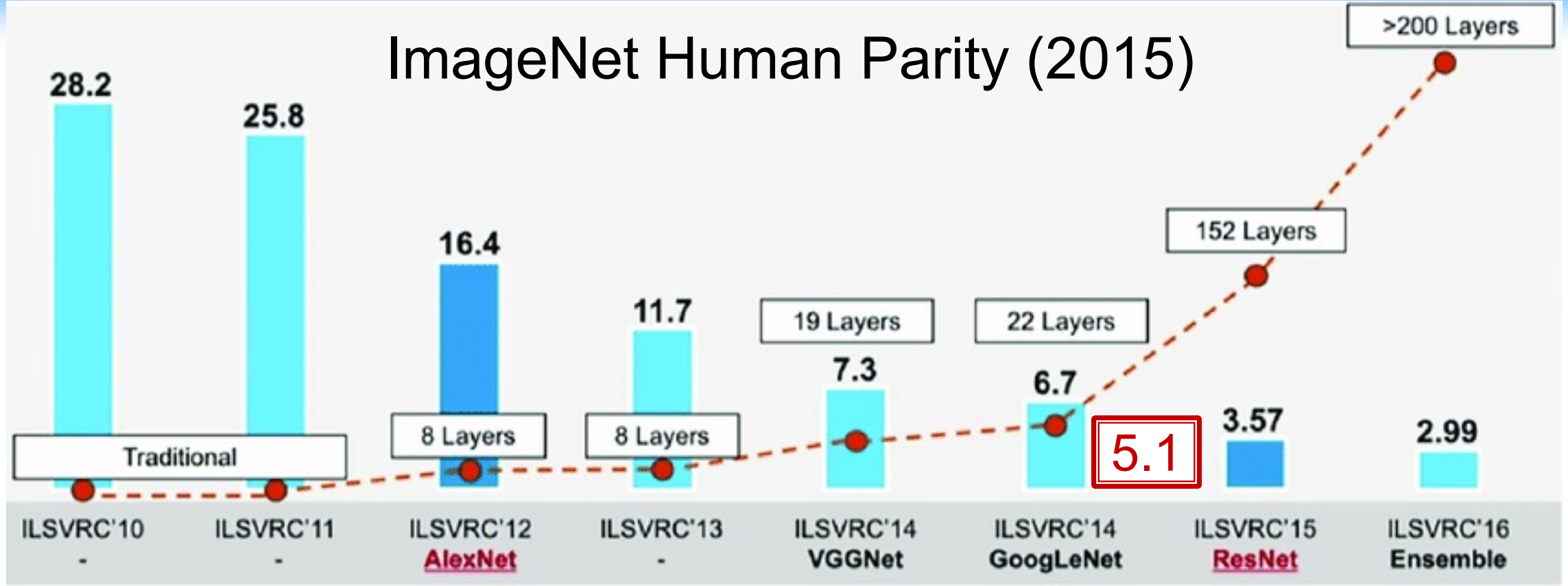
理解与探索大模型能力涌现

东 昱 晓

清华大学计算机系

https://keg.cs.tsinghua.edu.cn/yuxiao/
yuxiaod@tsinghua.edu.cn

ImageNet Human Parity (2015)



PC: Web

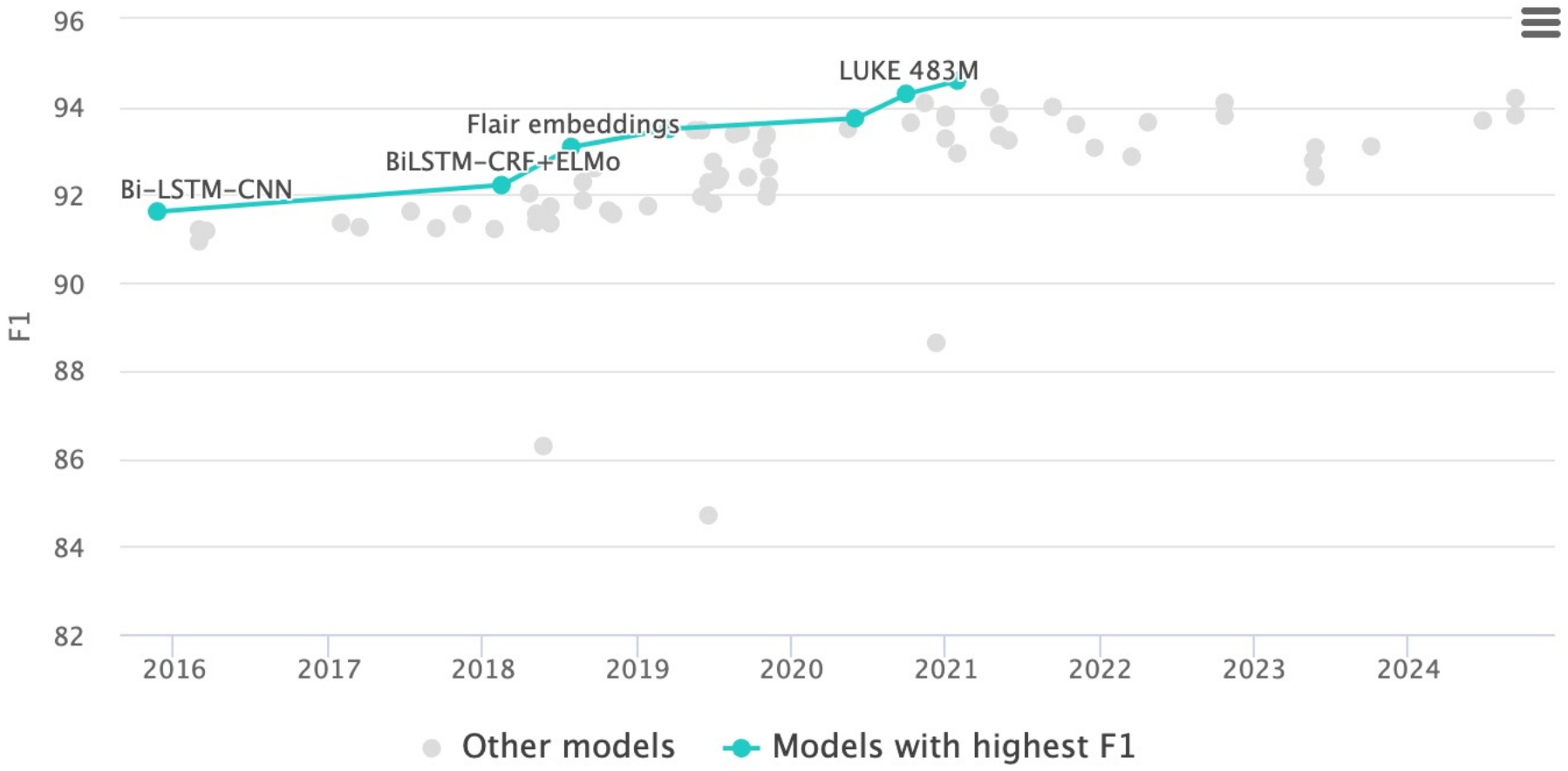
1950-2022

“The Great Wall of China was built from as early as the 7th century BC, with selective stretches later joined by Qin Shi Huang (220–206 BC), the first emperor of China. ”

“It is really gorgeous! ”

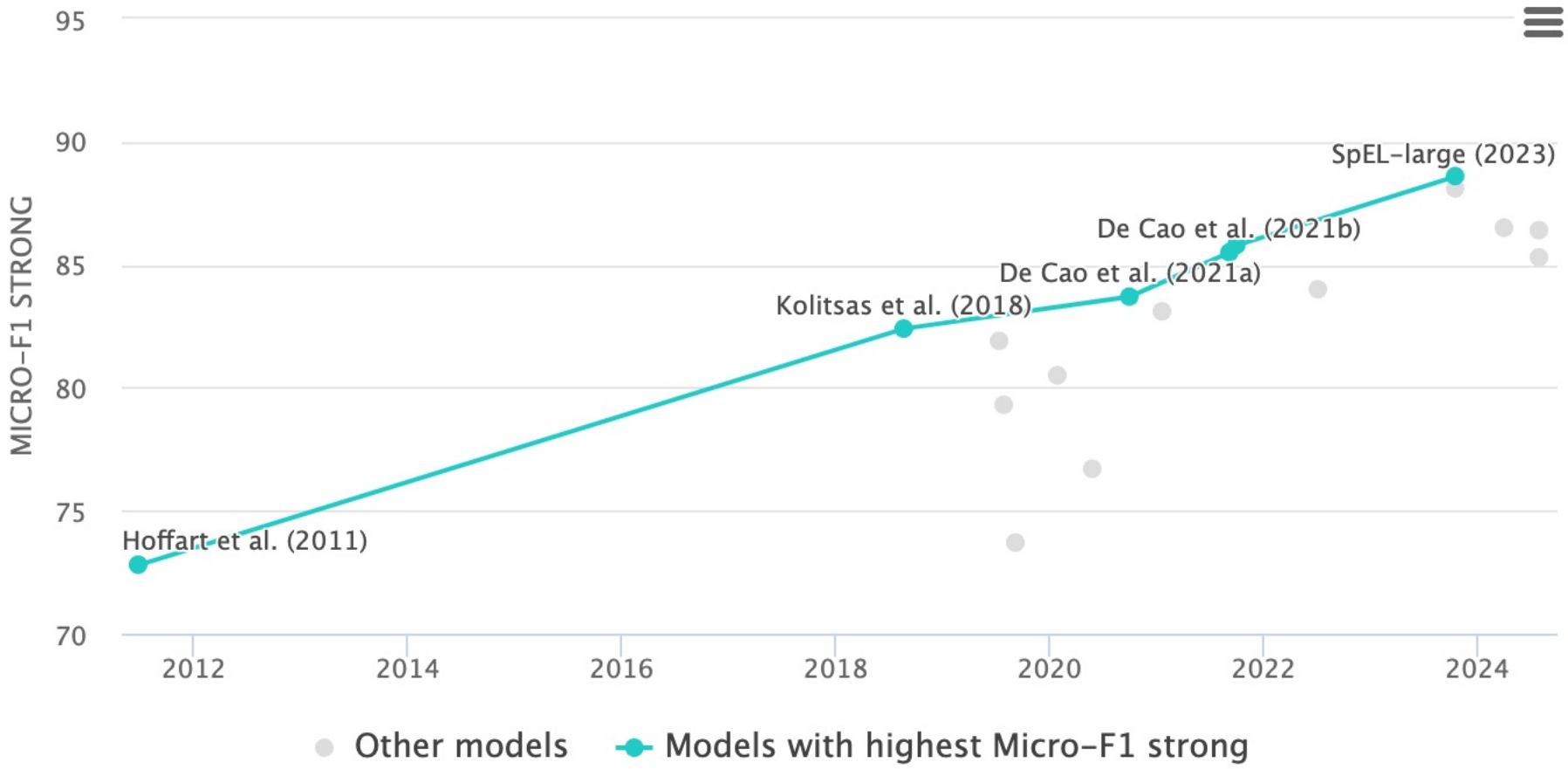
https://en.wikipedia.org/wiki/Great_Wall_of_China

