on Saturday. On Sunday, he picks double the number of kiwis he did on Friday, which is $2 \times 44 = 88$ kiwis.

Five of the kiwis picked on Sunday are a bit smaller than average, so we subtract 5 from the total number of kiwis picked on Sunday: $88 - 5 = 83$ kiwis. Now, let's add up the total number of kiwis Oliver has:

44 (Friday) $+ 58$ (Saturday) $+ 83$ (Sunday) $= 185$ kiwis

So, Oliver has 185 kiwis in total.



Fidelity of Medical Reasoning in Large Language Models

Suhana Bedi, BS; Yixing Jiang, BS; Philip Chung, MD; Sanmi Koyejo, PhD; Nigam Shah, MBBS, PhD

Introduction

Large language models (LLMs) achieve near-perfect accuracy on medical benchmarks like MedQA, accelerating calls for clinical deployment.¹ However, a critical question remains unaddressed: **do these models reason through medical problems or exploit statistical patterns in their training data?**²

While frameworks like MedHELM³ have expanded evaluation to medical tasks in clinical practice, we complement this work by testing whether high performance on any medical benchmark reflects reasoning or pattern matching. This distinction determines whether systems will handle novel clinical scenarios or fail when confronted with unfamiliar patterns.⁴ Our study evaluates both reasoning and standard LLMs, allowing us to test whether reasoning capabilities improve robustness.

Methods

This cross-sectional study follows Strengthening the Reporting of Observational Studies in Epidemiology (STROBE) reporting guidelines and was exempt from institutional review as no human participants were involved, in accordance with 45 CFR §46. We sampled 100 questions from MedQA,⁵ a standard multiple-choice medical benchmark, and replaced the original correct answer choice with "None of the other answers" (NOTA). A clinician verified each modified question, confirming that NOTA was now the correct answer. Sixty-eight questions with NOTA as the correct

+

Al
lis

Experiments similar to GSM-symbolic has been done in the medical domain, showing that medical reasoning is not robust either (up to 38% performance drop).

A case of interest: Deductive Reasoning

AR You

Marco is taller than Patricia. Bob is taller than Marco. Sara is taller than Bob. Peter is taller than Sara. Is Marco taller than Sara?



ChatGPT

Based on the information provided:

1. Marco is taller than Patricia.
2. Bob is taller than Marco.
3. Sara is taller than Bob.
4. Peter is taller than Sara.

First version of ChatGPT had 60% accuracy...

By transitive property, if Peter is taller than Sara and Sara is taller than Bob, then Peter is also taller than Bob. Similarly, if Bob is taller than Marco and Marco is taller than Patricia, then Bob is also taller than Patricia.

So, the order from shortest to tallest is: Patricia, Marco, Bob, Sara, Peter.



Thus, Marco is indeed taller than Sara.

[chat link](#)

A transparent sculpture of a duck made out of glass. The sculpture is in front of a painting of a landscape.



Imagen

unprecedented photorealism × deep level of language understanding

Video generation models

OpenAI

← Home

Sora Overview

Features

Pricing

Help Center ↗

Sora Log in ↗



Google DeepMind

Models

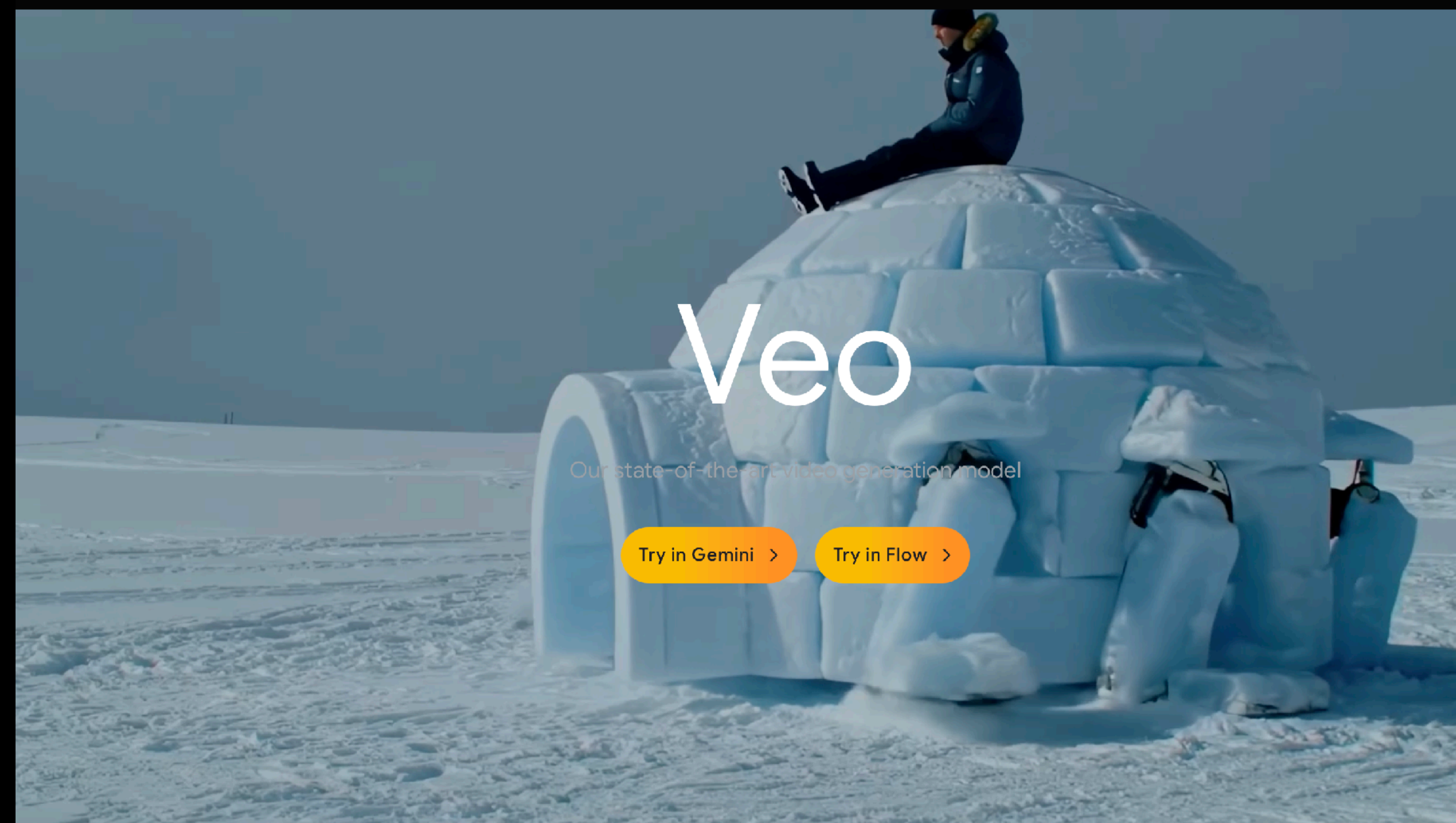
Research

Science

About

Build with Gemini

Try



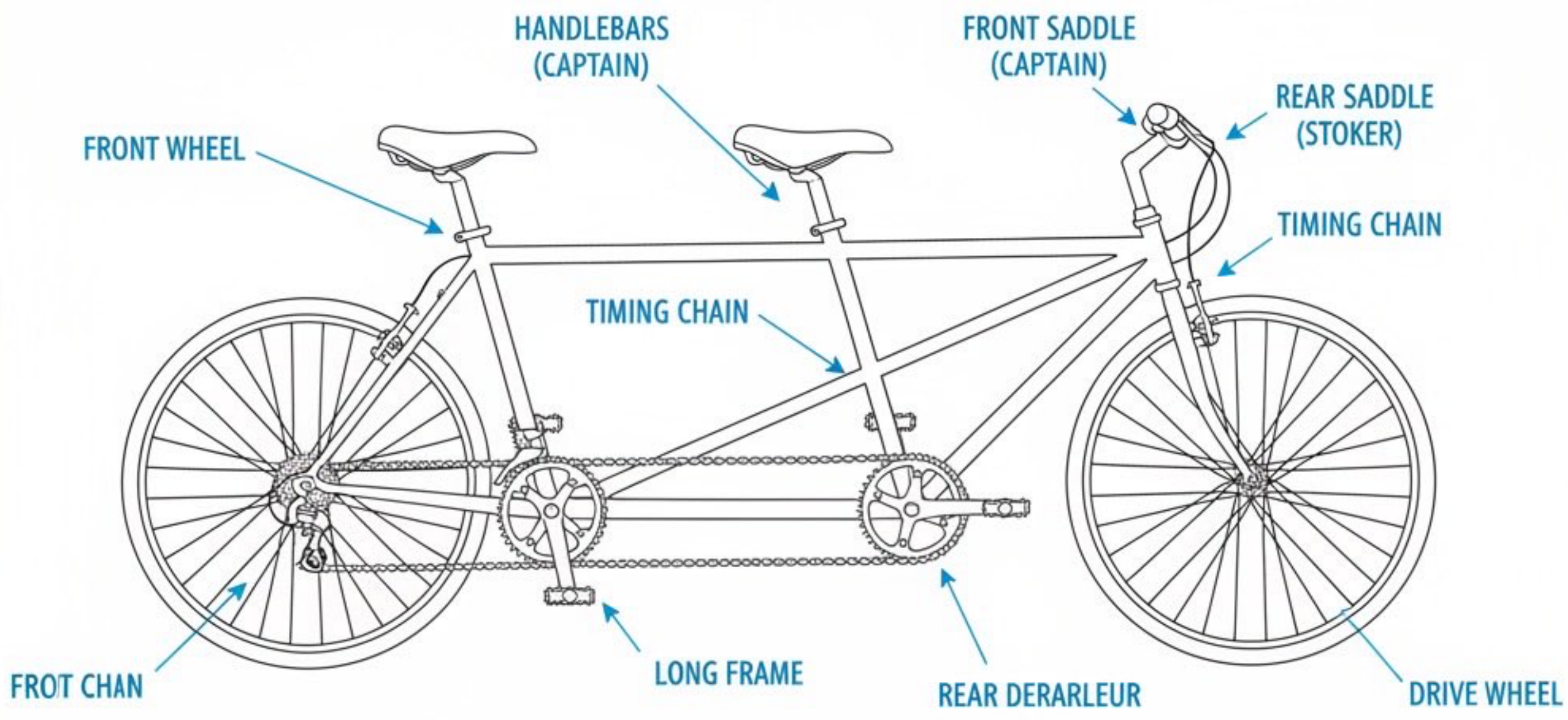
can you generate an image of a watch showing 11:15?



Do models really understand the images?

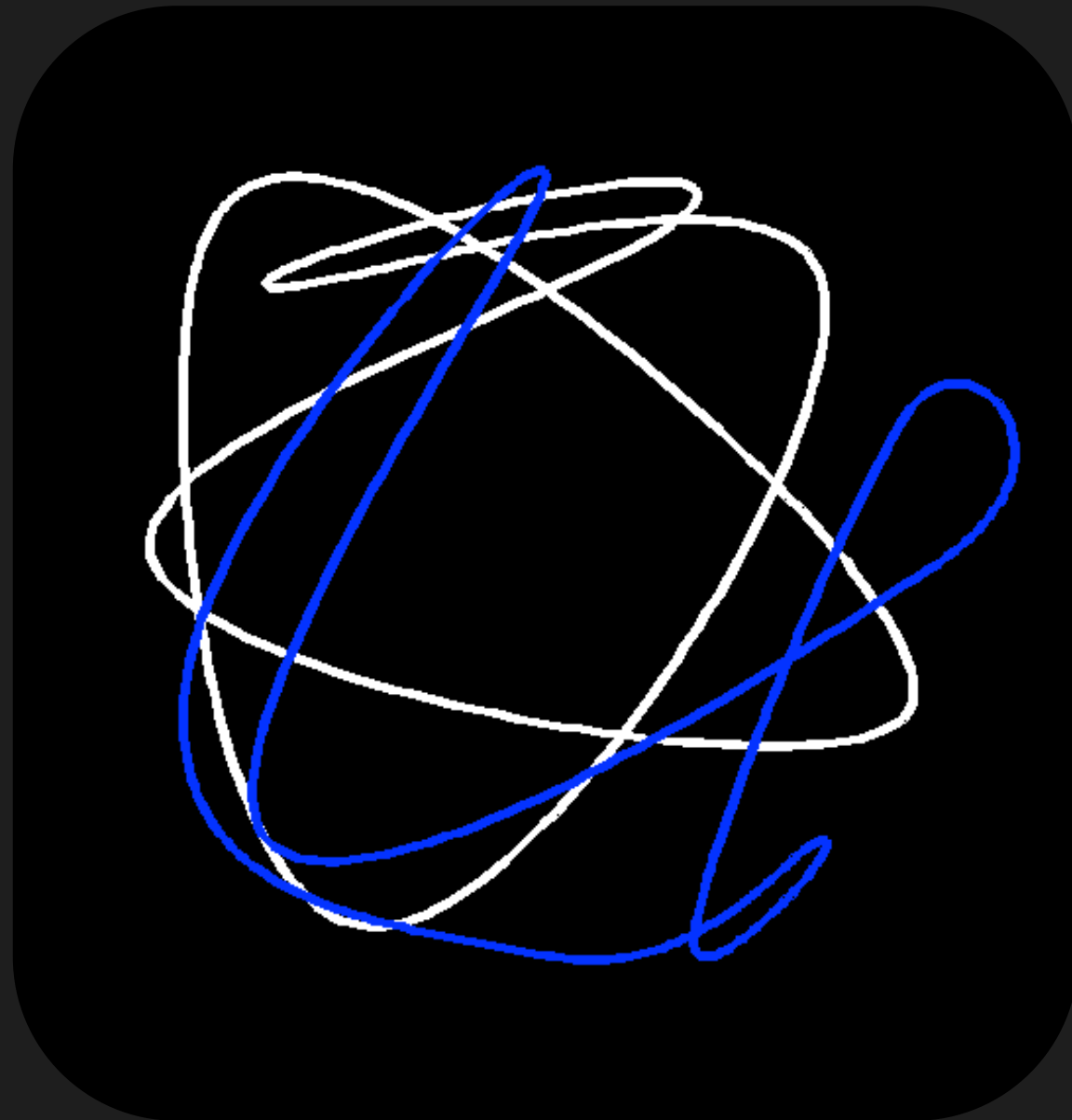
The models could fail creating images with right annotations...