Named Entity Recognition (NER) on CoNLL 2023



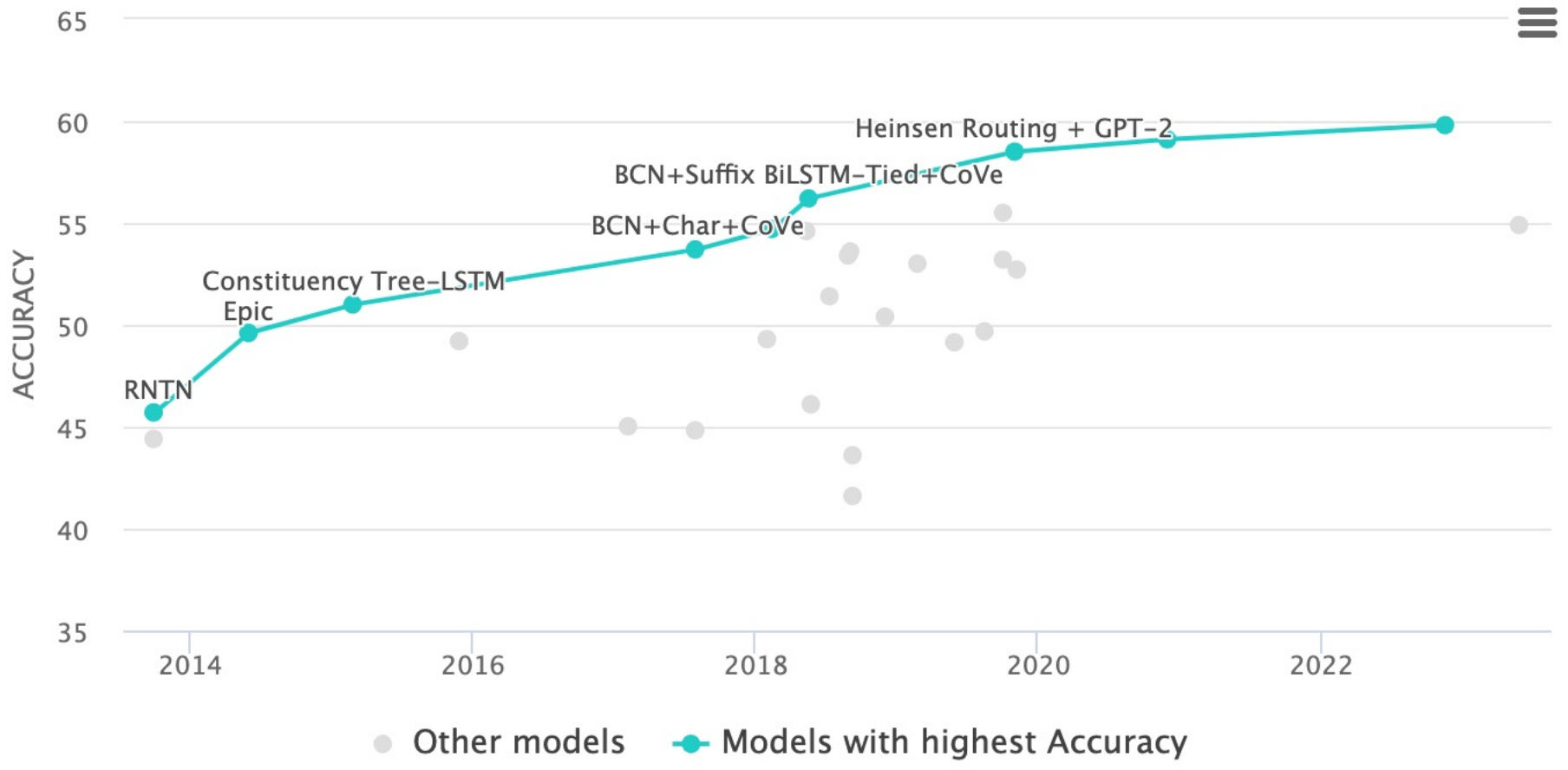
https://paperswithcode.com/sota/named-entity-recognition-ner-on-conll-2003

Entity Linking on AIDA-CoNLL



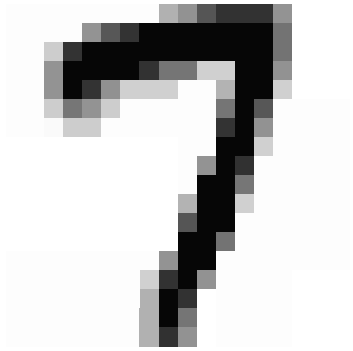
https://paperswithcode.com/sota/entity-linking-on-aida-conll

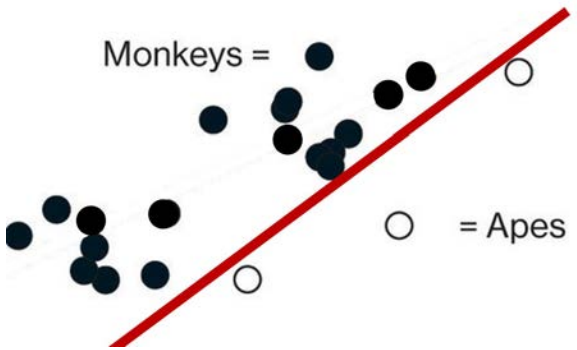
Sentiment Analysis on SST-5



https://paperswithcode.com/sota/sentiment-analysis-on-sst-5-fine-grained

Perception Tasks





Reasoning/Cognitive Tasks

- 一斤西瓜10元钱，5斤西瓜多少钱?
- 一斤苹果10元，一斤西瓜5元，那么2斤苹果和3斤西瓜多少钱?
- 我把可口可乐洒在了桌子上，接下来该如何清楚桌面?

2020~2022

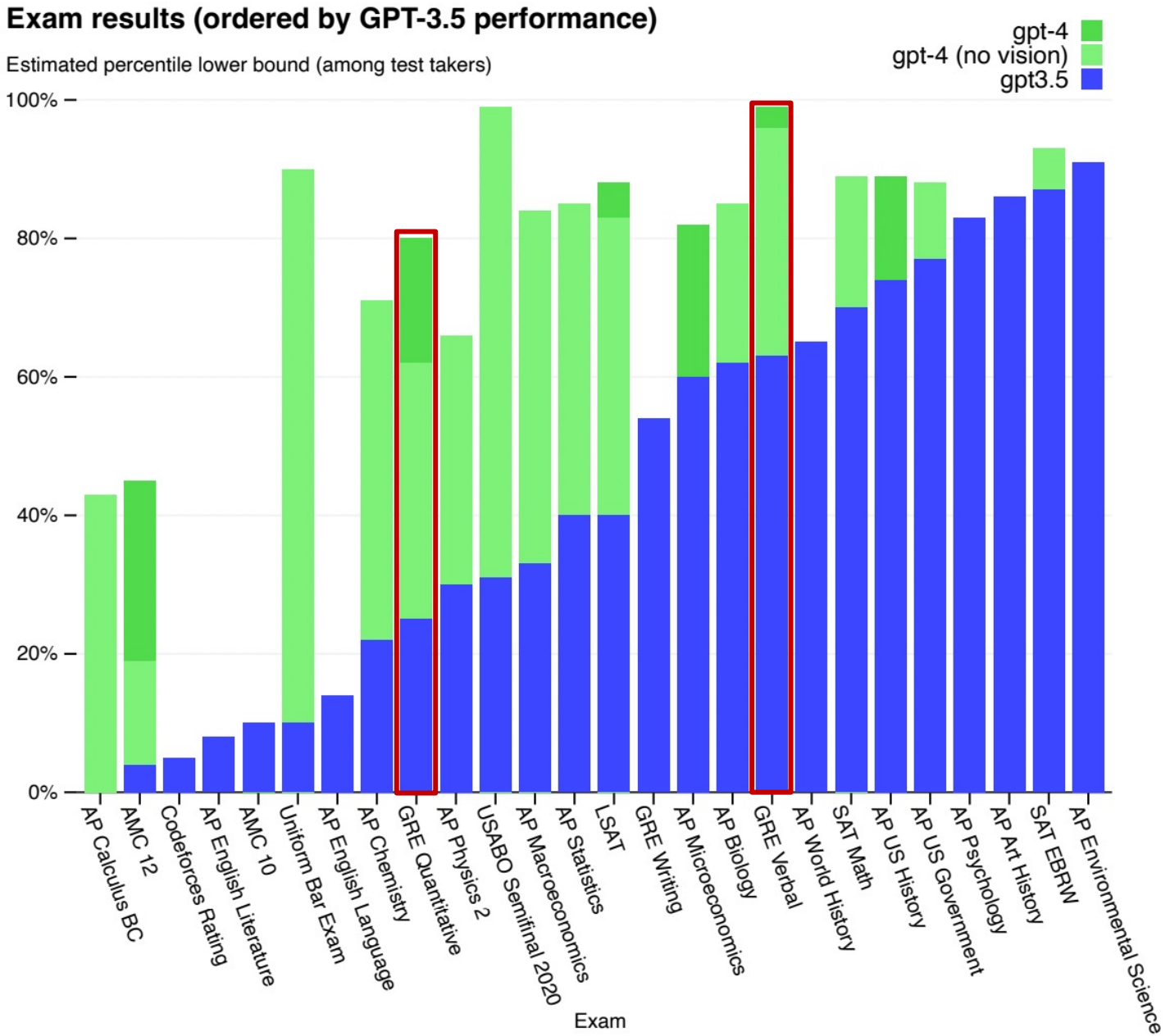
Math (GSM8k):

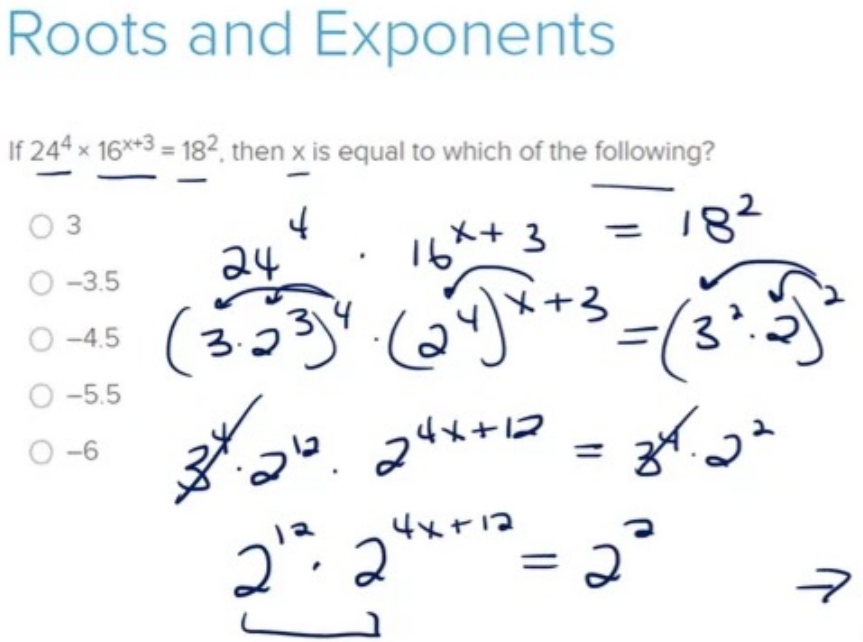
Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

Last Letter Concatenation:

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

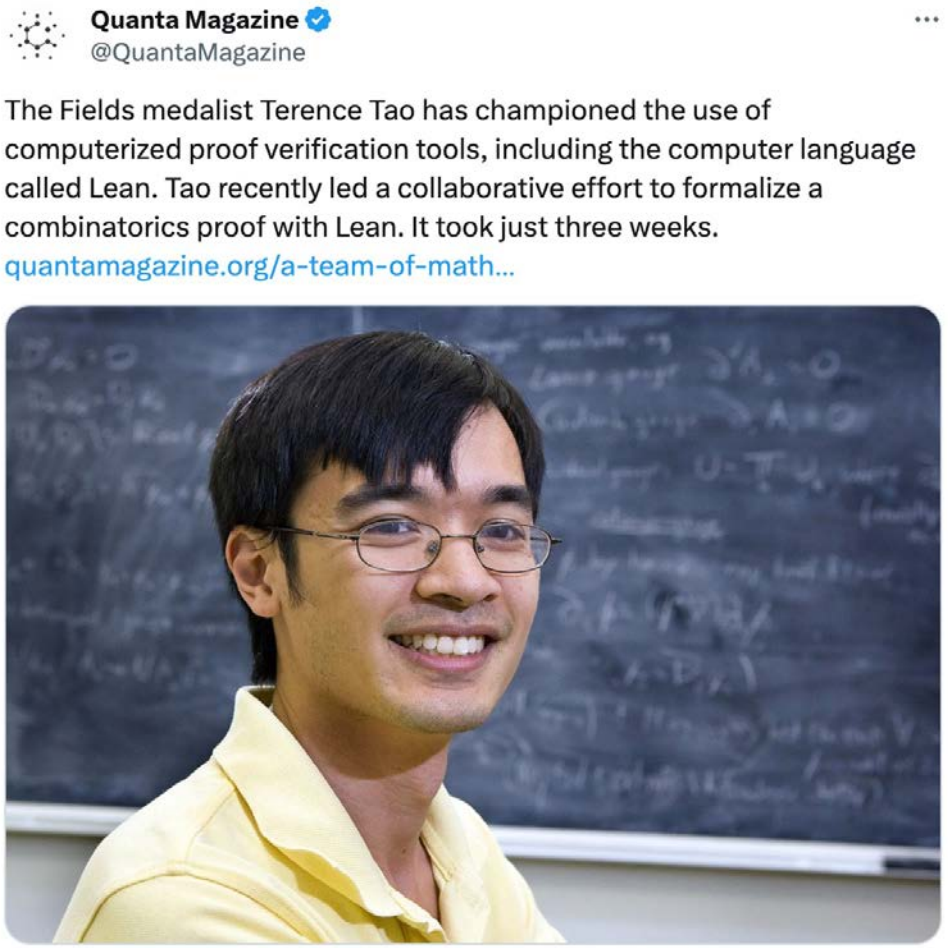
2023.3.14 GPT-4





OpenAI. GPT-4 Technical Report

2023.12.05



菲尔斯奖获得者陶哲轩使用GPT-4编写Lean代码证明Marton猜想 。2023年12月5日

"I expect, say, 2026-level AI, when used properly, will be a trustworthy co-author in mathematical research, and in many other fields as well."

Why Large Models?

语言 ⇨ **推理**

- 对话
  (理解+生成)

- 数学题
- 代码题
- 物化生题

GPT

GPT-1

Improving Language Understanding by Generative Pre-Training

Alec Radford OpenAI alec@openai.com
Karthik Narasimhan OpenAI karthikn@openai.com
Tim Salimans OpenAI tim@openai.com
Ilya Sutskever OpenAI ilyasu@openai.com

GPT-2

Language Models are Unsupervised Multitask Learners

Alec Radford [*1] **Jeffrey Wu** [*1] **Rewon Child** [1] **David Luan** [1] **Dario Amodei** [**1] **Ilya Sutskever** [**1]

GPT-3

Language Models are Few-Shot Learners

Tom B. Brown* **Benjamin Mann*** **Nick Ryder*** **Melanie Subbiah***

Jared Kaplan† **Prafulla Dhariwal** **Arvind Neelakantan** **Pranav Shyam** **Girish Sastry**

Amanda Askell **Sandhini Agarwal** **Ariel Herbert-Voss** **Gretchen Krueger** **Tom Henighan**

Rewon Child **Aditya Ramesh** **Daniel M. Ziegler** **Jeffrey Wu** **Clemens Winter**

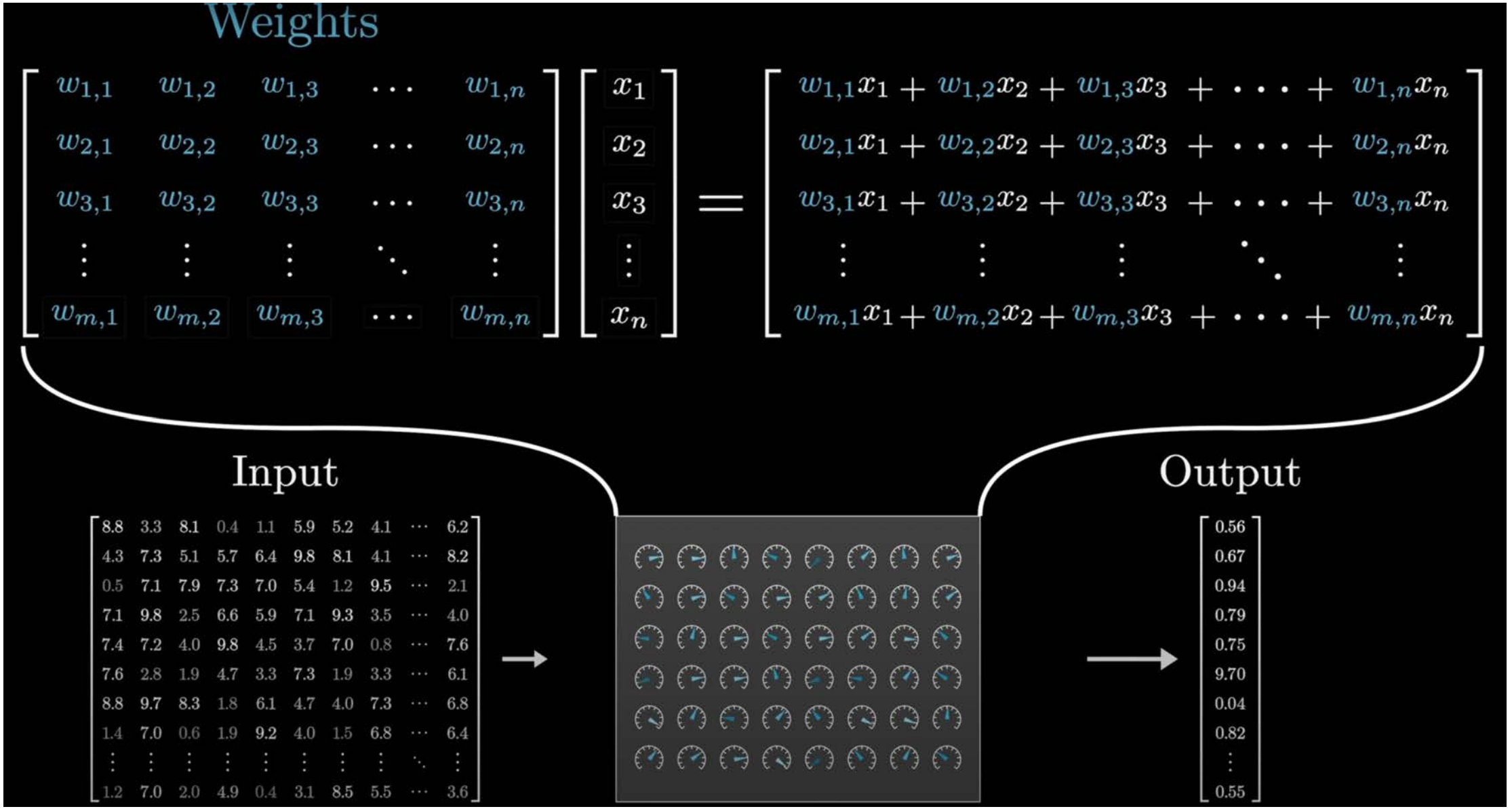
Christopher Hesse **Mark Chen** **Eric Sigler** **Mateusz Litwin** **Scott Gray**