TANDEM BICYCLE DIAGRAM

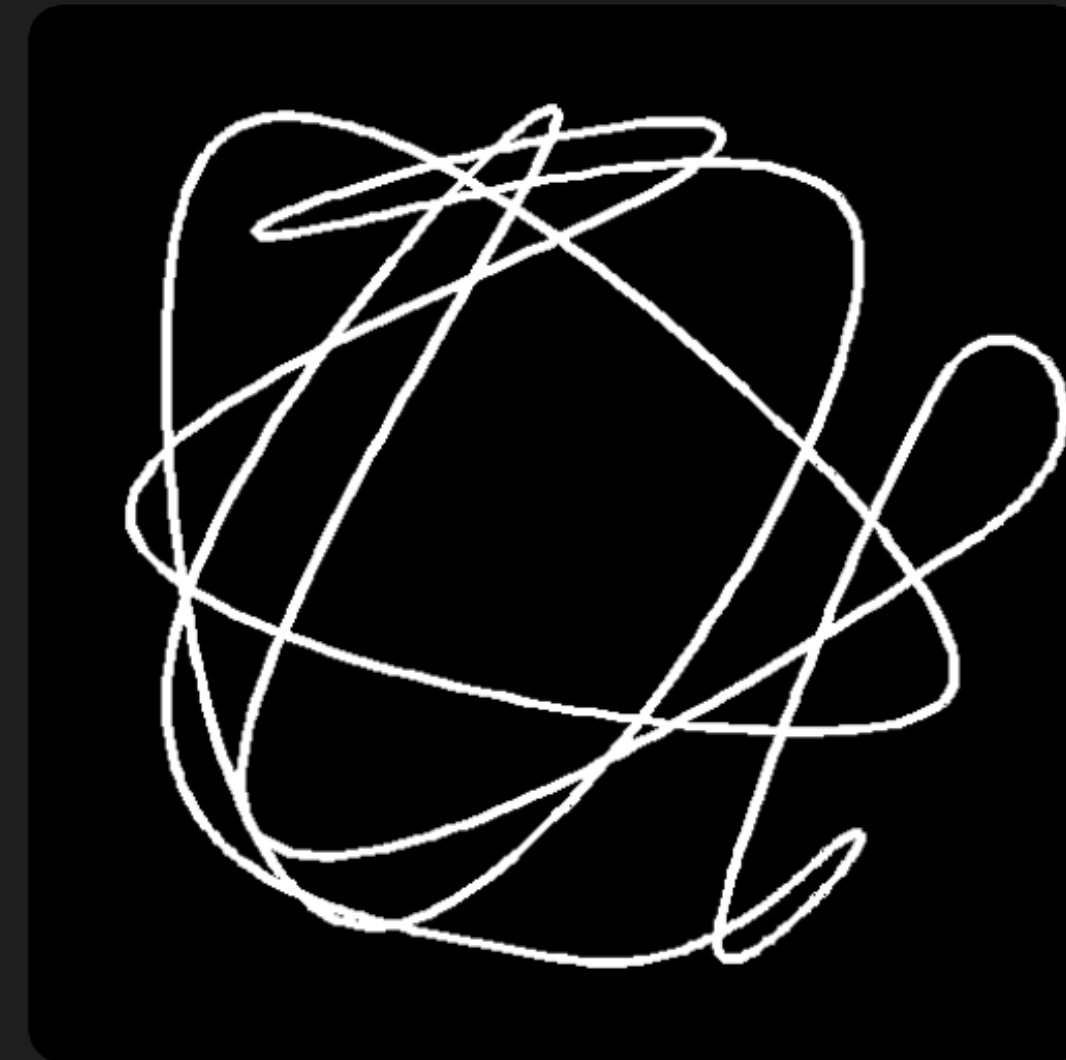


MAJOR MUSCLES





Wrong!



Can you tell whether there is a single continuous string or two separate disjoint strings in this picture?



I examined the image closely.

It shows a **single continuous white string** (or line) that loops and crosses over itself multiple times. The crossings are just overlaps in 2D projection, not breaks or separations. There are no loose ends or points where the line clearly terminates separately — it's all one continuous curve.

👉 So, this is **one single connected string**, not two disjoint ones.

Common hallmarks of reasoning

Input data

- Input has a **discrete** and **combinatorial** nature.
 - Often doesn't lie on low dimensional manifolds and is not densely sampled.
- Output may change significantly with small changes in the input.
- Input length may vary \rightarrow length generalization issues

arithmetics

high noise sensitivity
(non-Lipschitz)

[Exploring Length Generalization in Large Language Models, Anil et al., NeurIPS '22]

Common hallmarks of reasoning

Model

- Multiple steps & **composition** of logical steps/operations are needed.
- Model is often expected to **extrapolate: OOD and length generalization**.
- **Search & verification** may be needed.

e.g., algorithmic tasks, planning, theorem proving

We want to solve harder tasks than what is seen during training.

- May rely on tools and external components for reasoning.

e.g., calculator, python interpreter, search algorithm

Part 2

Successes of AI: game playing - from Chess to debating



WORLD

Machines Beating Mankind

1 MINUTE READ

BY TARA JOHN X MARCH 17, 2016 6:17 AM EDT

In a milestone for AI, a Google program beat South Korean grand master Lee Sedol at the strategy game Go (left) on March 15. But machines have been mastering board games since the 1970s.

BACKGAMMON

Friday briefing: An AI has finally beaten human poker champions

Unpredictability is key to the Pluribus AI's Texas Hold'em winning streak, some Google Assistant recordings are listened to – and leaked – by humans

All products featured on WIRED are independently selected by our editors. However, we may receive compensation from retailers and/or from purchases of products through these links. [Learn more.](#)



TODAY IN ROBOTS

A Human Beat A.I. in a Debate Tournament, and Robot Takeover Has Been Held Off Another Day

Sure, robots can put a car together with no hands, but can they *debate*?



BY KENZIE BRYANT

FEBRUARY 13, 2019

Successes of AI: advancing science

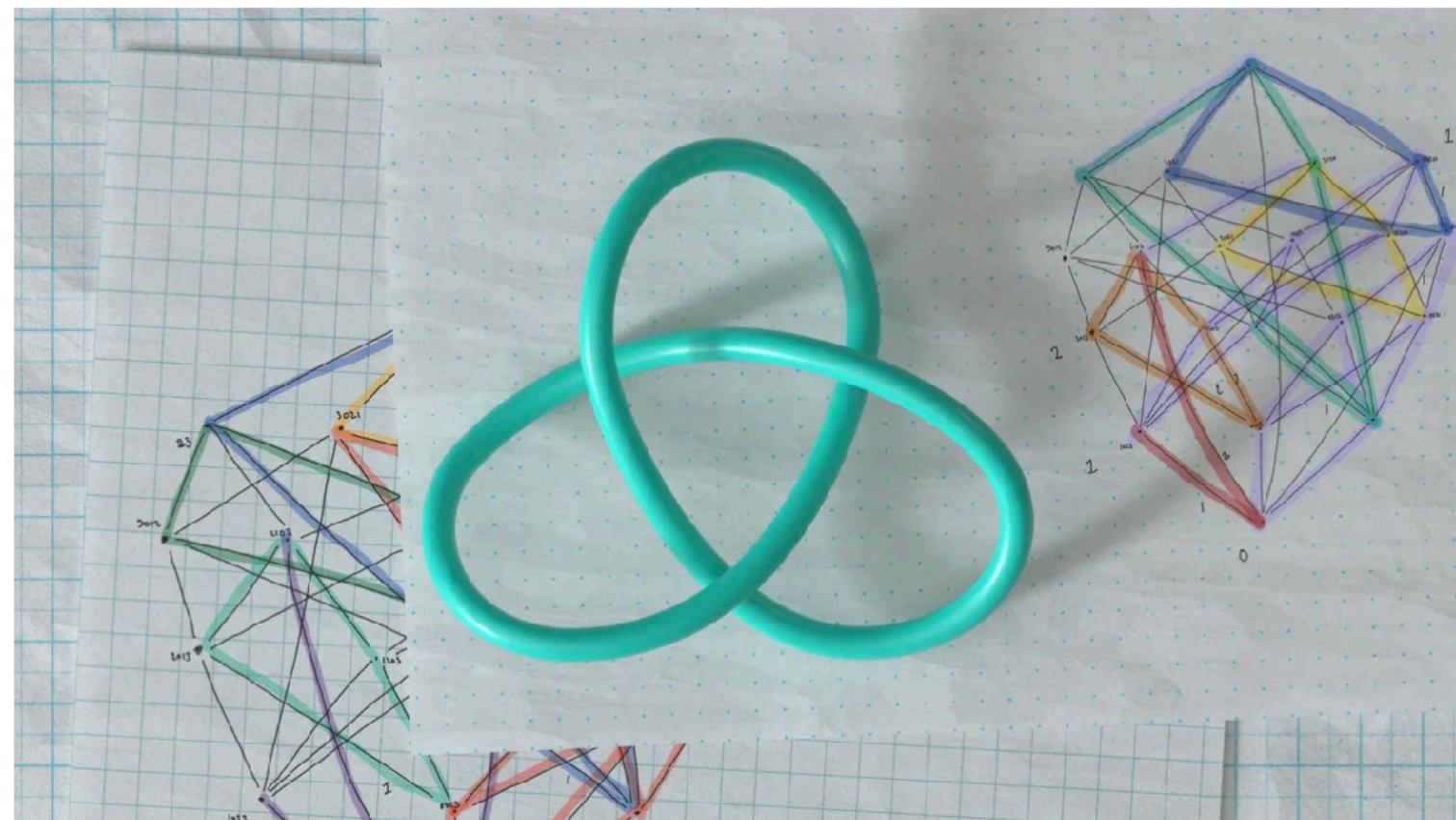
DeepMind cracks 'knot' conjecture that bedeviled mathematicians for decades

News By Stephanie Pappas published December 6, 2021

The artificial intelligence company DeepMind is delving into pure math.



When you purchase through links on our site, we may earn an affiliate commission. [Here's how it works.](#)



A knot and a graph representing the problems that the artificial intelligence program DeepMind tackled. (Image credit: DeepMind)

FunSearch: Making new discoveries in mathematical sciences using Large Language Models

14 DECEMBER 2023

Alhussein Fawzi and Bernardino Romera Paredes

Fig. 4: Result of applying FunSearch to the cap set problem.

a

n	3	4	5	6	7	8
Best known	9	20	45	112	236	496
FunSearch	9	20	45	112	236	512

Numerous AI-powered reasoning tools

- Math/Logic (Lean, Coq)
- Coding (Copilot, Tabnine, CodeWhisperer)
- Writing/Office (Grammarly, Notion AI)
- Science/Research (AlphaFold, Elicit, Semantic Scholar)
- Creative (DALL·E, Midjourney, Runway)

Model failures: common sense questions

DP I have a cup whose bottom is cut off and which has a sealed waterproof top. Can I still use this cup for its intended purpose?

No, you wouldn't be able to use this cup for its intended purpose of holding liquids. With the bottom cut off, any liquid you pour in would immediately flow out through the opening at the base, making it impossible to contain or drink from.

The cup would essentially function more like a funnel or tube rather than a container. Even though the top is sealed and waterproof, without a bottom to contain the liquid, the cup can't fulfill its basic function of holding beverages.

You might be able to repurpose it for other uses - like as a funnel, a planter sleeve (if you put another container inside), or for some kind of craft project - but it wouldn't work as a drinking cup anymore.



   Retry 

Claude can make mistakes. Please double-check responses.

Model failures: symbolic and visual manipulation tasks

QUESTION: Take the last letters of the words in “Daniel Friedman” and concatenate them.