Benjamin Chess **Jack Clark** **Christopher Berner**

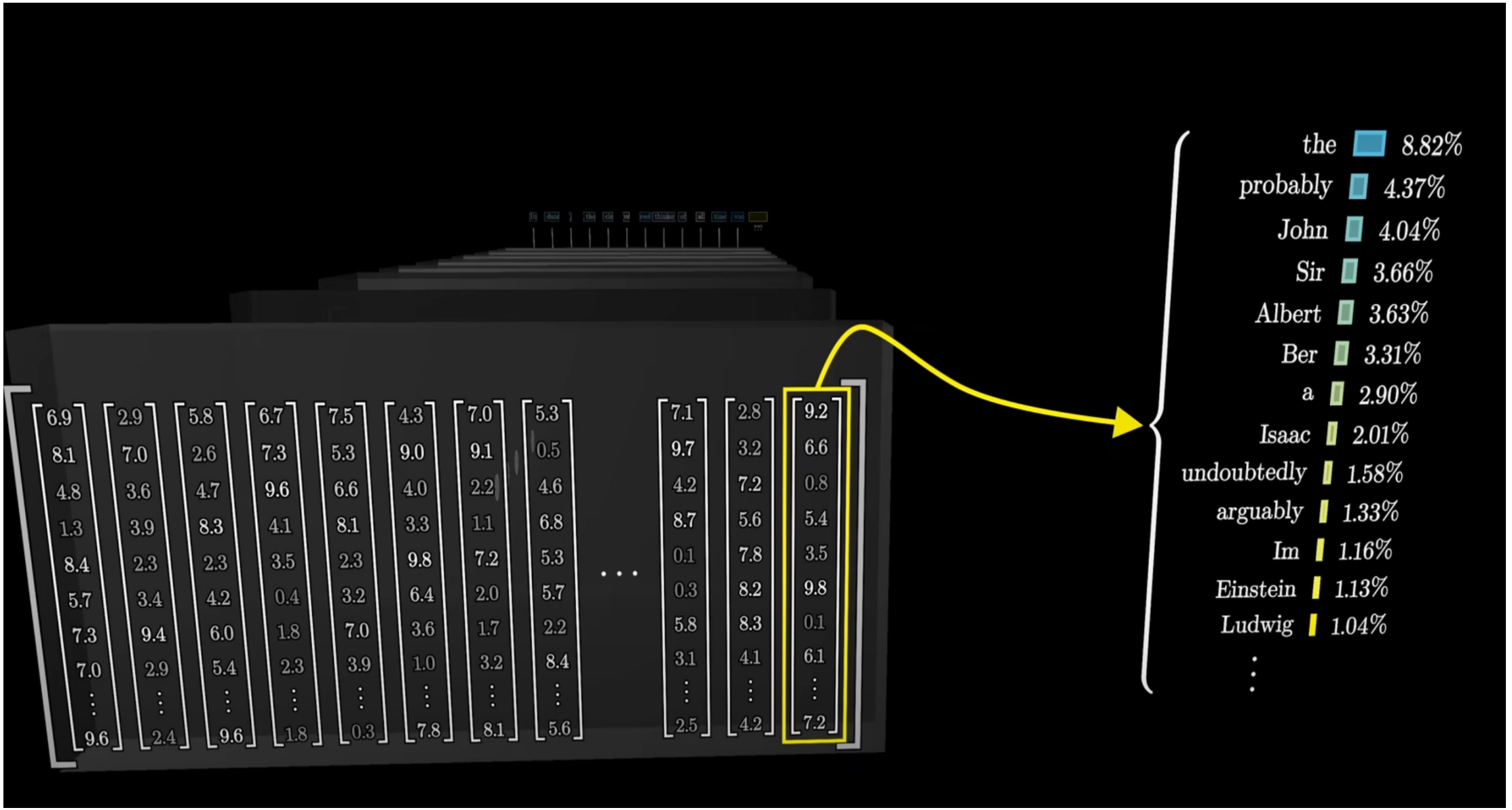
Sam McCandlish **Alec Radford** **Ilya Sutskever** **Dario Amodei**

Next Token Prediction

$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta)$$



3Blue1Brown

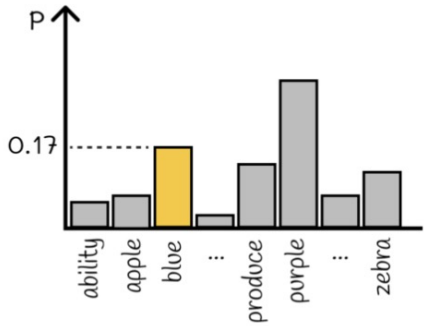
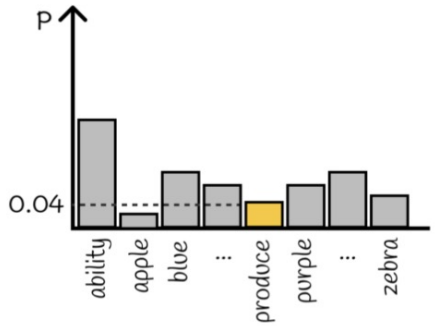
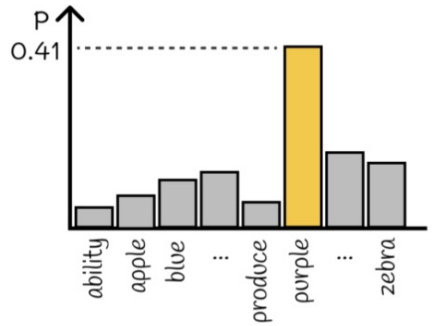

6.9 8.1 4.8 1.3 8.4 5.7 7.3 7.0 9.6
2.9 7.0 3.6 3.9 2.3 3.4 9.4 2.9 2.4
5.8 2.6 4.7 8.3 2.3 4.2 6.0 5.4 9.6
6.7 7.3 9.6 4.1 3.5 0.4 1.8 2.3 1.8
7.5 5.3 6.6 8.1 2.3 3.2 7.0 3.9 0.3
4.3 9.0 4.0 3.3 9.8 6.4 3.6 1.0 7.8
7.0 9.1 2.2 1.1 7.2 2.0 1.7 3.2 8.1
5.3 0.5 4.6 6.8 5.3 5.7 2.2 8.4 5.6
7.1 9.7 4.2 8.7 0.1 0.3 5.8 3.1 2.5
2.8 3.2 7.2 5.6 7.8 8.2 8.3 4.1 4.2
9.2 6.6 0.8 5.4 3.5 9.8 0.1 6.1 7.2
the 8.82%
probably 4.37%
John 4.04%
Sir 3.66%
Albert 3.63%
Ber 3.31%
a 2.90%
Isaac 2.01%
undoubtedly 1.58%
arguably 1.33%
Im 1.16%
Einstein 1.13%
Ludwig 1.04%

Next Token Prediction

$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta)$$

Mixing red and blue produces purple.

1 2 3 4 5 6

subsequence	Mixing red and ? (1 2 3 4)	red and blue ? (2 3 4 5)	and blue produces ? (3 4 5 6)
model's probability distribution			
P	$P(u_4 \mid u_1, u_2, u_3) = 0.17$	$P(u_5 \mid u_2, u_3, u_4) = 0.04$	$P(u_6 \mid u_3, u_4, u_5) = 0.41$
log P	$\log 0.17 = -1.77$	$\log 0.04 = -3.22$	$\log 0.41 = -0.89$

Language modeling loss: $L_1 = -1.77 - 3.22 - 0.89 = -5.88$

Image: https://towardsdatascience.com/large-language-models-gpt-1-generative-pre-trained-transformer-7b895f296d3b/

Next Token Prediction

Prefix	Next word [task]
A transformer is a deep learning architecture, initially proposed in	2017 [factual recall]
A transformer is a deep learning architecture, initially proposed in 2017	, [comma prediction]
A transformer is a deep learning architecture, initially proposed in 2017,	that [grammar]
A transformer is a deep learning architecture, initially proposed in 2017, that	relies [impossible task?]

$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta)$$

Jason Wei, https://www.jasonwei.net/blog/some-intuitions-about-large-language-models

Next Token Prediction

Prefix {choice_1, choice_2}	Task
In my free time, I like to {run, banana}	Grammar
I went to the zoo to see giraffes, lions, and {zebras, spoon}	Lexical semantics
The capital of Denmark is {Copenhagen, London}	World knowledge
I was laughing the entire time, the movie was {good, bad}	Sentiment analysis
The word for "pretty" in Spanish is {bonita, hola}	Translation
First grade arithmetic exam: 3 + 8 + 4 = {15, 11}	Math question

Jason Wei, https://www.jasonwei.net/blog/some-intuitions-about-large-language-models