MODEL ANSWER (INCORRECT): The last letter of “Daniel” is “l”. The last letter of “Friedman” is “m”. Concatenating them is “lm”. So the answer is lm. ✘

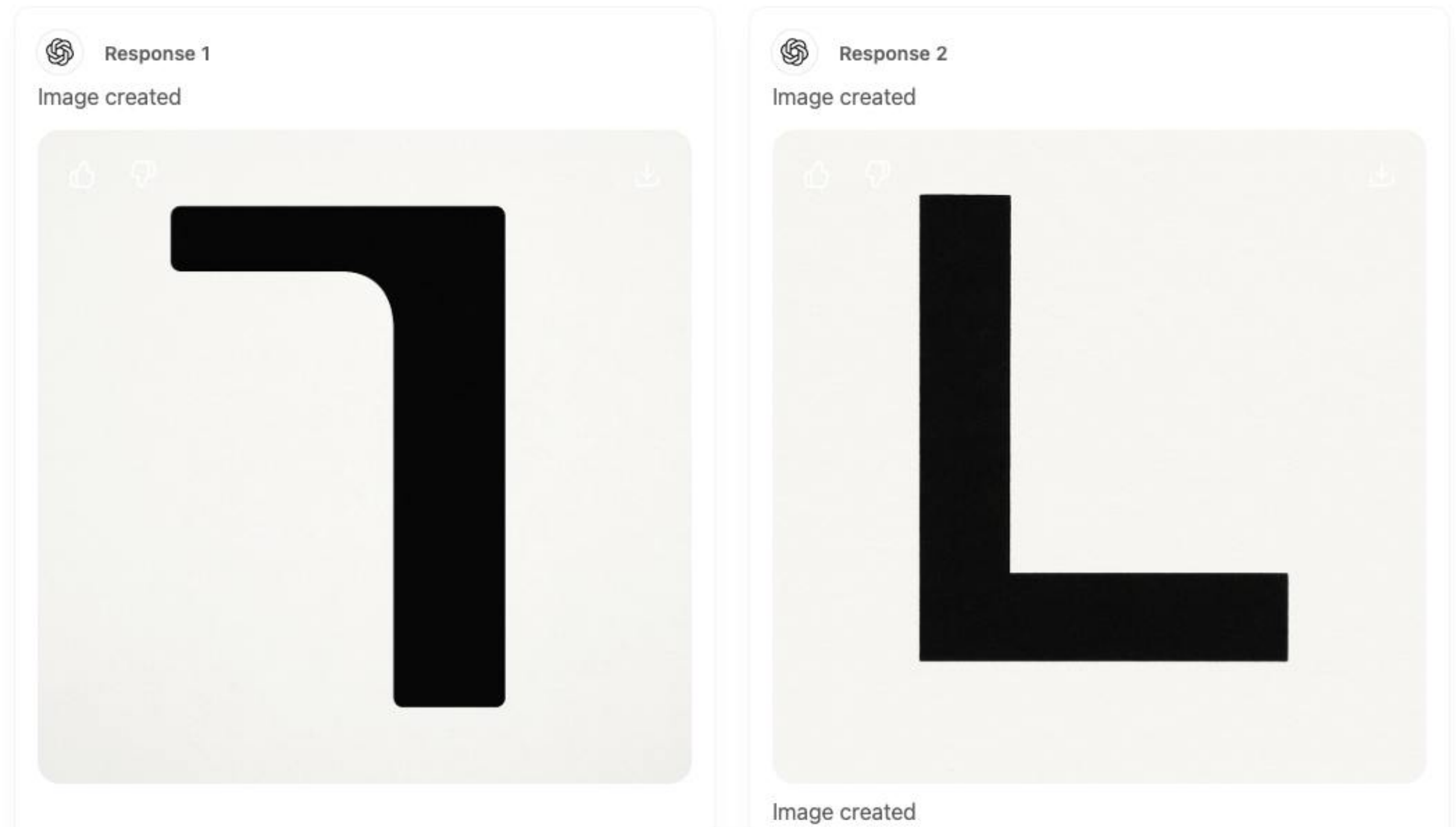
ChatGPT 5

Share

Taken from (Wei et. al., 2023)

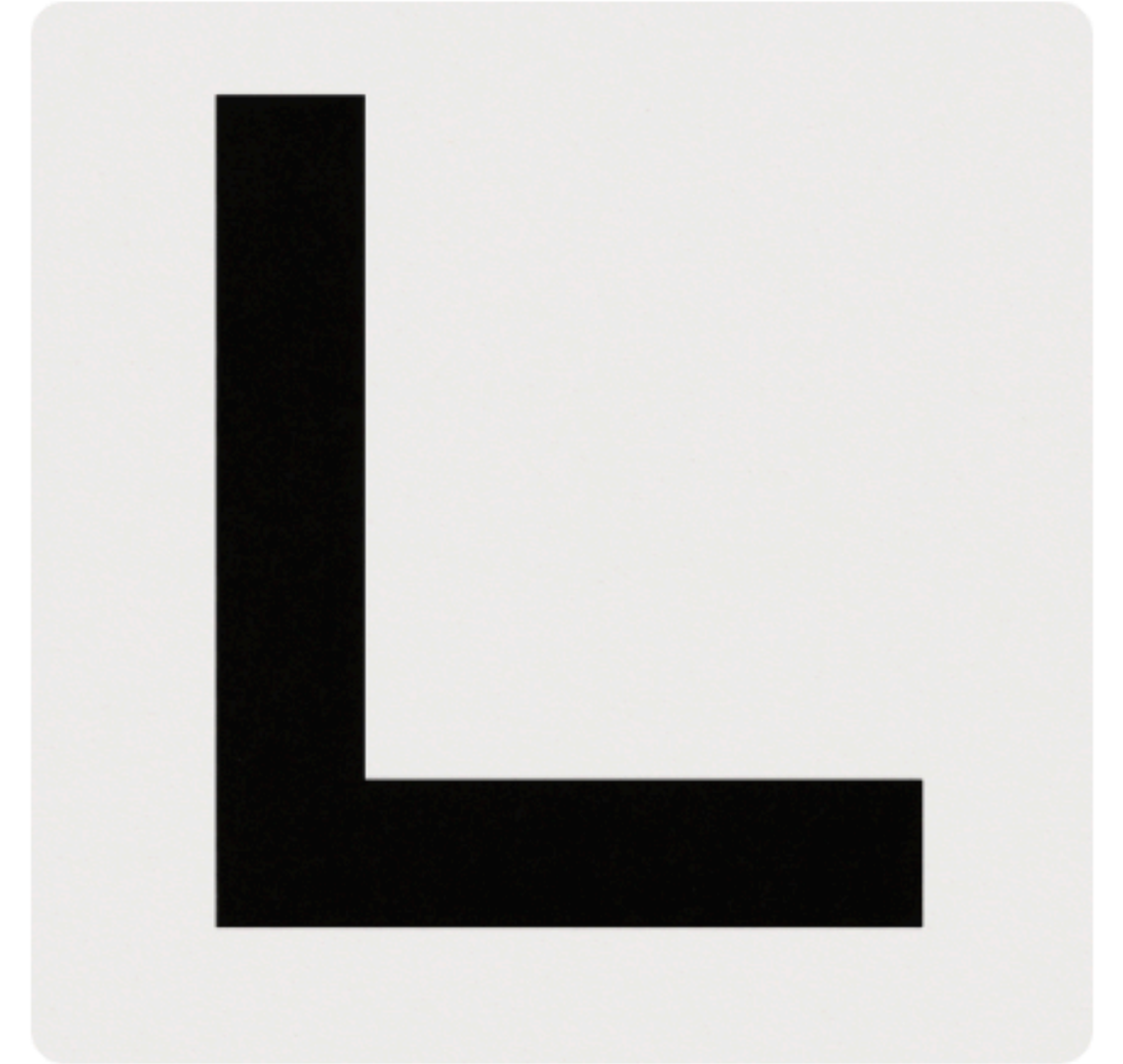
Generate me an image of figure L, flipped upside-down?

You're giving feedback on a new version of ChatGPT.
Which response do you prefer? Responses may take a moment to load.



None is correct!

Model's reasoning is not robust



Imaging you are tracing the perimeter of the following figure with a pen clockwise. How many left turns you have to make to complete a full walk around the figure?

80% accuracy

Thought for 2m 6s

1

Clockwise around this L-shape, every outer corner is a right turn. The only time you turn left is at the single concave (inside) corner where the two bars meet. Hence just **one left turn** for the entire perimeter.

Model's reasoning is not robust



Imaging you are tracing the perimeter of the following figure with a pen clockwise. How many left turns you have to make to complete a full walk around the figure?

Let's carefully analyze the figure and trace its perimeter clockwise.

The shape is essentially a thick L form, like a right-angle block.

Step-by-step tracing clockwise:

1. Start at the top-left corner, moving right along the top edge.
2. At the end of the top edge, turn **right** to move downward.
3. At the end of that downward stroke, turn **left** to move left along the inner edge.
4. At the end of that stroke, turn **left** again to move downward.
5. At the end of that downward stroke, turn **right** to move along the bottom edge (to the left).
6. At the end of that stroke, turn **right** again to go upward along the left edge.
7. Finally, at the top, turn **right** to close the loop back to the start.

Counting left turns:

- Turn 3 → left
- Turn 4 → left
- ✓ Total 2 left turns.

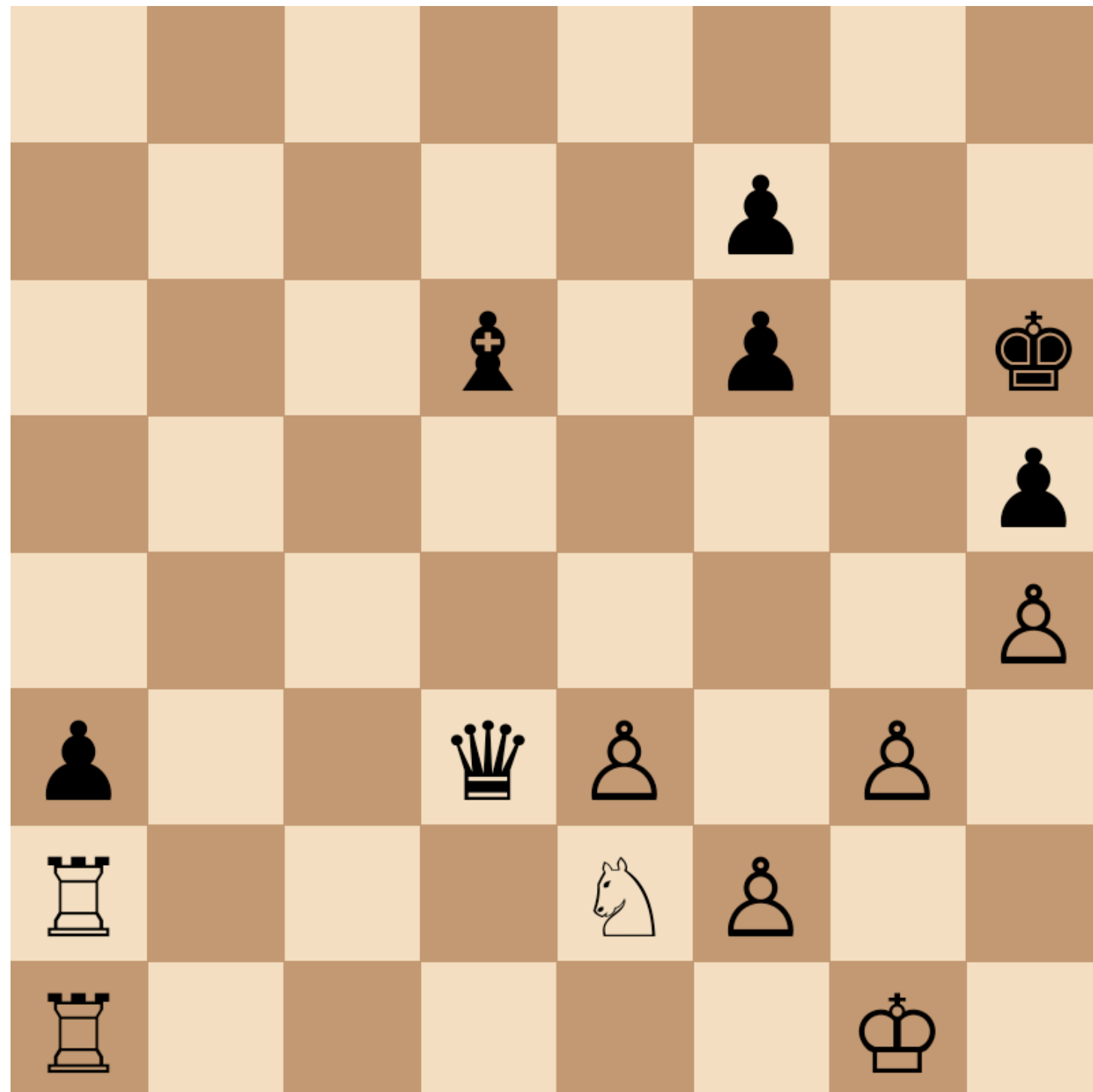
Accuracy drops to 30%

Model can execute sophisticated instructions

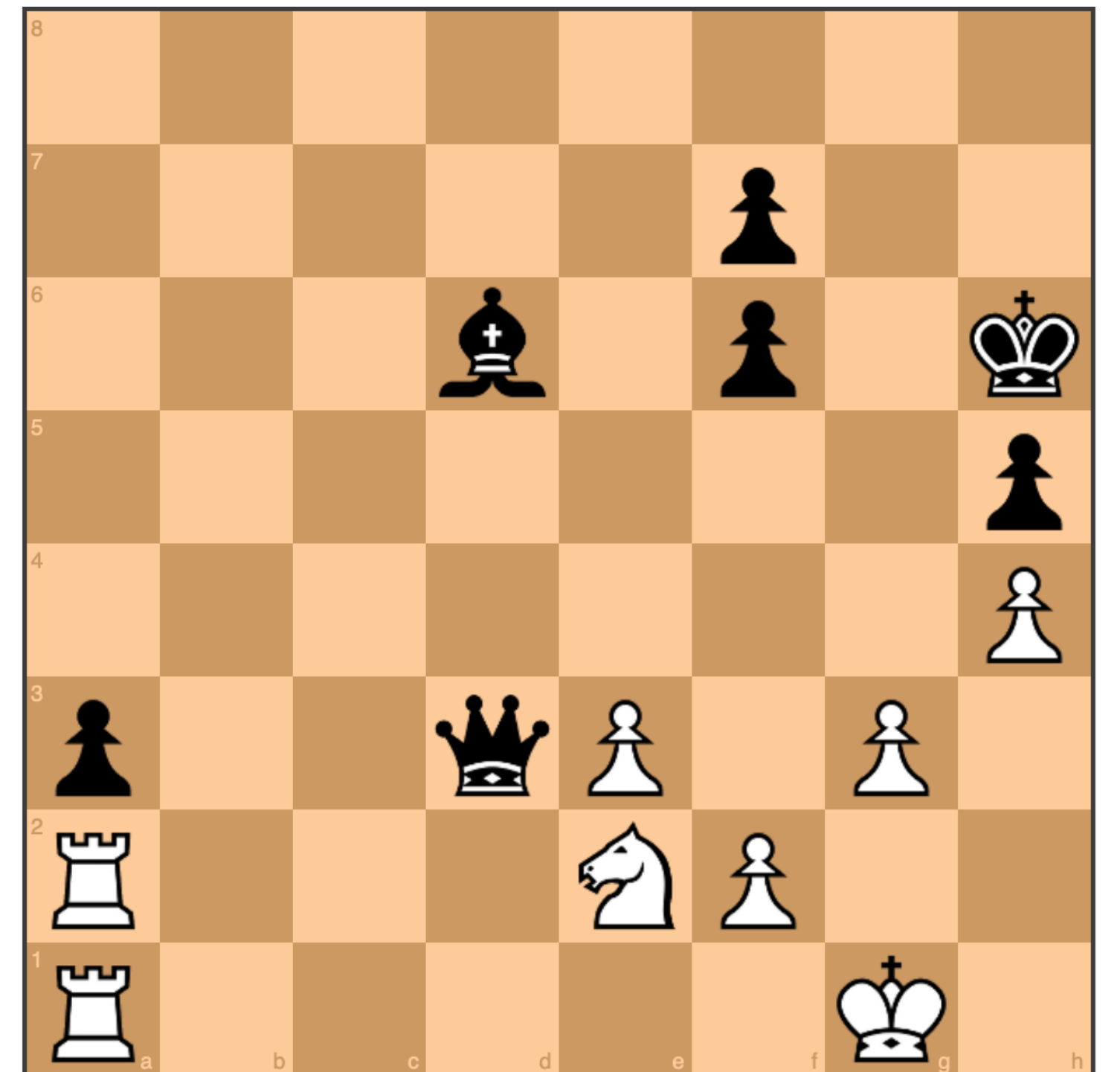
Here is the chess board at the start of the game. Create the image of the board after the following game was played:

1. d4 Nf6 2. Nf3 d5 3. g3 e6 4. Bg2 Be7 5. O-O O-O 6. b3 c5 7. dxc5 Bxc5 8. c4 dxc4 9. Qc2 Qe7 10. Nbd2 Nc6 11. Nxc4 b5 12. Nce5 Nb4 13. Qb2 Bb7 14. a3 Nc6 15. Nd3 Bb6 16. Bg5 Rfd8 17. Bxf6 gxf6 18. Rac1 Nd4 19. Nxd4 Bxd4 20. Qa2 Bxg2 21. Kxg2 Qb7+ 22. Kg1 Qe4 23. Qc2 a5 24. Rfd1 Kg7 25. Rd2 Rac8 26. Qxc8 Rxc8 27. Rxc8 Qd5 28. b4 a4 29. e3 Be5 30. h4 h5 31. Kh2 Bb2 32. Rc5 Qd6 33. Rd1 Bxa3 34. Rxb5 Qd7 35. Rc5 e5 36. Rc2 Qd5 37. Rdd2 Qb3 38. Ra2 e4 39. Nc5 Qxb4 40. Nxe4 Qb3 41. Rac2 Bf8 42. Nc5 Qb5 43. Nd3 a3 44. Nf4 Qa5 45. Ra2 Bb4 46. Rd3 Kh6 47. Rd1 Qa4 48. Rda1 Bd6 49. Kg1 Qb3 50. Ne2 Qd3

Generated image



Correct position



But it heavily relies on external tools!

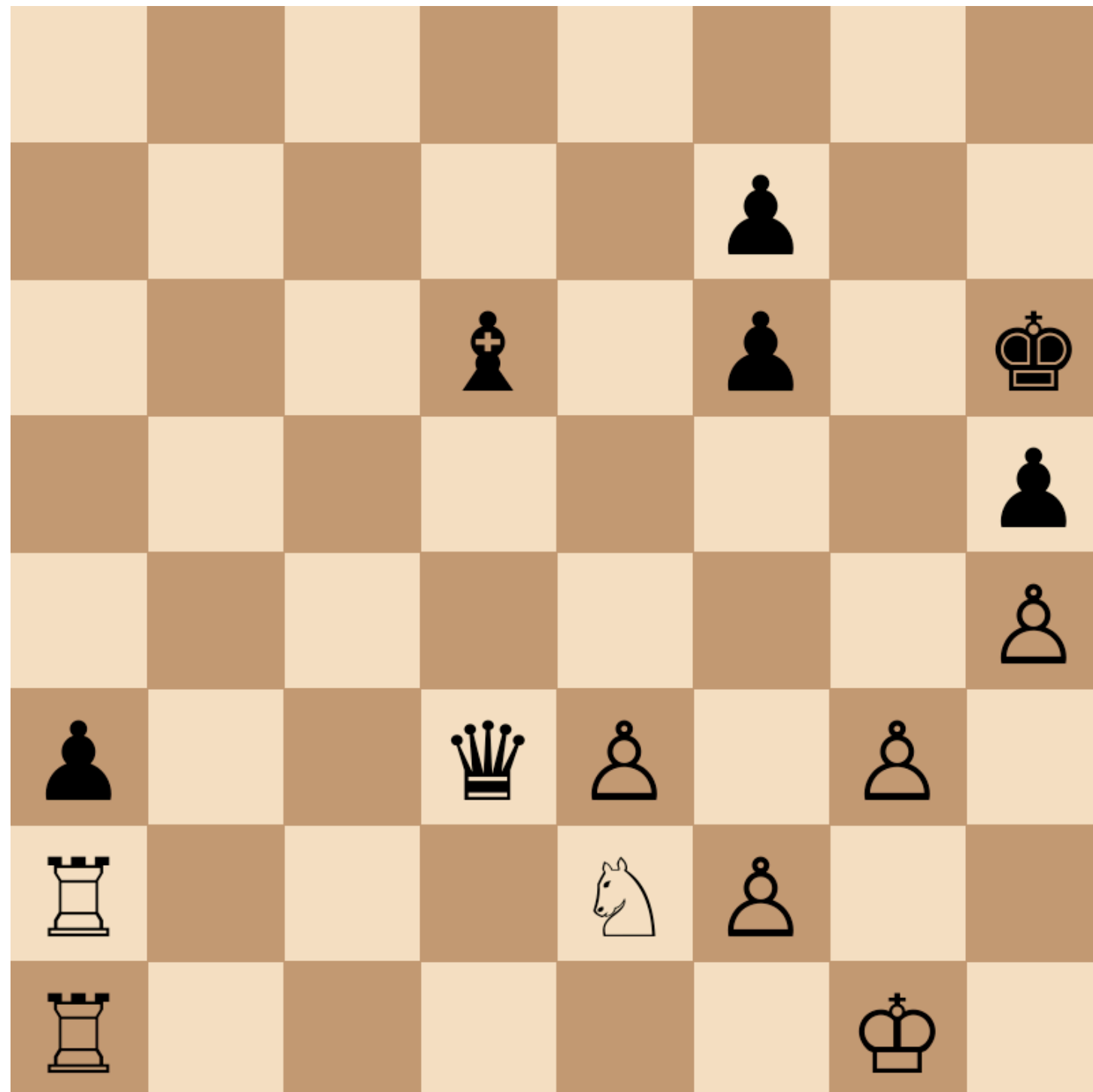
Here is the chess board at the start of the game. Create the image of the board after the following game was played:

1. d4 Nf6 2. Nf3 d5 3. g3 e6 4. Bg2 Be7 5. O-O O-O 6. b3 c5 7. dxc5 Bxc5 8. c4 dxc4 9. Qc2 Qe7 10. Nbd2 Nc6 11. Nxc4 b5 12. Nce5 Nb4 13. Qb2 Bb7 14. a3 Nc6 15. Nd3 Bb6 16. Bg5 Rfd8 17. Bxf6 gxf6 18. Rac1 Nd4 19. Nxd4 Bxd4 20. Qa2 Bxg2 21. Kxg2 Qb7+ 22. Kg1 Qe4 23. Qc2 a5 24. Rfd1 Kg7 25. Rd2 Rac8 26. Qxc8 Rxc8 27. Rxc8 Qd5 28. b4 a4 29. e3 Be5 30. h4 h5 31. Kh2 Bb2 32. Rc5 Qd6 33. Rd1 Bxa3 34. Rxb5 Qd7 35. Rc5 e5 36. Rc2 Qd5 37. Rdd2 Qb3 38. Ra2 e4 39. Nc5 Qxb4 40. Nxe4 Qb3 41. Rac2 Bf8 42. Nc5 Qb5

Thought for 6m 24s

Analyzed ^

Generated image



python

Always show details | [Copy code](#)

```
# Create and display the final chess position as an image
```

```
from PIL import Image, ImageDraw, ImageFont
```

```
# Board settings
```

```
square_size = 100
```

```
board_size = square_size * 8
```

```
light = (240, 217, 181) # classic light square
```

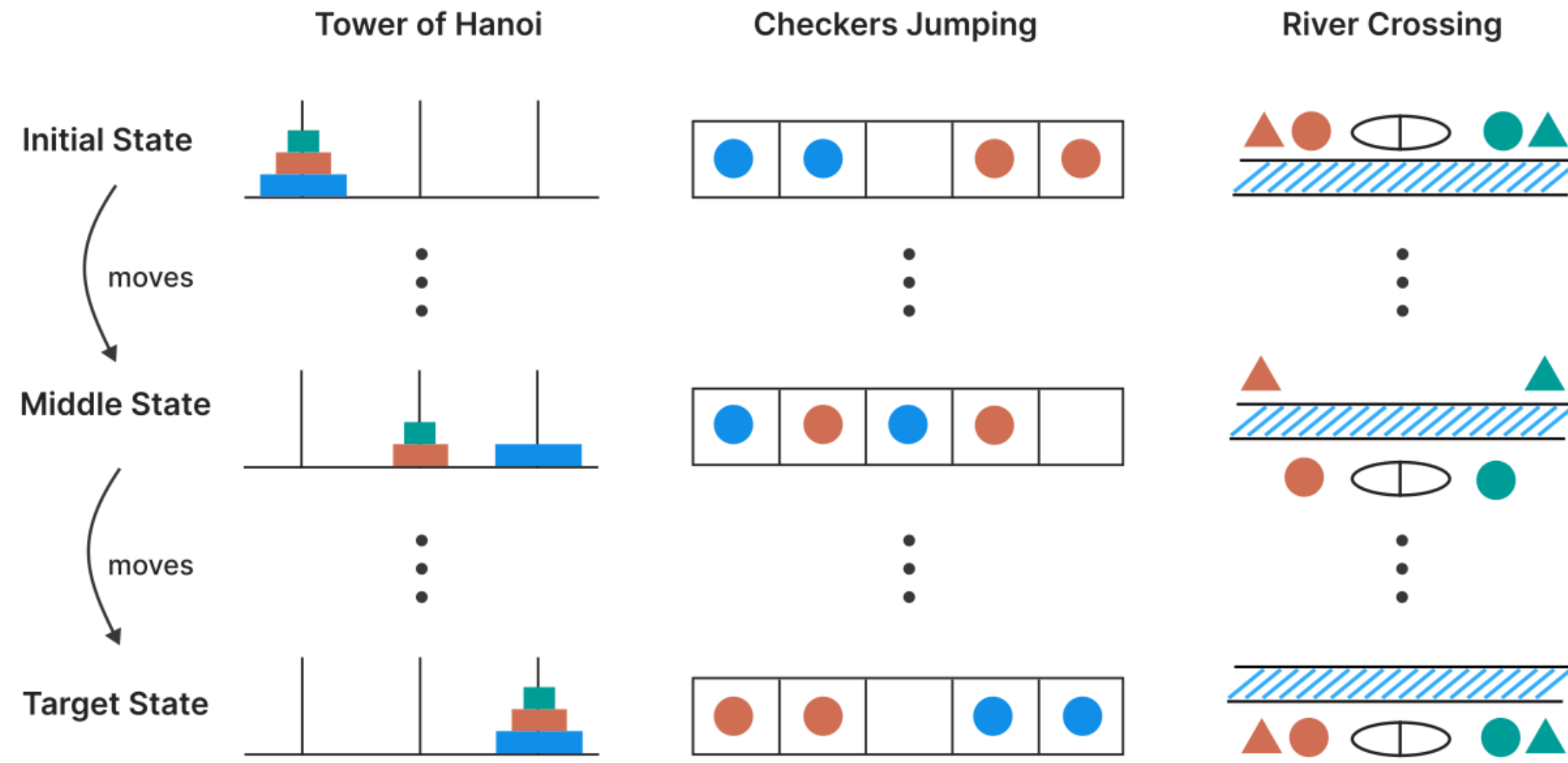
```
dark = (181, 136, 99) # classic dark square
```

```
img = Image.new("RGB", (board_size, board_size), light)
```

```
draw = ImageDraw.Draw(img)
```

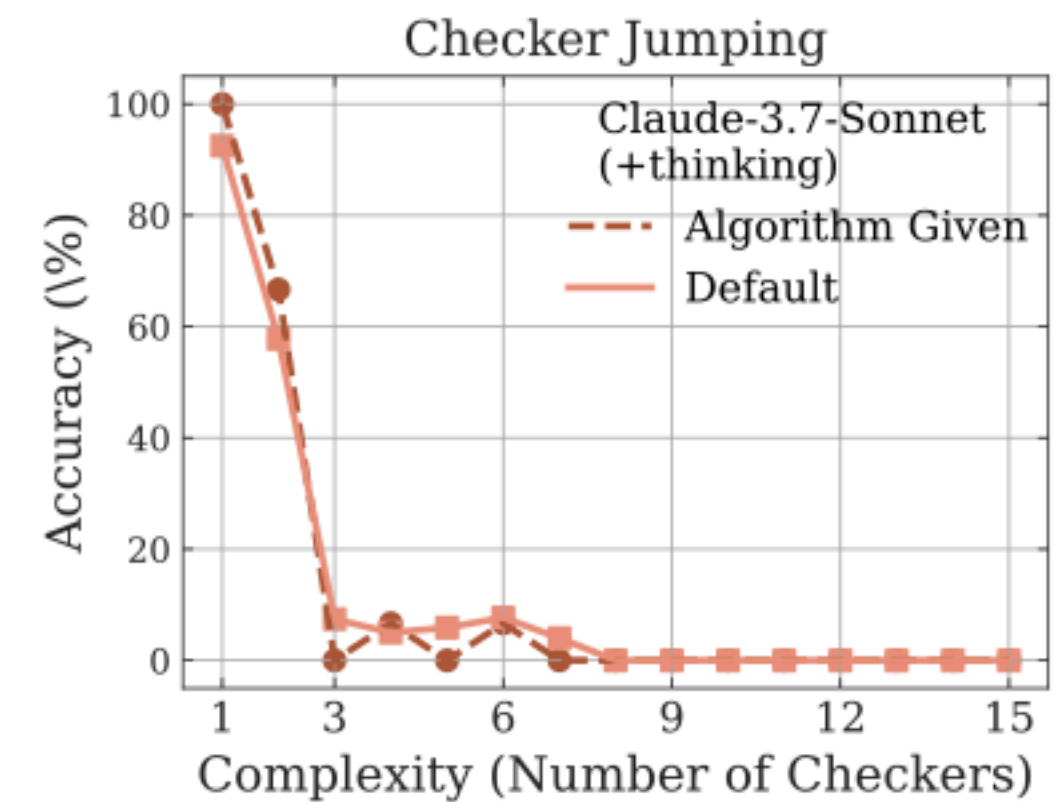
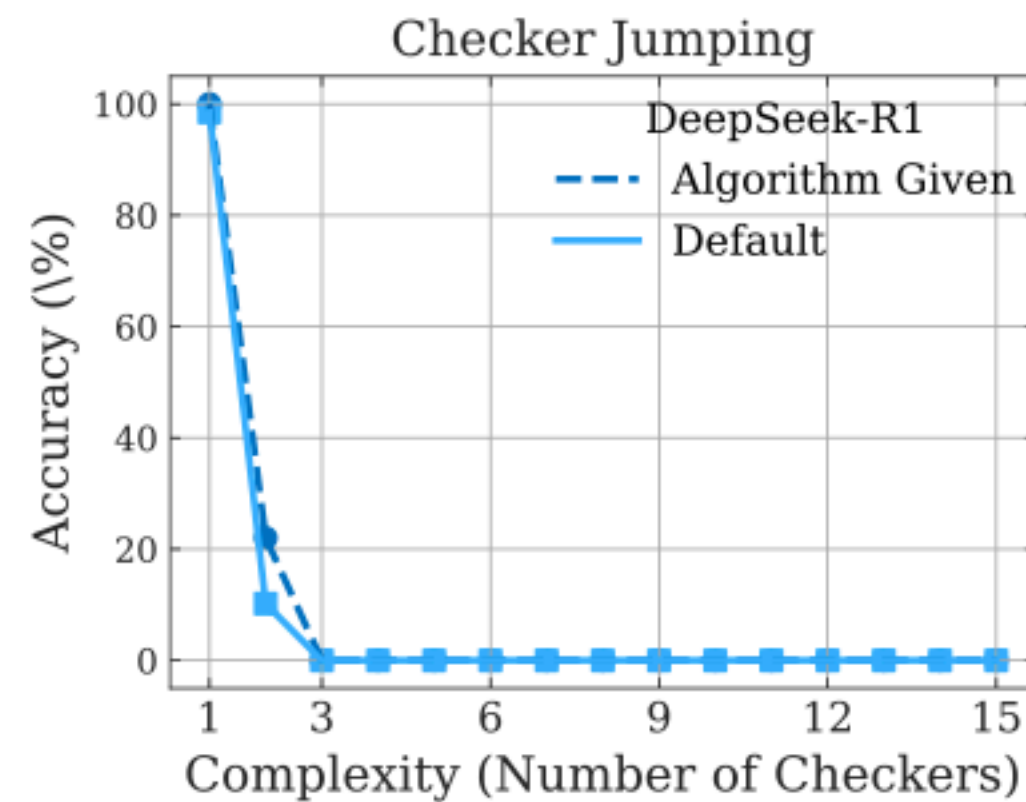
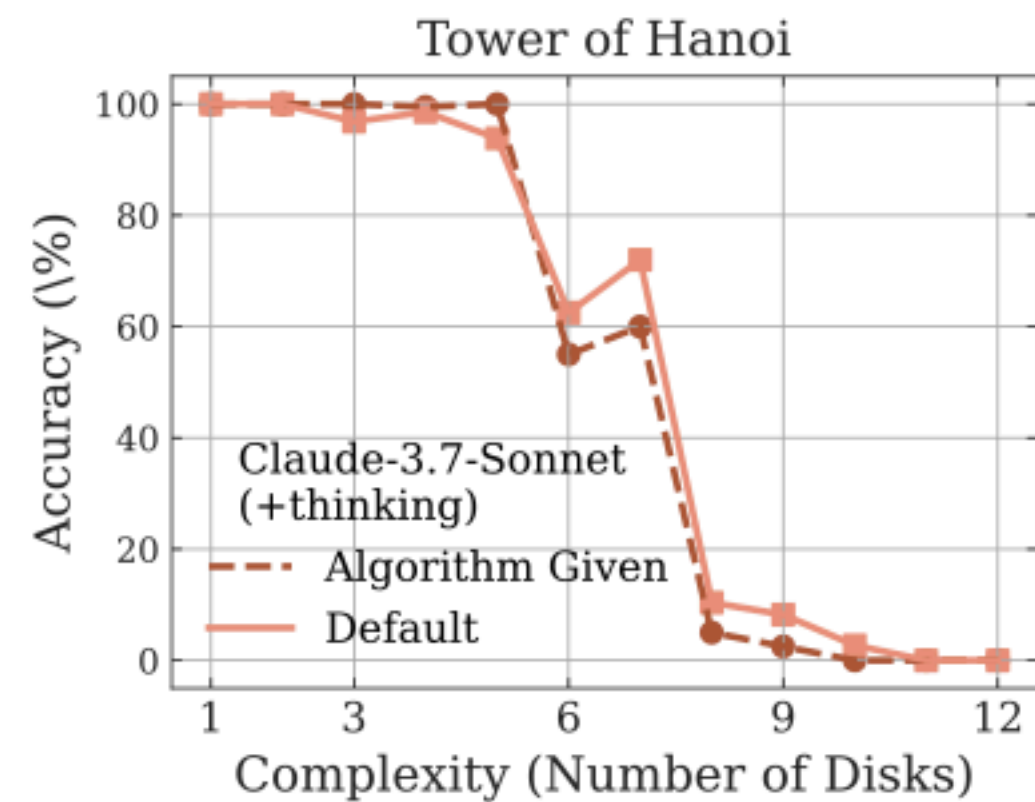
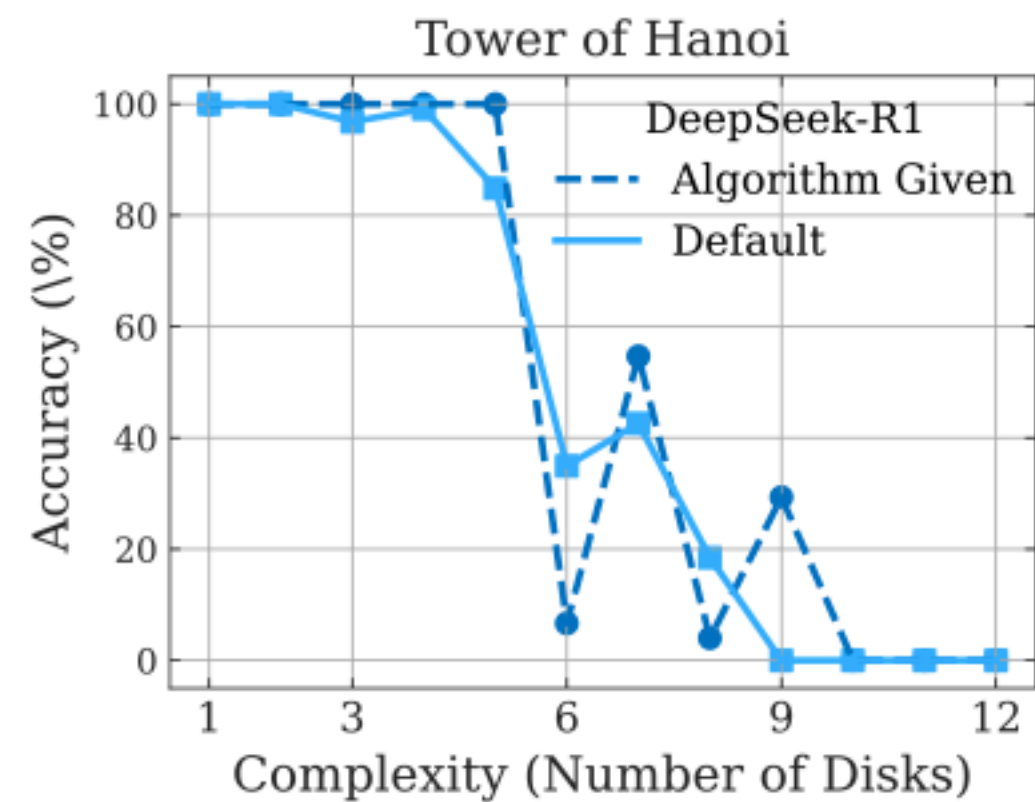
Models without tool calling

Model's output



```
<answer>
moves = [
  {'move': ['R', 1, 2], 'state': ['R', '_', 'R', 'B', 'B']},
  {'move': ['B', 3, 1], 'state': ['R', 'B', 'R', '_', 'B']},
  {'move': ['B', 4, 3], 'state': ['R', 'B', 'R', 'B', '_']},
  {'move': ['R', 2, 4], 'state': ['R', 'B', '_', 'B', 'R']},
  {'move': ['R', 0, 2], 'state': ['_', 'B', 'R', 'B', 'R']},
  {'move': ['B', 1, 0], 'state': ['B', '_', 'R', 'B', 'R']},
  {'move': ['B', 3, 1], 'state': ['B', 'B', 'R', '_', 'R']},
  {'move': ['R', 2, 3], 'state': ['B', 'B', '_', 'R', 'R']}
]
</answer>
```

Taken from "The Illusion of Thinking ..." (2025)



Part 3

Everyone talks about reasoning

The New York Times

Artificial Intelligence > | What is Vibecoding? | OpenAI and Musk | Paris A.I. Summit | A Look at OpenAI's Operator

What to Know About DeepSeek and How It Is Upending A.I.

How did a little-known Chinese start-up cause the markets and U.S. tech giants to quake? Here's what to know.

Artificial Intelligence > | What is Vibecoding? | OpenAI and Musk | Paris A.I. Summit | A Look at OpenAI's Operator | Quiz

OpenAI Unveils New ChatGPT That Can Reason Through Math and Science

Driven by new technology called OpenAI o1, the chatbot can test various strategies and try to identify mistakes as it tackles complex tasks.

Listen to this article · 4:39 min [Learn more](#) [Share full article](#) [Bookmark](#) [Comments 177](#)



Reuters


World > | Business > | Markets > | Sustainability > | Legal > | Breakingviews > | Technology > | Investigations

Microsoft developing AI reasoning models to compete with OpenAI, The Information reports

By Reuters

March 7, 2025 8:51 AM PST · Updated 5 days ago

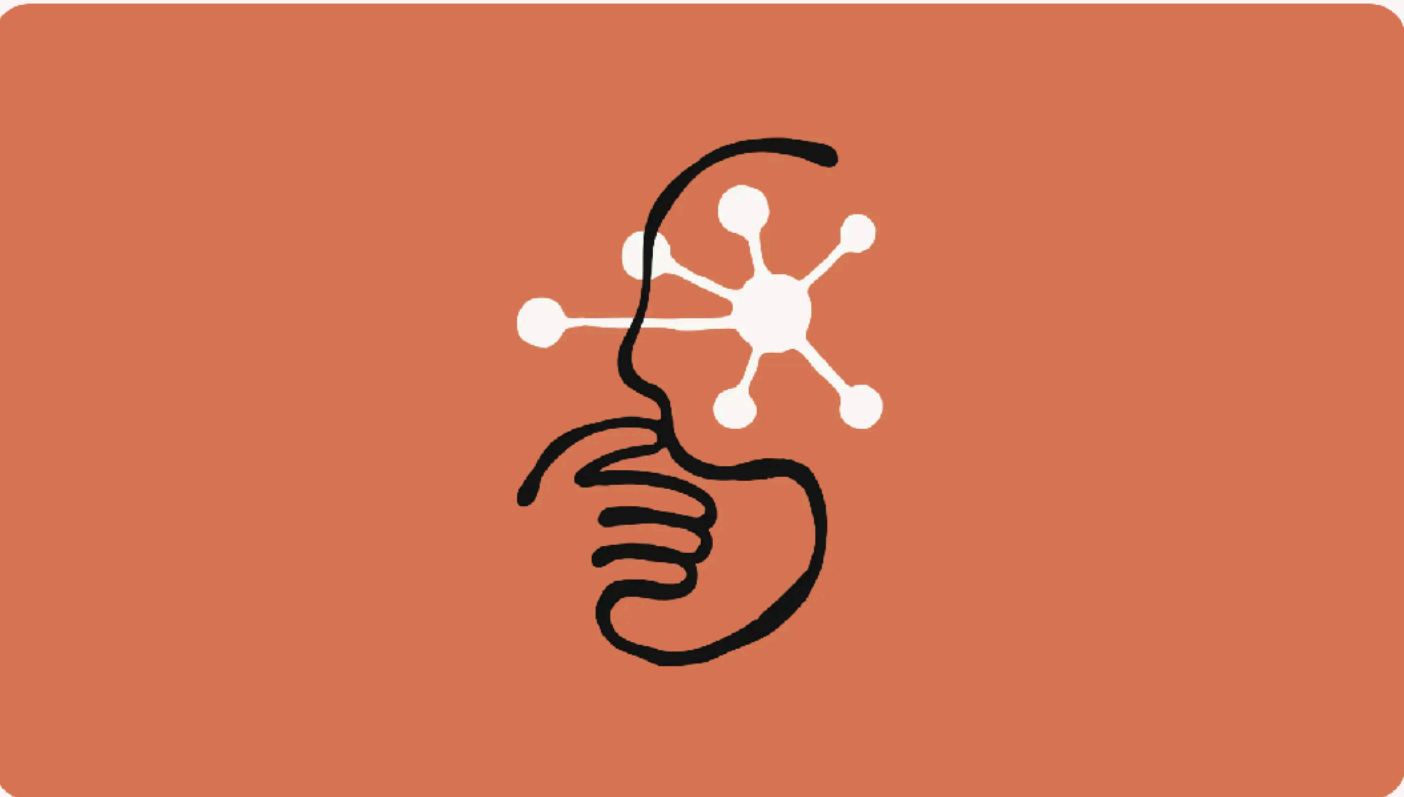
578



[1/2] Microsoft logo is seen in this illustration taken February 16, 2025. REUTERS/Dado Ruvic/Illustration/File Photo [Purchase Licensing Rights](#)

Claude 3.7 Sonnet and Claude Code

Feb 24, 2025 · 5 min read



Today, we're announcing Claude 3.7 Sonnet¹, our most intelligent model to date and the first hybrid reasoning model on the market.