千亿基座模型预训练

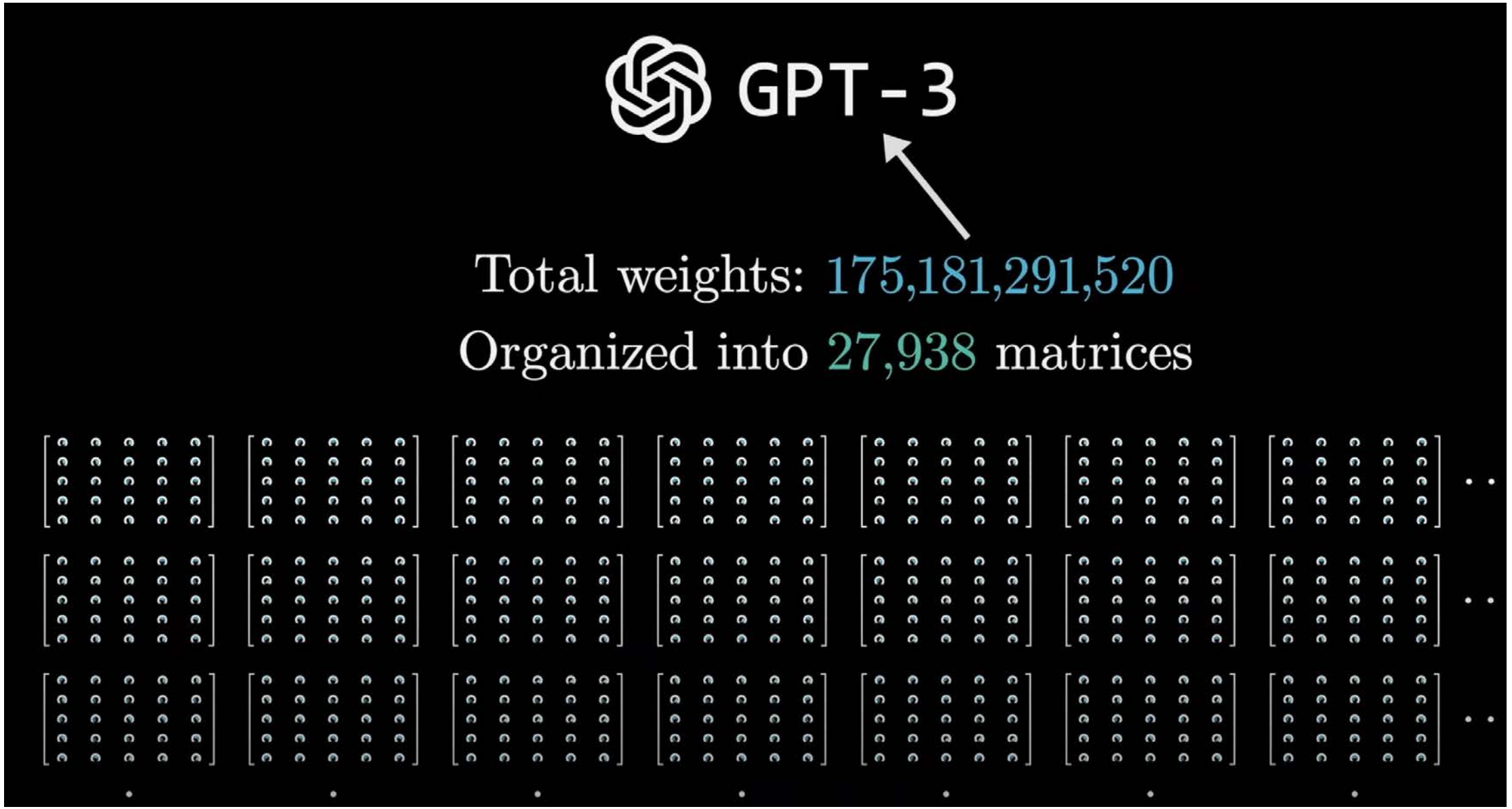
1. 175B pre-trained model

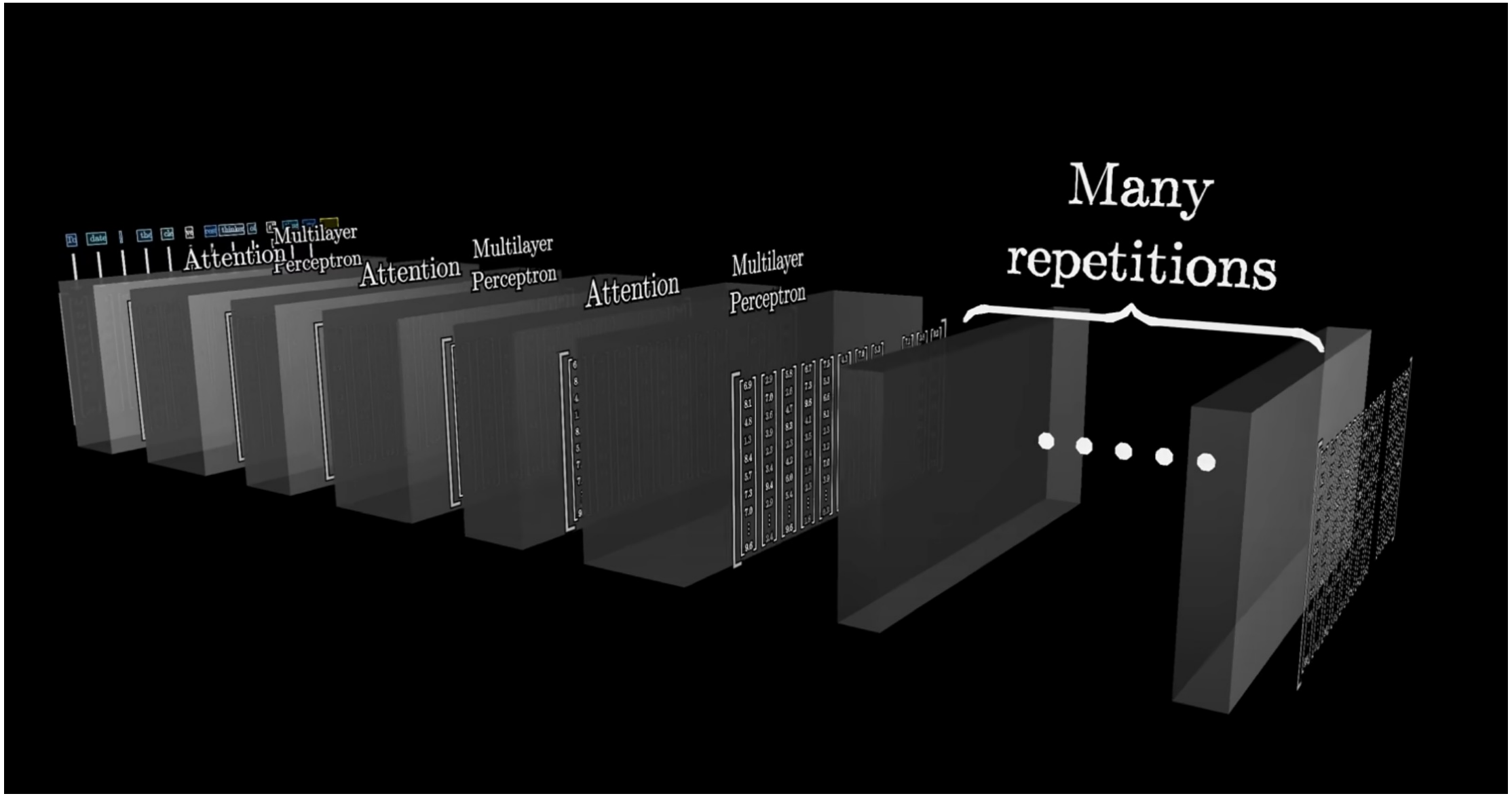
May 2020

GPT-3

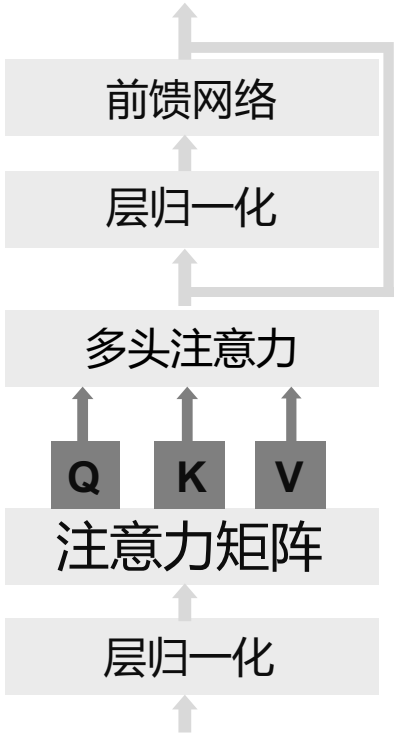
GPT-3
+ RLHF

GPT-3: 175 Billion


GPT-3
Total weights: 175,181,291,520
Organized into 27,938 matrices

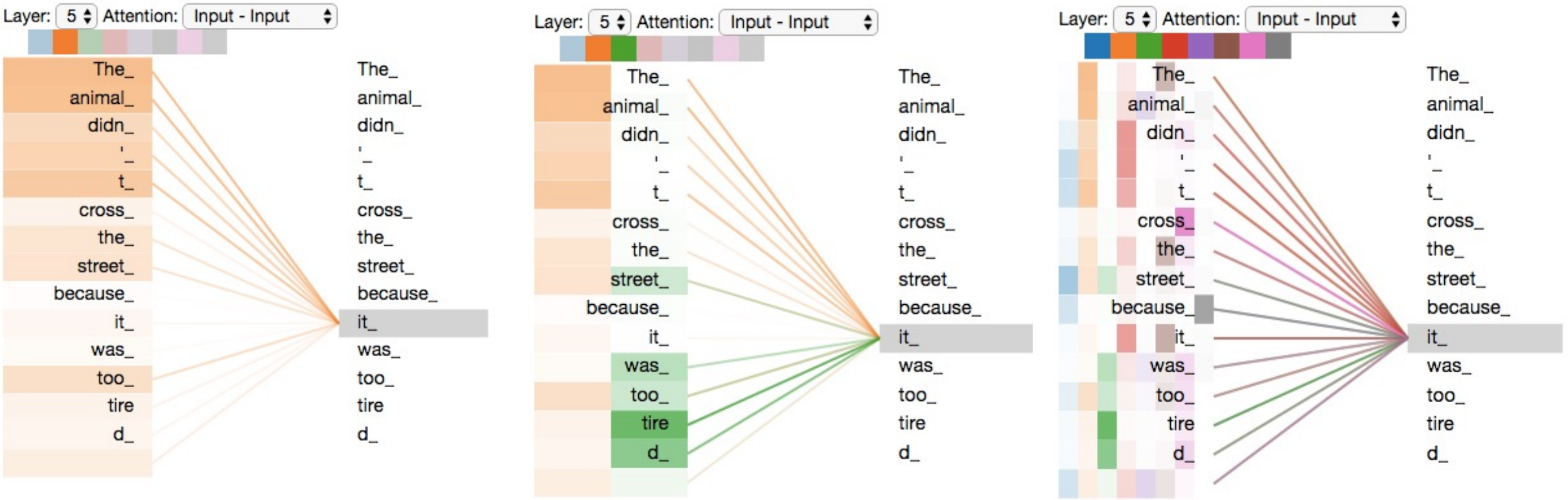

Attention
Multilayer Perceptron
Attention
Multilayer Perceptron
Attention
Multilayer Perceptron
Many repetitions

Attention



$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

Attention



”The animal didn't cross the street because it was too tired”

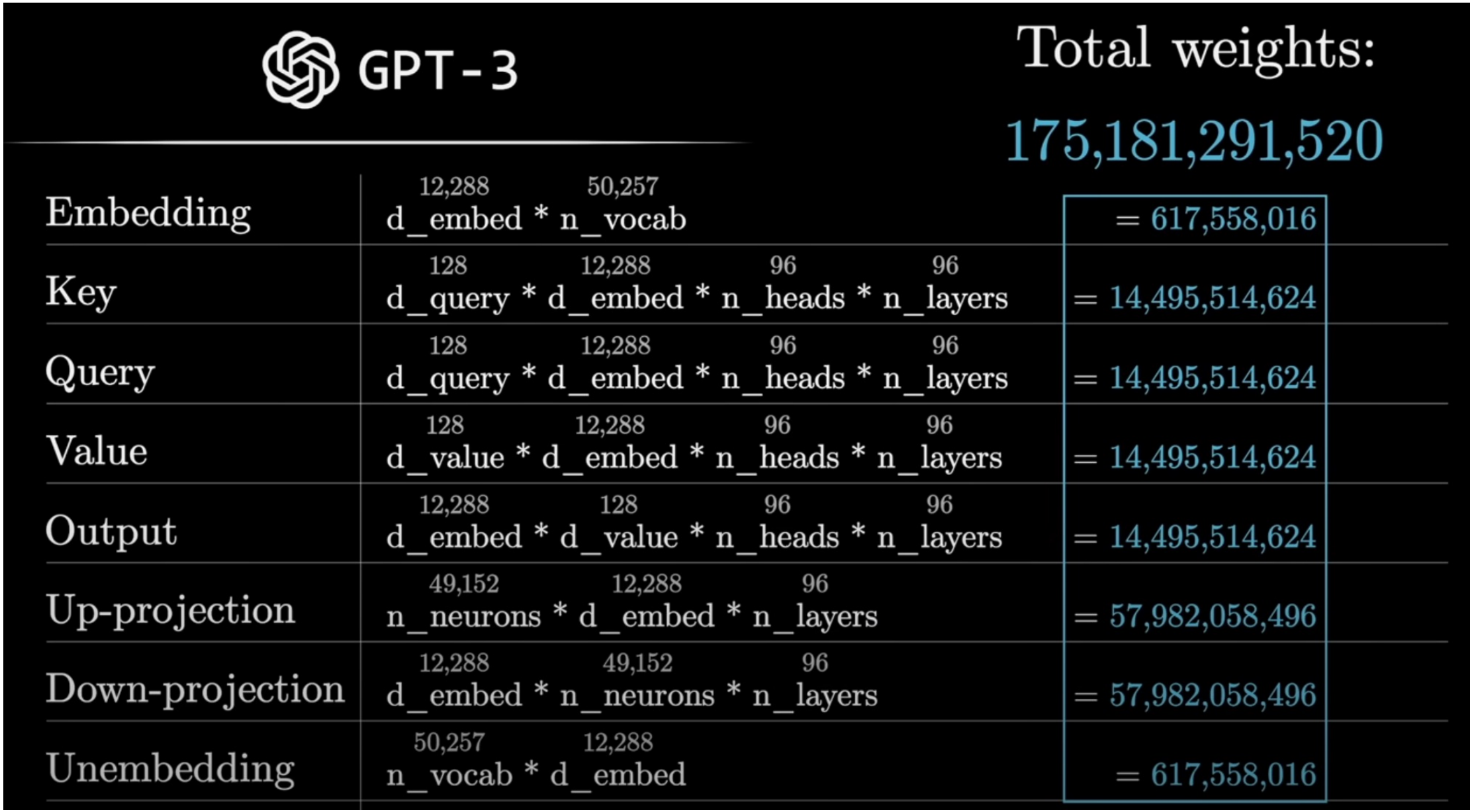
Image Credit: Jay Alammar. https://jalammar.github.io/illustrated-transformer/

Total weights: 175,181,291,520

Organized into 27,938 matrices

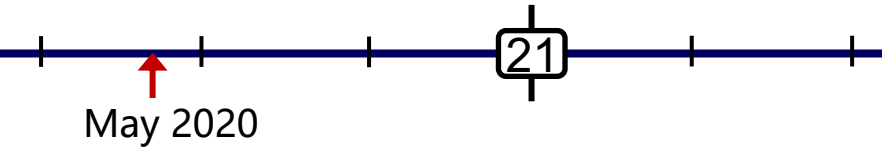
GPT-3

Embedding	
Key	
Query	
Value	
Output	
Up-projection	
Down-projection	
Unembedding	


GPT-3
Total weights:
175,181,291,520
Embedding
d_embed * n_vocab
12,288 50,257
= 617,558,016
Key
d_query * d_embed * n_heads * n_layers
128 12,288 96 96
= 14,495,514,624
Query
d_query * d_embed * n_heads * n_layers
128 12,288 96 96
= 14,495,514,624
Value
d_value * d_embed * n_heads * n_layers
128 12,288 96 96
= 14,495,514,624
Output
d_embed * d_value * n_heads * n_layers
12,288 128 96 96
= 14,495,514,624
Up-projection
n_neurons * d_embed * n_layers
49,152 12,288 96
= 57,982,058,496
Down-projection
d_embed * n_neurons * n_layers
12,288 49,152 96
= 57,982,058,496
Unembedding
n_vocab * d_embed
50,257 12,288
= 617,558,016

GPT

1. 175B pre-trained model



GPT-3

GPT-3
+ RLHF

GPT-3: 175 Billion

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1  Translate English to French:          ← task
2  sea otter => loutre de mer            ← exar
3  peppermint => menthe poivrée          ←
4  plush girafe => girafe peluche        ←
5  cheese =>                             ← prompt
```

OpenAI.

Pre-Training + Fine-Tuning ➔ Few/Zero-Shot Learning

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  cheese =>                           ← prompt
```

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  sea otter => loutre de mer          ← example
3  cheese =>                           ← prompt
```

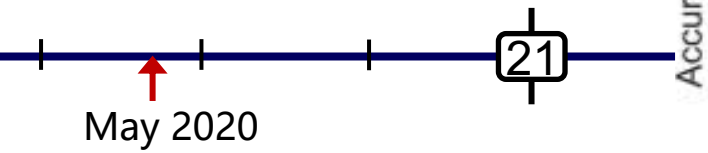
Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  sea otter => loutre de mer          ← examples
3  peppermint => menthe poivrée        ←
4  plush girafe => girafe peluche      ←
5  cheese =>                           ← prompt
```

GPT

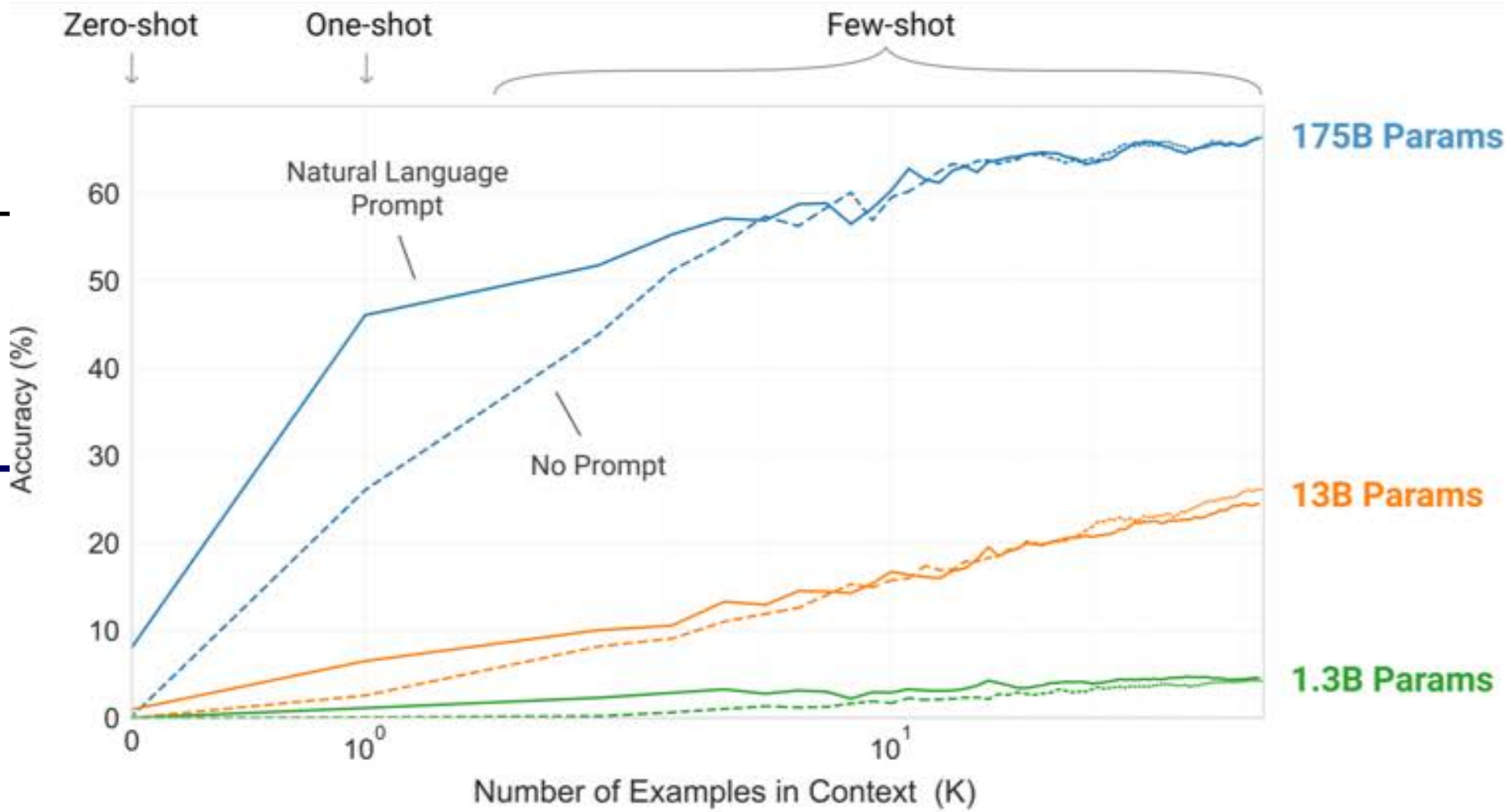
1. 175B pre-trained



GPT-3

GPT-3
+ RLHF

GPT-3: 175 Billion

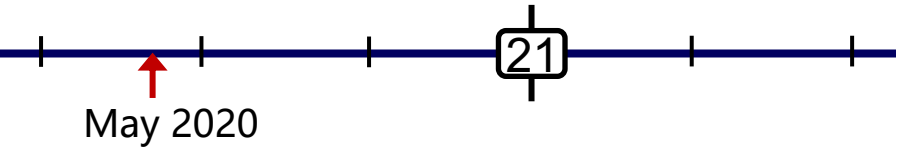


OpenAI.