What is reasoning in AI?

An approach to solving tasks by decomposing them into manageable steps

An approach to solving tasks by searching and selecting solution paths more effectively

An approach to solving logical and mathematical tasks

A learning process to reach more dynamic & efficient generalization

what is reasoning?

Reasoning is essentially the process of using conscious thought to draw conclusions, make judgments, or solve problems. It involves applying logic, facts, and evidence to form rational opinions or decisions. Here's a breakdown of key aspects:

- **Core Function:**

- Reasoning is about using our mental faculties to understand and make sense of information.
- It's how we move from existing knowledge to new insights.

- **Key Components:**

- **Logic:** The principles that guide valid reasoning.
- **Inference:** Drawing conclusions based on evidence or assumptions.
- **Analysis:** Breaking down complex information into smaller, more manageable parts.
- **Evaluation:** Assessing the validity and reliability of information.

- **Types of Reasoning:**

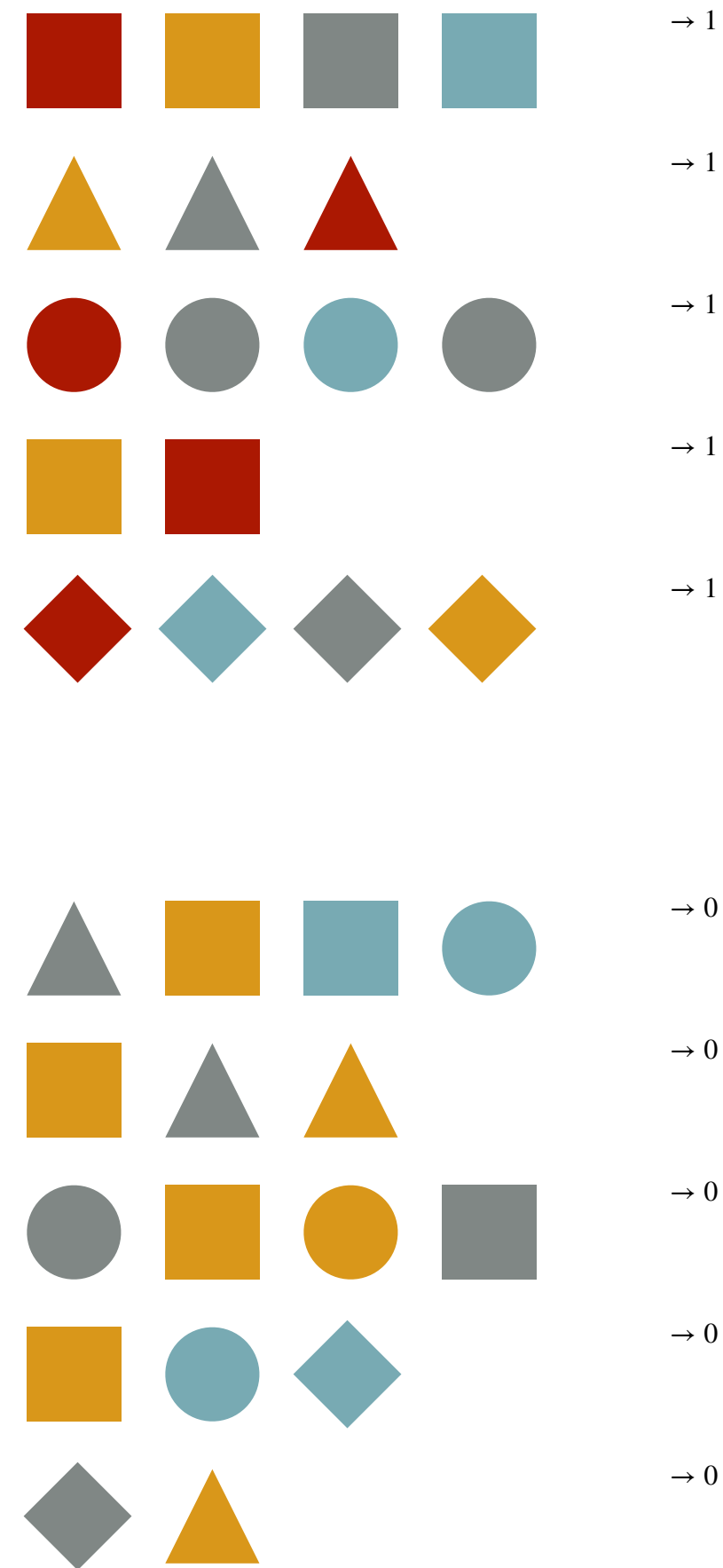
- **Deductive Reasoning:** Moving from general principles to specific conclusions (e.g., "All humans are mortal. Socrates is human. Therefore, Socrates is mortal").
- **Inductive Reasoning:** Moving from specific observations to general conclusions (e.g., "Every time I eat peanuts, I sneeze. Therefore, I might be allergic to peanuts").
- **Abductive Reasoning:** Forming a likely explanation for an observation (e.g., "The grass is wet. It must have rained").

- **Applications:**

- Reasoning is fundamental to:
 - Problem-solving
 - Decision-making
 - Critical thinking
 - Scientific inquiry
 - Everyday life

In essence, reasoning is the mental process that allows us to think critically and make informed judgments.

Toy task



Some equally good learners in the eye of train-loss:

1. "Are the shapes all the same?"

2. "Is there a red shape?"

3. "Are the first and last shapes the same?"

4. "Is there 1 red square, 1 yellow square, 1 gray square and 1 blue square, or 1 yellow triangle, 1 gray triangle, 1 red triangle,

...,

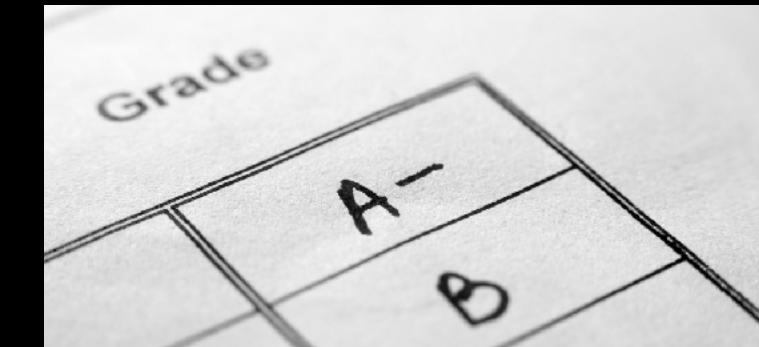
or 1 red diamond, 1 blue diamond, 1 gray diamond, 1 yellow diamond, or a sequence only modifying 1 color or removing 1 element from the above?"

"shortcuts"
-> oversimplified

"memorizers"
-> overcomplicated

Has the model learned?

For long we measured this with a static value:
test error on a benchmark



One may achieve low test error in different ways...
E.g., patching many memorized cases.
This is not the most **efficient** and **dynamic** way.

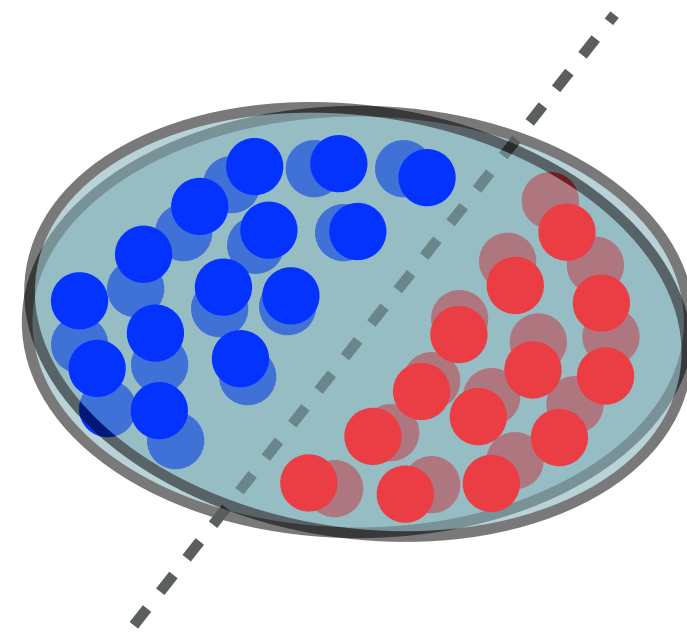


"Reasoning": how efficient and dynamic the generalization is.

Efficiency and robustness

If test and train distributions **match** well enough (for the target sensitivity and tolerated gen. error) then shortcuts and memorizers are not necessarily problematic (e.g., common text applications).

- Shortcuts may simply be good solutions in this regime
- Memorizers still have the issue of **inefficiency** (sample complexity, model size)

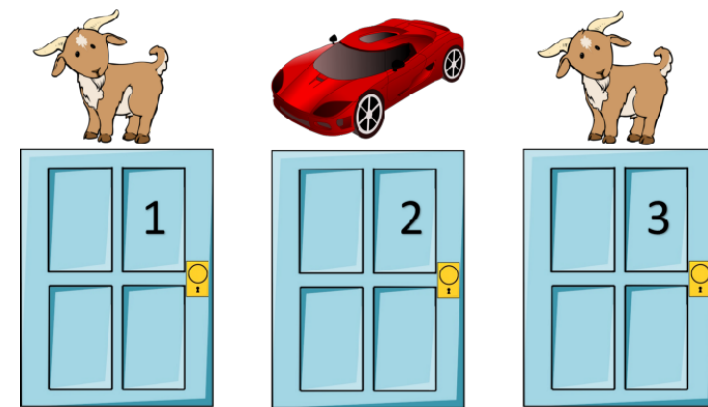


Efficiency and robustness

If test and train distributions **match** well enough (for the target sensitivity and tolerated gen. error) then shortcuts and memorizers are not necessarily problematic (e.g., common text applications).

- Shortcuts may simply be good solutions in this regime
- Memorizers still have the issue of **inefficiency** (sample complexity, model size)

-> **With more mismatch/sensitivity:** both can be dramatic (e.g., reasoning tasks, counter-factuals)



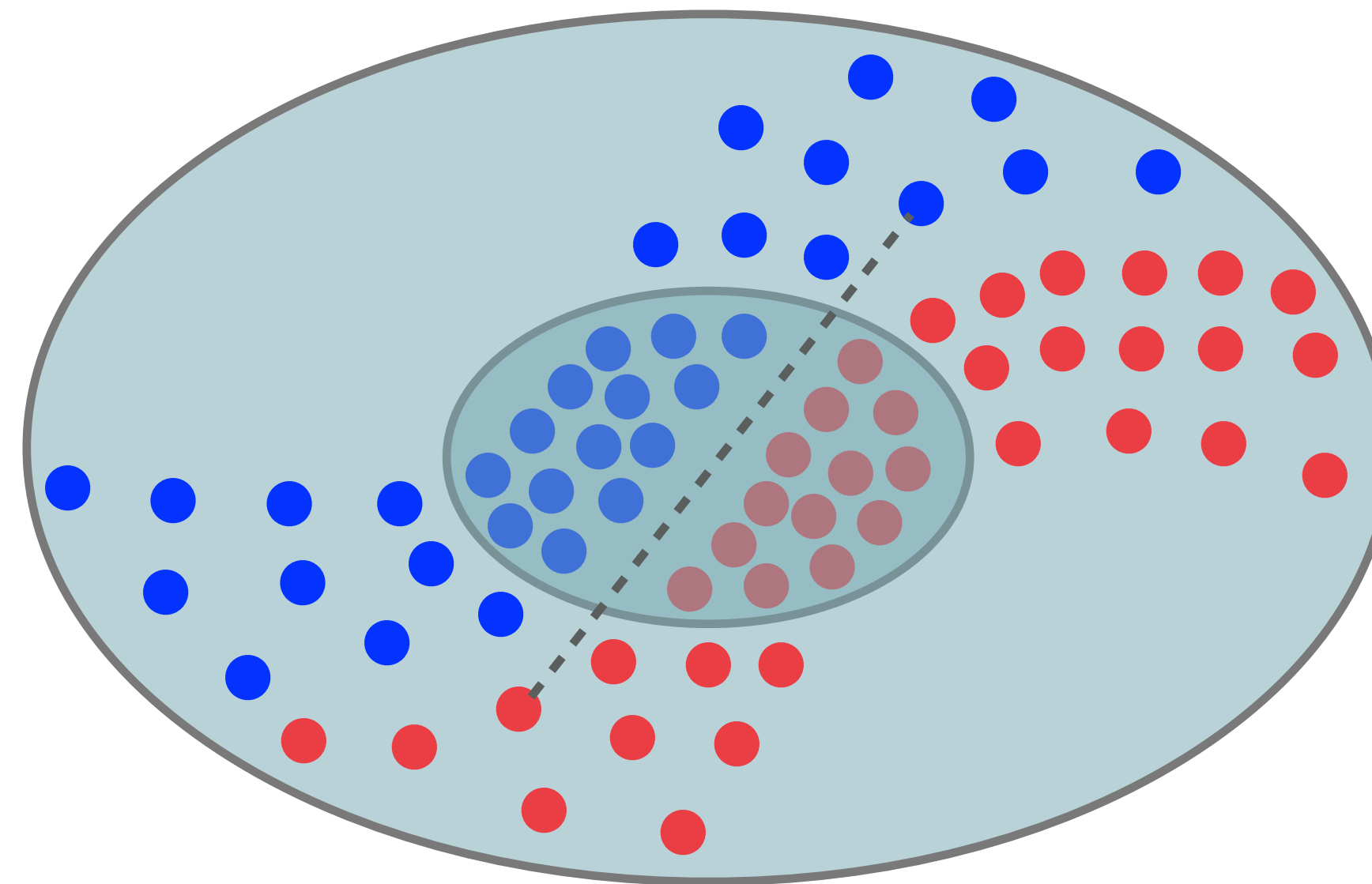
You
In a variant of the Monty Hall problem, one door hides a car and two doors hide goats. I am asked to pick a door in order to win what's hiding behind it. After picking a door, I find the car, but the host opens a second door with a goat and asks me whether I would like to change my door. Should I change door in order to maximize my chance of winning the car?

ChatGPT
Yes, you should change your door. This variant of the Monty Hall problem is similar to the original version, where switching doors after the host reveals a goat increases your chances of winning the car.

Here's why:

When you initially pick a door, there's a 1/3 chance you picked the door with the car and a 2/3 chance you picked a door with a goat. When the host reveals a door with a goat that you didn't pick, it doesn't change the fact that there's still a 1/3 chance the car is behind your original door and a 2/3 chance it's behind one of the other two doors.

By switching doors, you effectively transfer your initial 1/3 chance of picking the car to the remaining door that you didn't initially choose. Therefore, switching doors gives you a 2/3 chance of winning the car, whereas sticking with your original choice only gives you a 1/3 chance.

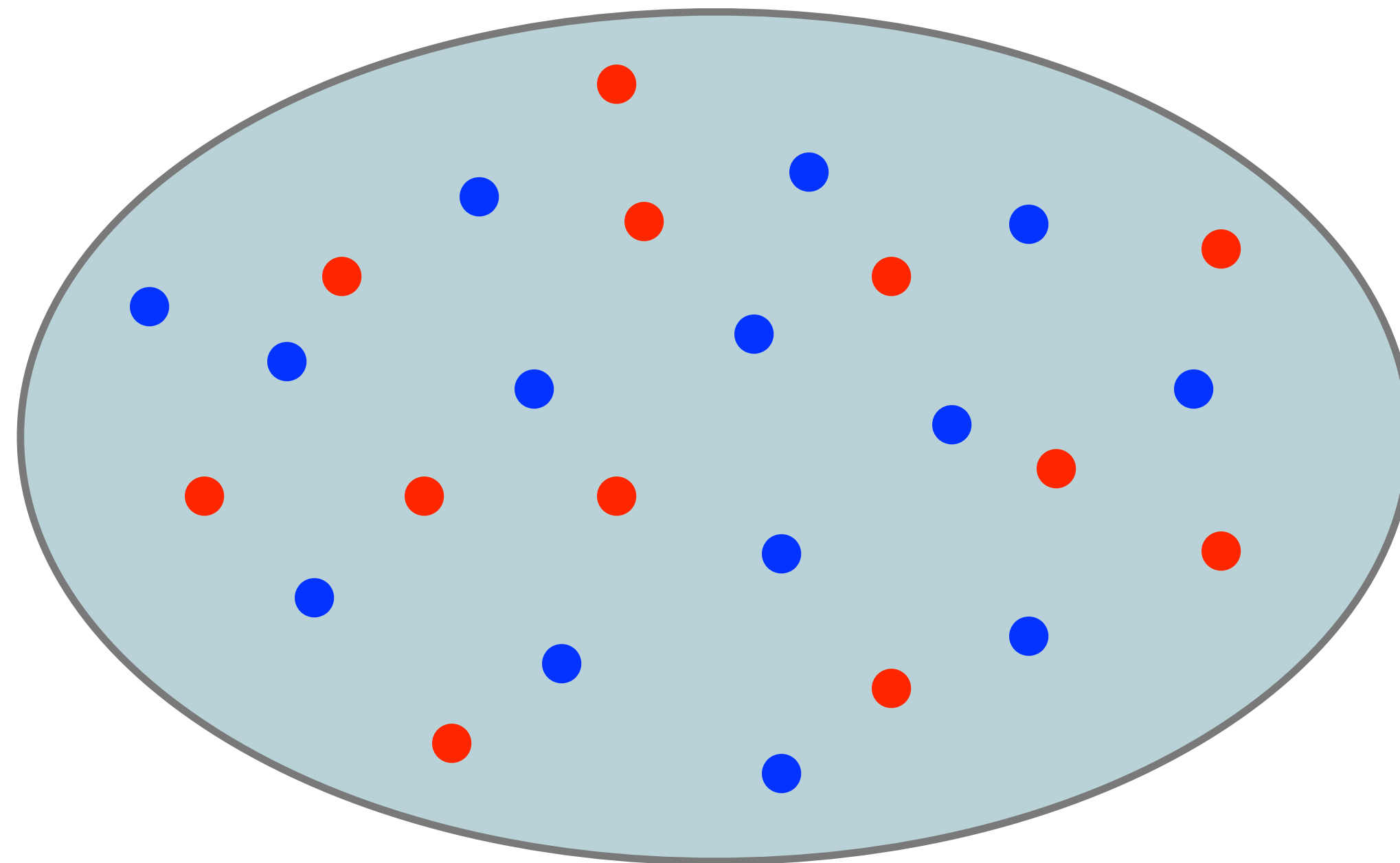


Generalizing OOD (extrapolation) :
rely on composing when possible
rather than memorizing everything

Generalizing OOD cannot be
expected -even with lots of data-
if the out-distribution is not
manifested in the data.

Extrapolation

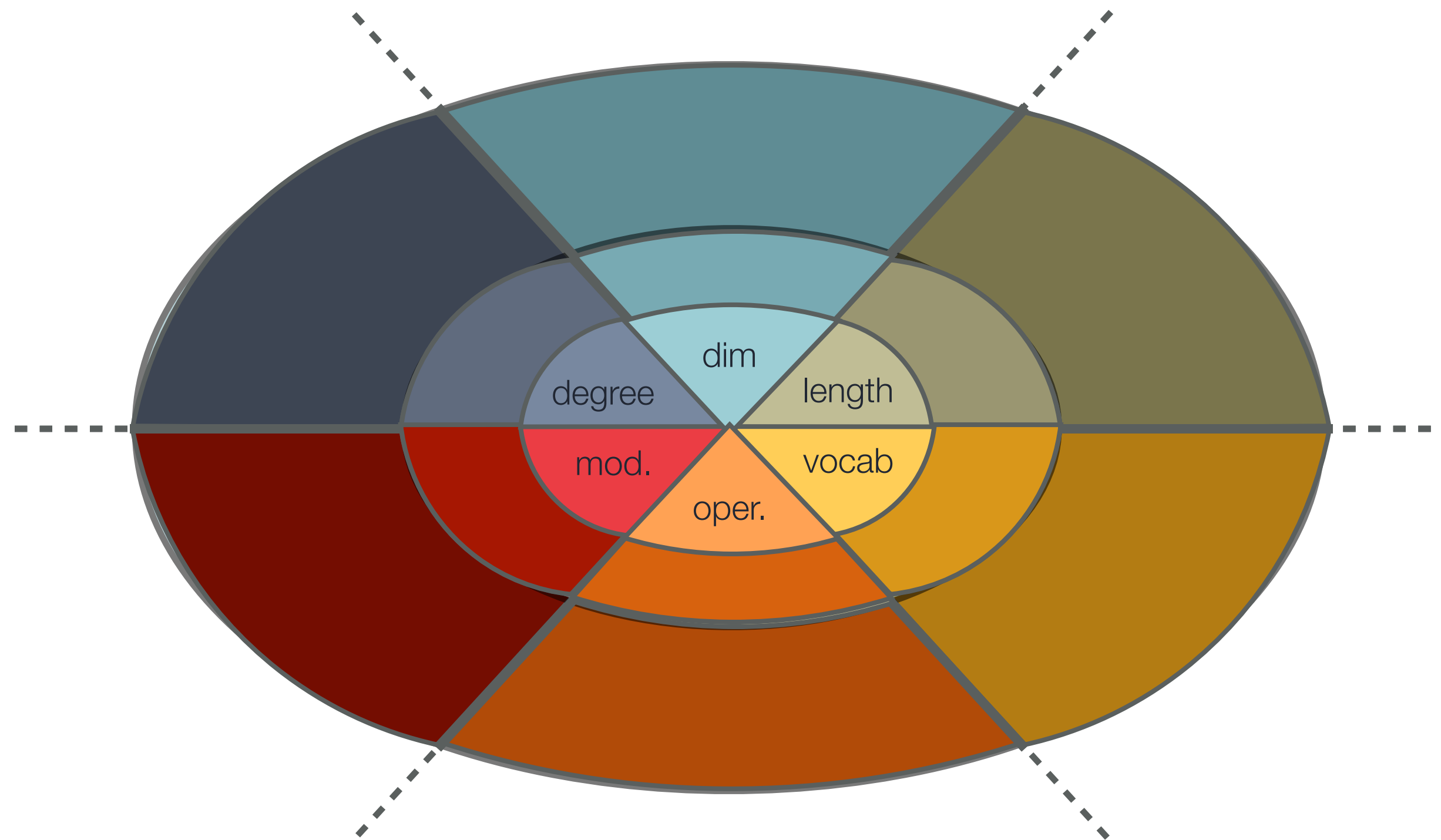
The out-distribution may not be explicitly manifested in the train data but could be implicitly brought up.



Extrapolation

The out-distribution may not be explicitly manifested in the train data but could be implicitly brought up.

1. Structure the data according to relevant complexity measures

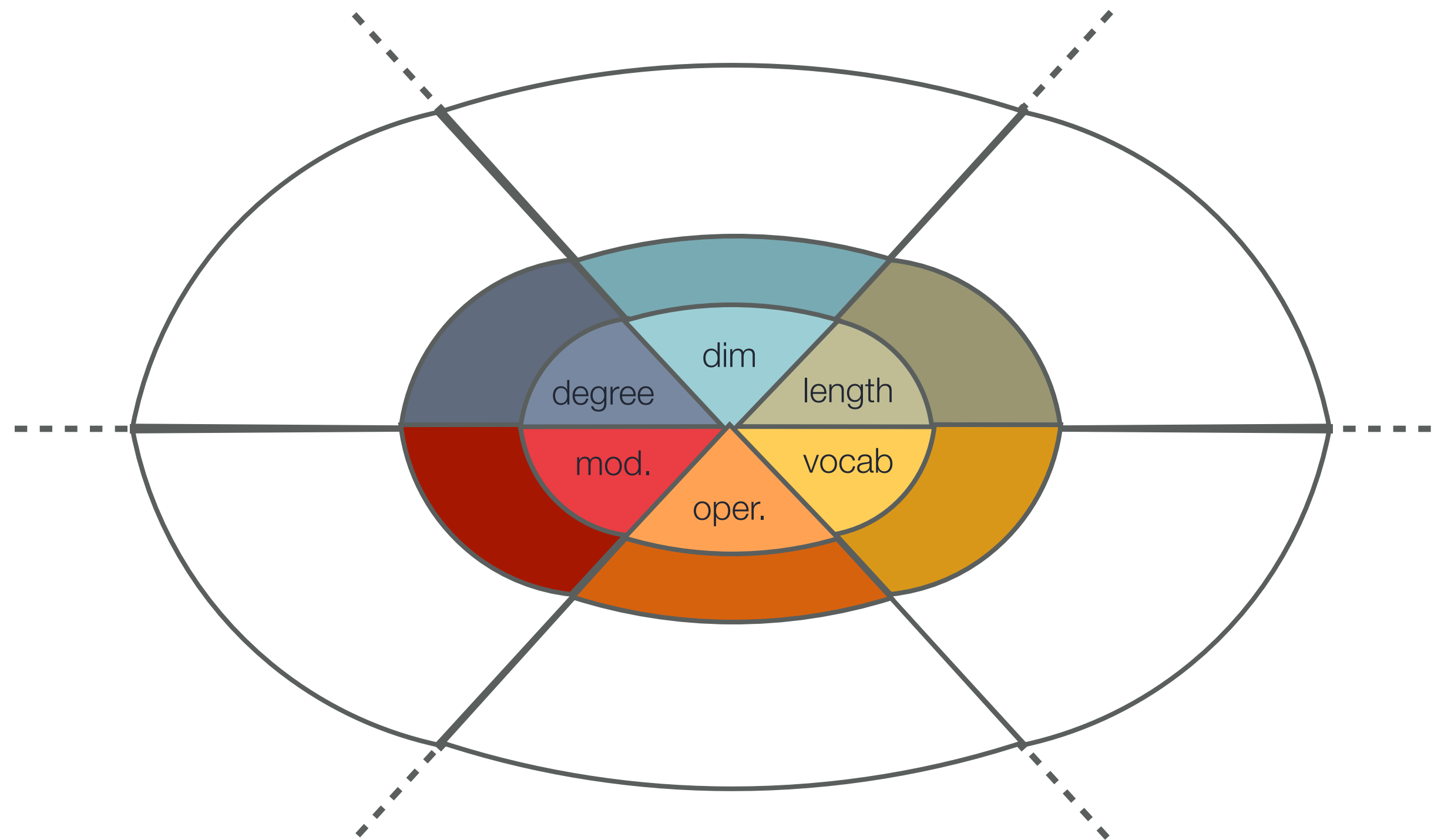


Extrapolation

The out-distribution may not be explicitly manifested in the train data but could be implicitly brought up.

1. Structure the data according to relevant complexity measures
2. Mask domains appropriately and come up with learners that extrapolate well on these.

This gives hope that the extrapolation can “propagate”.