GPT

- In-context Learning： A is B; C is D; Please answer: E is __ ?

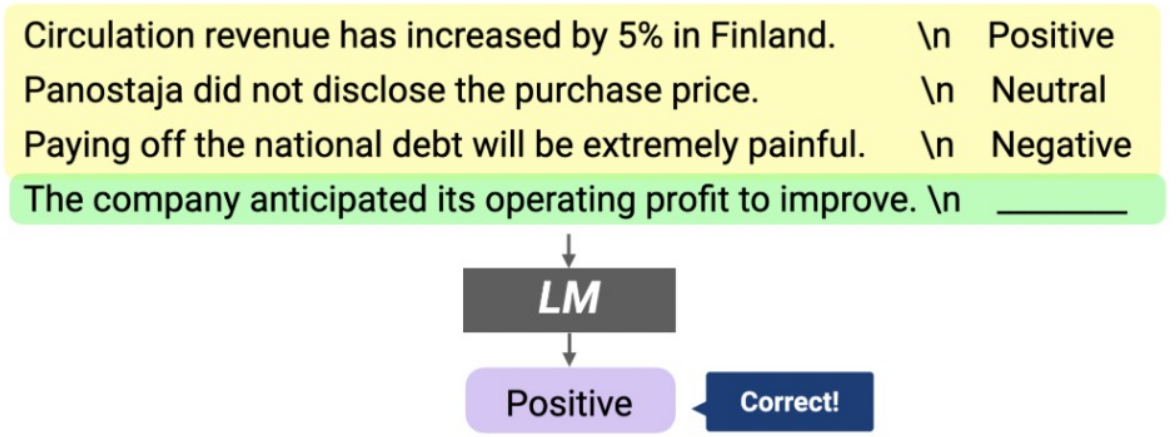
1. 175B pre-trained model

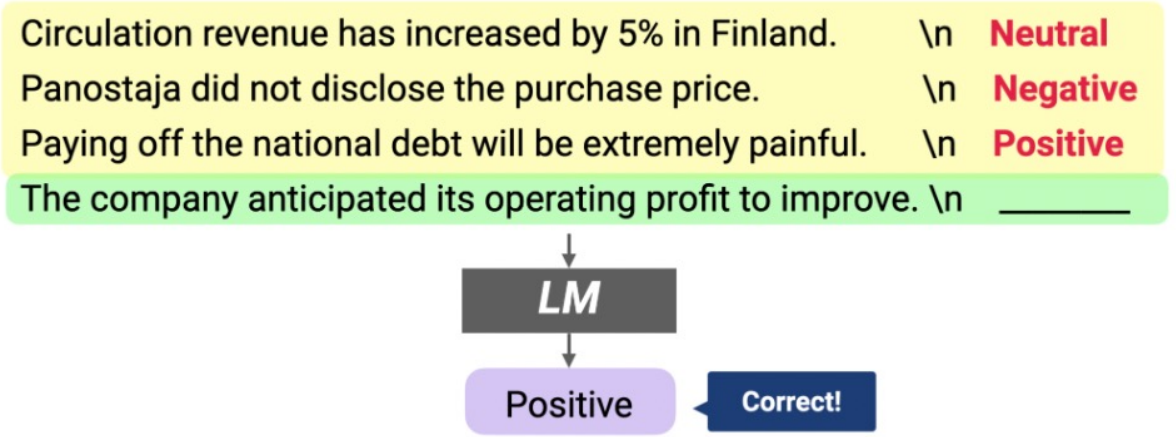


GPT-3

GPT-3
+ RLHF

GPT-3: 175 Billion





Facebook.

Chain of Thought (CoT)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: ***Let's think step by step.***

(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✓

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 tennis balls.
`tennis_balls = 5`
2 cans of 3 tennis balls each is
`bought_balls = 2 * 3`
tennis balls. The answer is
`answer = tennis_balls + bought_balls`

Q: The bakers at the Beverly Hills Bakery baked 200 loaves of bread on Monday morning. They sold 93 loaves in the morning and 39 loaves in the afternoon. A grocery store returned 6 unsold loaves. How many loaves of bread did they have left?

Model Output

A: The bakers started with 200 loaves
`loaves_baked = 200`
They sold 93 in the morning and 39 in the afternoon
`loaves_sold_morning = 93`
`loaves_sold_afternoon = 39`
The grocery store returned 6 loaves.
`loaves_returned = 6`
The answer is

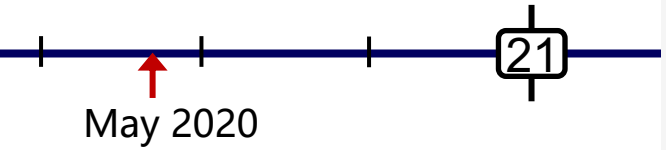
```
answer = loaves_baked - loaves_sold_morning
  - loaves_sold_afternoon + loaves_returned
```

```
>>> print(answer)
74
```

Jason Wei, et al., Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. arXiv: 2201.11903

GPT

1. 175B pre-trained



GPT-3

GPT-3
+ RLHF

GPT-3: 175 Billion

Prompt

Explain the moon landing to a 6 year old in a few sentences.

Completion

GPT-3

```
Explain the theory of gravity to a 6 year old.

Explain the theory of relativity to a 6 year old in a few
sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.
```

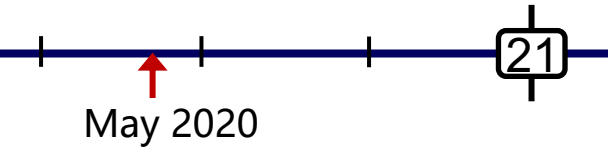
OpenAI.

GPT

2. Supervised Fine-Turning

1. 175B pre-trained model

3. RL from Human Feedback



GPT-3

GPT-3
+ RLHF

GPT-3: 175 Billion

Finetune on many tasks ("instruction-tuning")

Input (Commonsense Reasoning)

Here is a goal: Get a cool sleep on summer days.

How would you accomplish this goal?

OPTIONS:

-Keep stack of pillow cases in fridge.

-Keep stack of pillow cases in oven.

Target

keep stack of pillow cases in fridge

Input (Translation)

Translate this sentence to Spanish:

The new office building was built in less than three months.

Target

El nuevo edificio de oficinas se construyó en tres meses.

Sentiment analysis tasks

Coreference resolution tasks

...

Inference on unseen task type

Input (Natural Language Inference)

Premise: At my age you will probably have learnt one lesson.

Hypothesis: It's not certain how many lessons you'll learn by your thirties.

Does the premise entail the hypothesis?

OPTIONS:

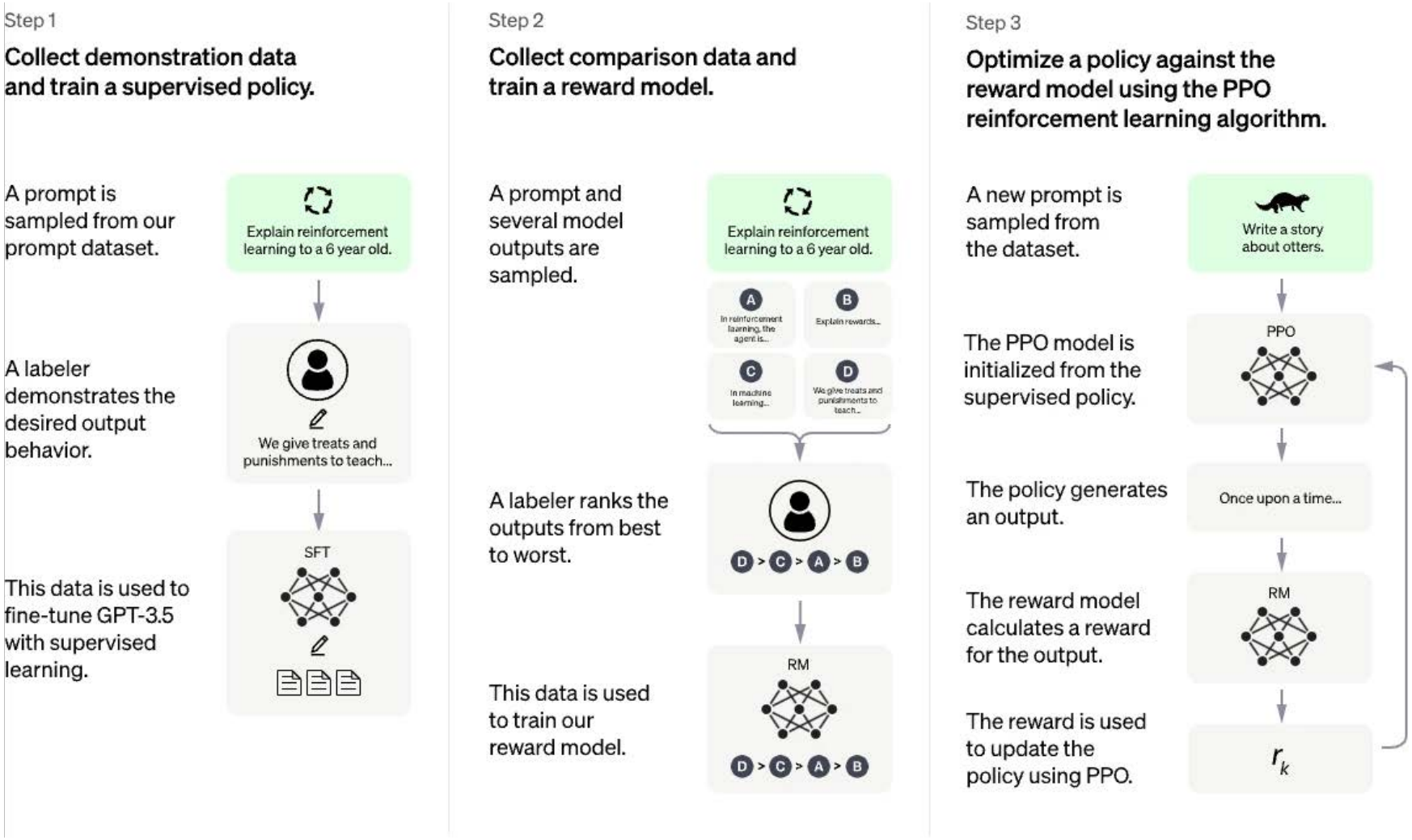
-yes -it is not possible to tell -no

FLAN Response

It is not possible to tell

Google.