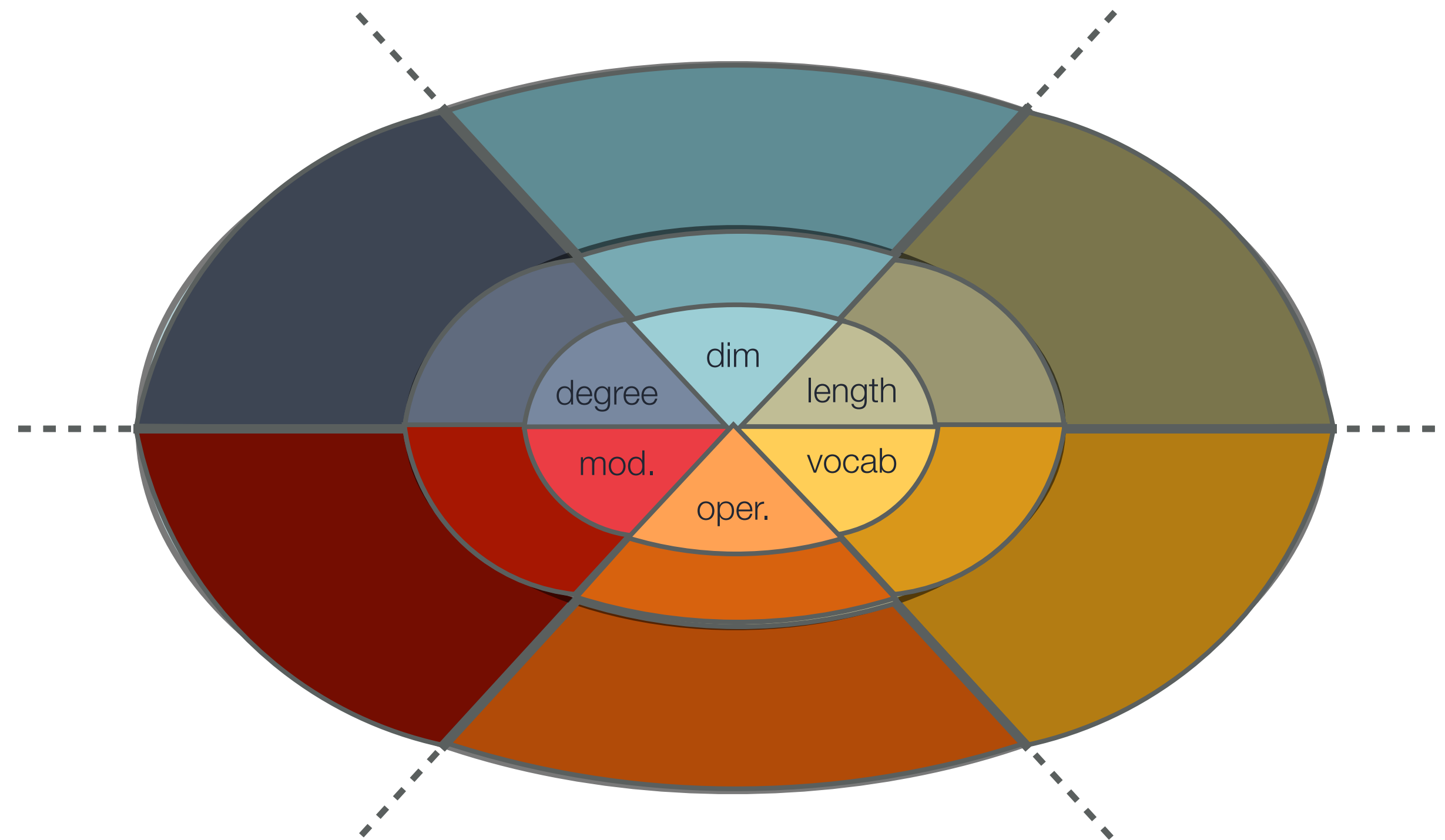


Extrapolation

The out-distribution may not be explicitly manifested in the train data but could be implicitly brought up.

1. Structure the data according to relevant complexity measures
2. Mask domains appropriately and come up with learners that extrapolate well on these.

This gives hope that the extrapolation can “propagate”.

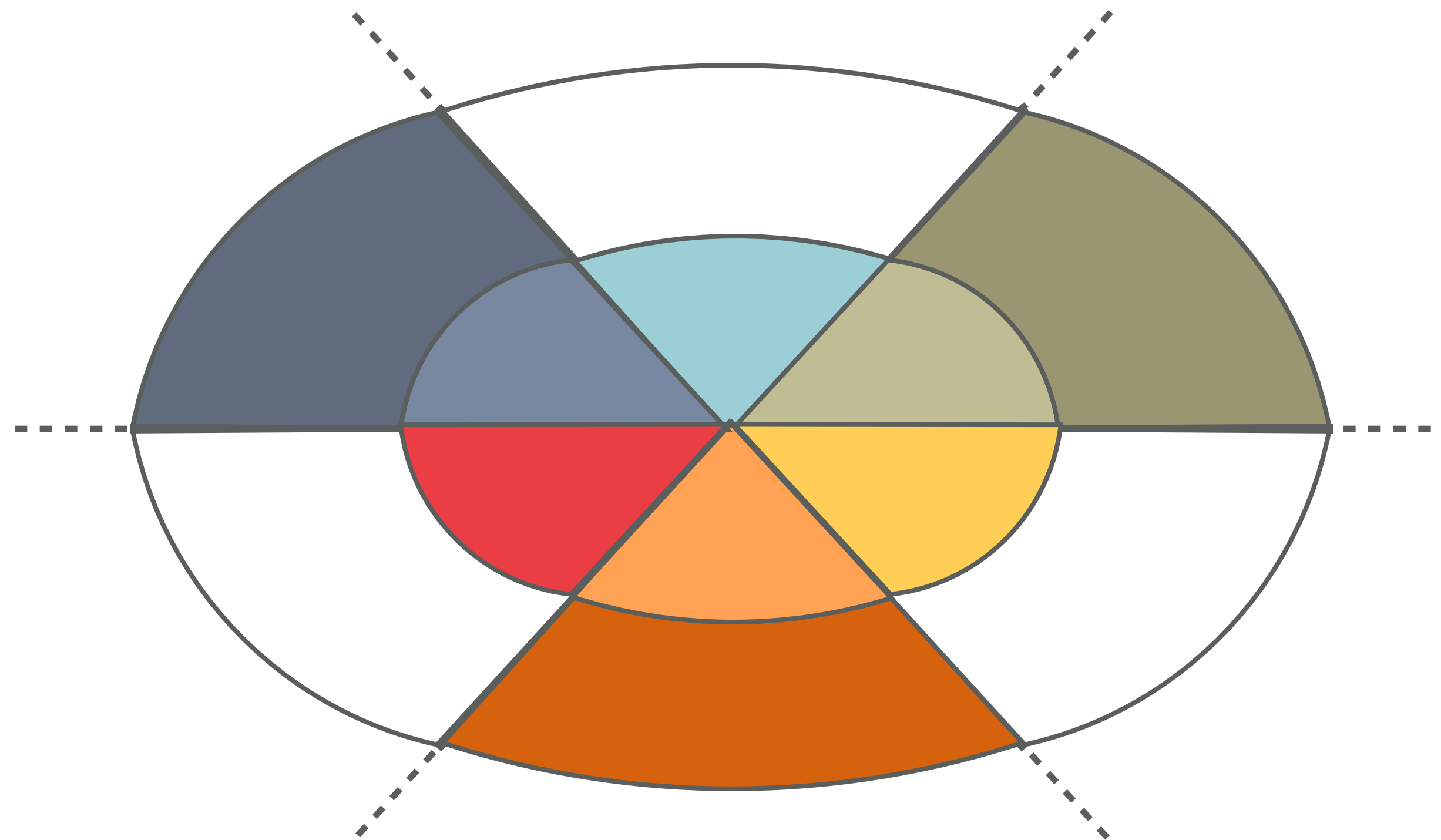


Extrapolation

The out-distribution may not be explicitly manifested in the train data but could be implicitly brought up.

1. Structure the data according to relevant complexity measures
2. Mask domains appropriately and come up with learners that extrapolate well on these.

This gives hope that the extrapolation can “propagate”.

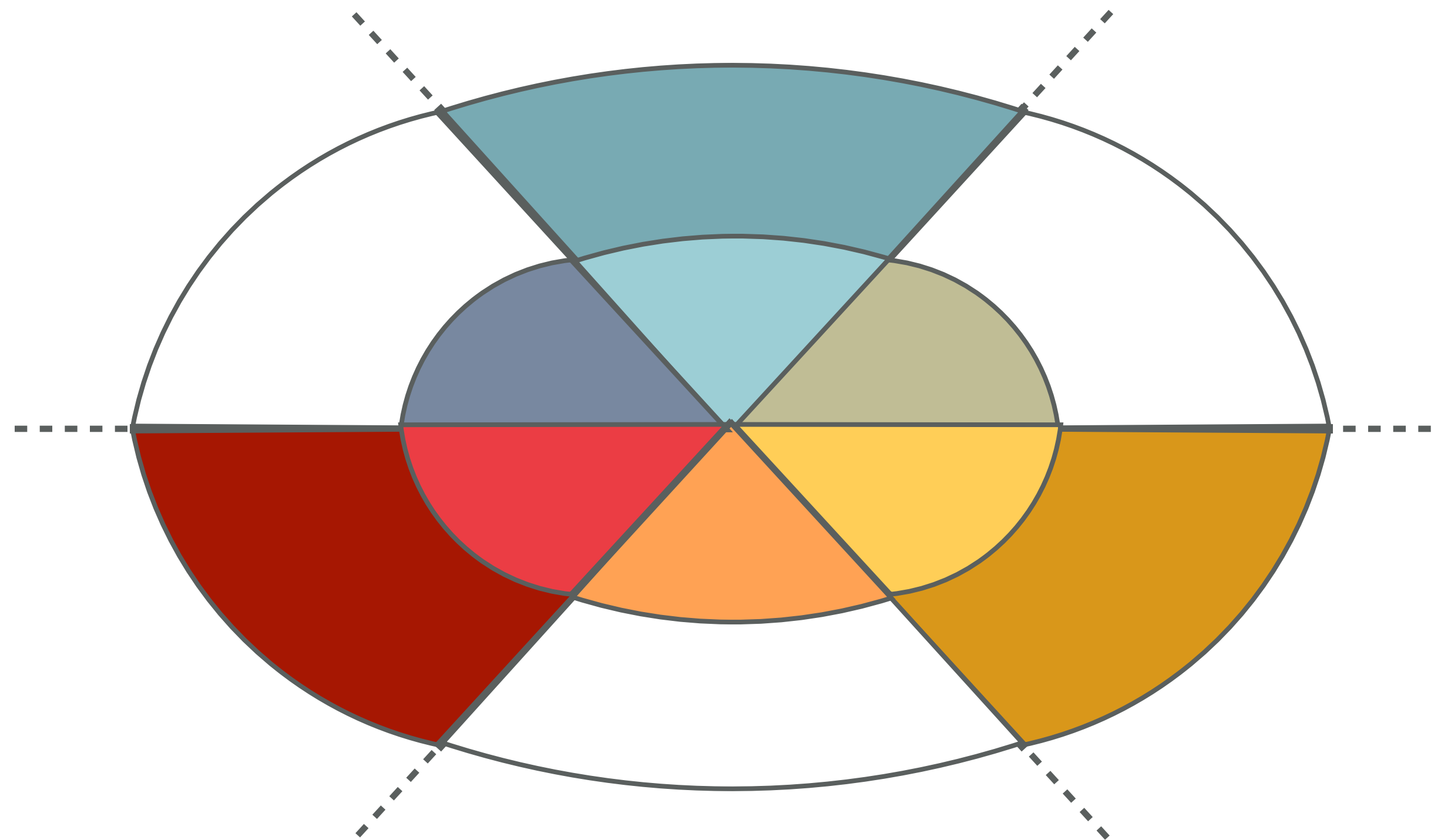


Extrapolation

The out-distribution may not be explicitly manifested in the train data but could be implicitly brought up.

1. Structure the data according to **relevant complexity measures**
2. Mask domains appropriately and **come up with learners that extrapolate** well on these.

This gives hope that the extrapolation can “propagate”.





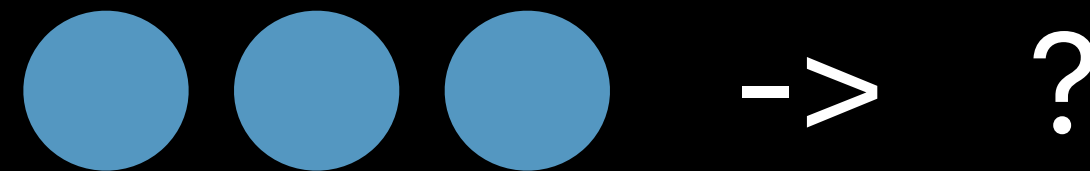
unseen patterns



unseen symbols



unseen lengths



Many of the current models fail on such tasks

-> too much memorization, lack of reasoning

Questions

1. What are relevant “complexity measures” for learning with NN+SGD?

- SQ, CP, information exponent, INAL
- Leap + Distribution locality

2. How to measure “extrapolation” and how NN+SGD extrapolate?

- model for strong OOD -> GOTU
- minimal-degree-profile bias

3. How to come up with “better extrapolators” ?

- the hard way: model design, regularization (in practice: rely also on engines/programs)
- the soft way: multitasking, curriculum, scratchpad and climbing routes

Why reasoning is important

- As tasks become more **complex** (e.g., logic, maths, planning, health)
- As models or datasets become **smaller**
- As AIs get deployed in evolving environments with **distribution shifts**

We need better reasoning to sustain generalization

How to formalize and address these challenges?

Syllabus

1. Class introduction, “What is reasoning?”, successes and failures so far
2. Basics of learning: Generalization notions, deep learning theory, OOD/GOTU
3. Foundation models: LLMs, Transformers, instruction tuning, RLHF, pre- vs. post-training, Scaling laws
4. Scratchpad, CoT, ICL, Curriculum
5. RL for reasoning (as a training method) + RL at test-time (mention Distillation/quantization)?
6. Finish the RL part + Illusion of thinking (prompting etc)
7. Abstractions + tool calling (EvoTune)
8. Reasoning tasks: planning, logic, math, formal maths, coding [if we already use these before, change topic]
9. Other topics: distillation and self-improvement
10. Using the rest for projects.