ChatGPT



"ChatGPT is fine-tuned from a model in the GPT-3.5 series, which finished training in early 2022."
https://openai.com/index/chatgpt/ Nov. 2022

OpenAI GPT

Prompt

Explain the moon landing to a 6 year old in a few sentences.

Completion

GPT-3

```
Explain the theory of gravity to a 6 year old.

Explain the theory of relativity to a 6 year old in a few
sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.
```

InstructGPT

```
People went to the moon, and they took pictures of what they
saw, and sent them back to the earth so we could all see them.
```

https://openai.com/research/instruction-following

Alignment

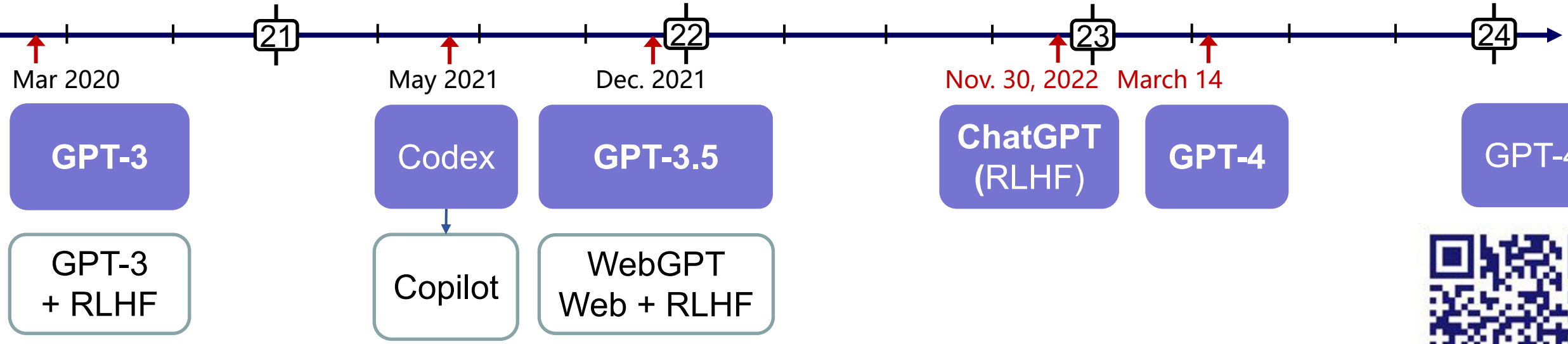
2. Supervised Fine-Turning

1. 175B pre-trained model

3. RL from Human Feedback

预训练

对齐训练



GLM-130B千亿基座模型预训练

OpenAI

- GPT-3 175B

[2020]

Microsoft NVIDIA

- Megatron-Turing-530B

[2021]

Google

- LaMDA 137B
- PaLM 540B

[2021~2022]

DeepMind

- Chinchilla 70B
- Gopher 260B

[2021~2022]

清華大學 Tsinghua University

智谱·AI

- GLM-130B

[2022]

Meta

- OPT 175B

[2022]

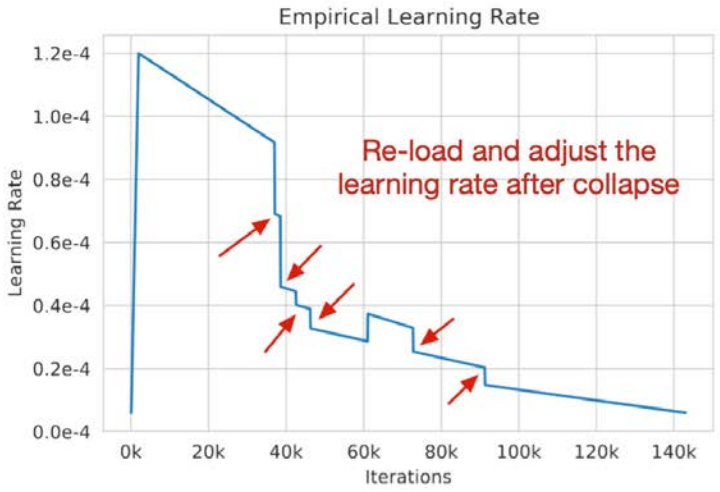
BigScience

- BLOOM 176B

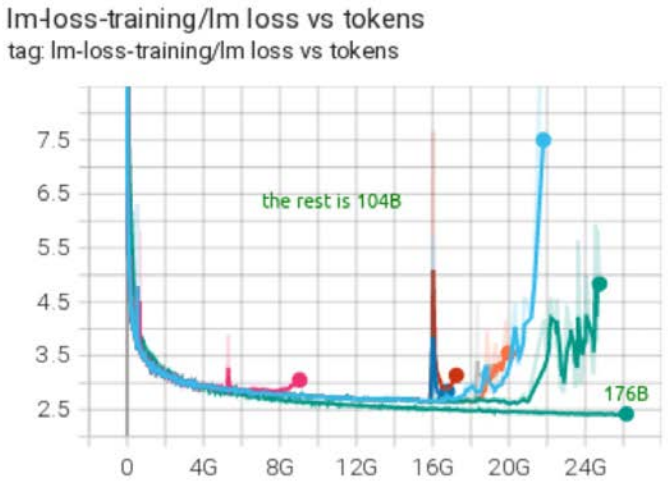
[2022]

GLM-130B千亿模型训练最大挑战：训练稳定性

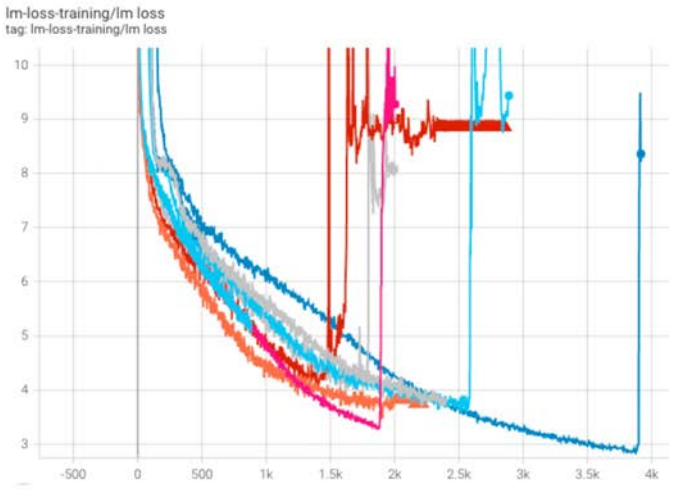
- 权衡利弊：训练稳定性（高精度低效）还是训练效率（低精度高效）
- 目前已开源训练过程大模型的解决方案
  - **FB OPT-175B**：训练崩溃时反复调整学习率/跳过数据（权宜之计，损失性能）
  - **HF BLOOM 176B**：embedding norm和BF16（损失性能，有限适配平台）



(a) OPT 175B's experiments



(b) BLOOM 176B's experiments



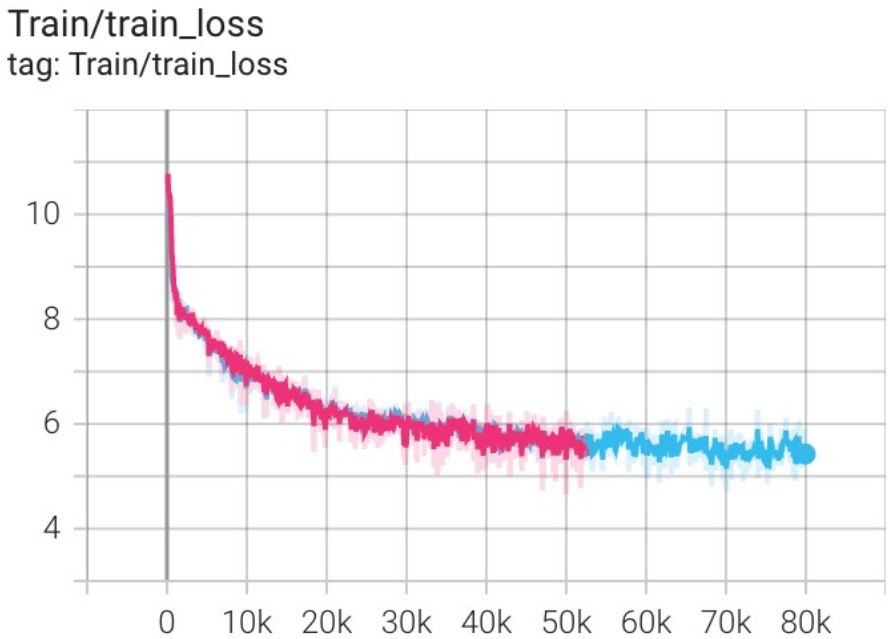
(c) GLM 130B's experiments

Sources: OPT / BLOOM / GLM-130B

跨平台高效训练千亿模型

- 跨平台兼容：swDeepSpeed 训练库 ⇔ 与 DeepSpeed API 兼容
  - 支持神威超算**申威**架构，一行代码无缝替换兼容
  - 实现并行通信策略，混合精度策略，ZeRO 优化器
  - 同一套训练框架可在**不同架构（英伟达、申威、海光DCU、昇腾）集群**上对齐训练曲线

```
import swDeepSpeed as deepspeed
model, optimizer, _, _ = deepspeed.initialize(
    model=model,
    model_parameters=param_groups,
    args=args,
    mpu=mpu,
    dist_init_required=False,
    config_params=config_params
)
```



计算机系KEG实验室、PACMAN实验室、NLP实验室等

Zeng, Liu, et al. GLM-130B: An Open Bilingual Pre-trained Model. ICLR'23

跨平台高效训练千亿模型

斯坦福大学的世界主流大模型评测，GLM-130B 亚洲唯一入选模型：

- 准确性、恶意性与GPT-3持平；
- 鲁棒性和校准误差在所有模型中表现最佳；

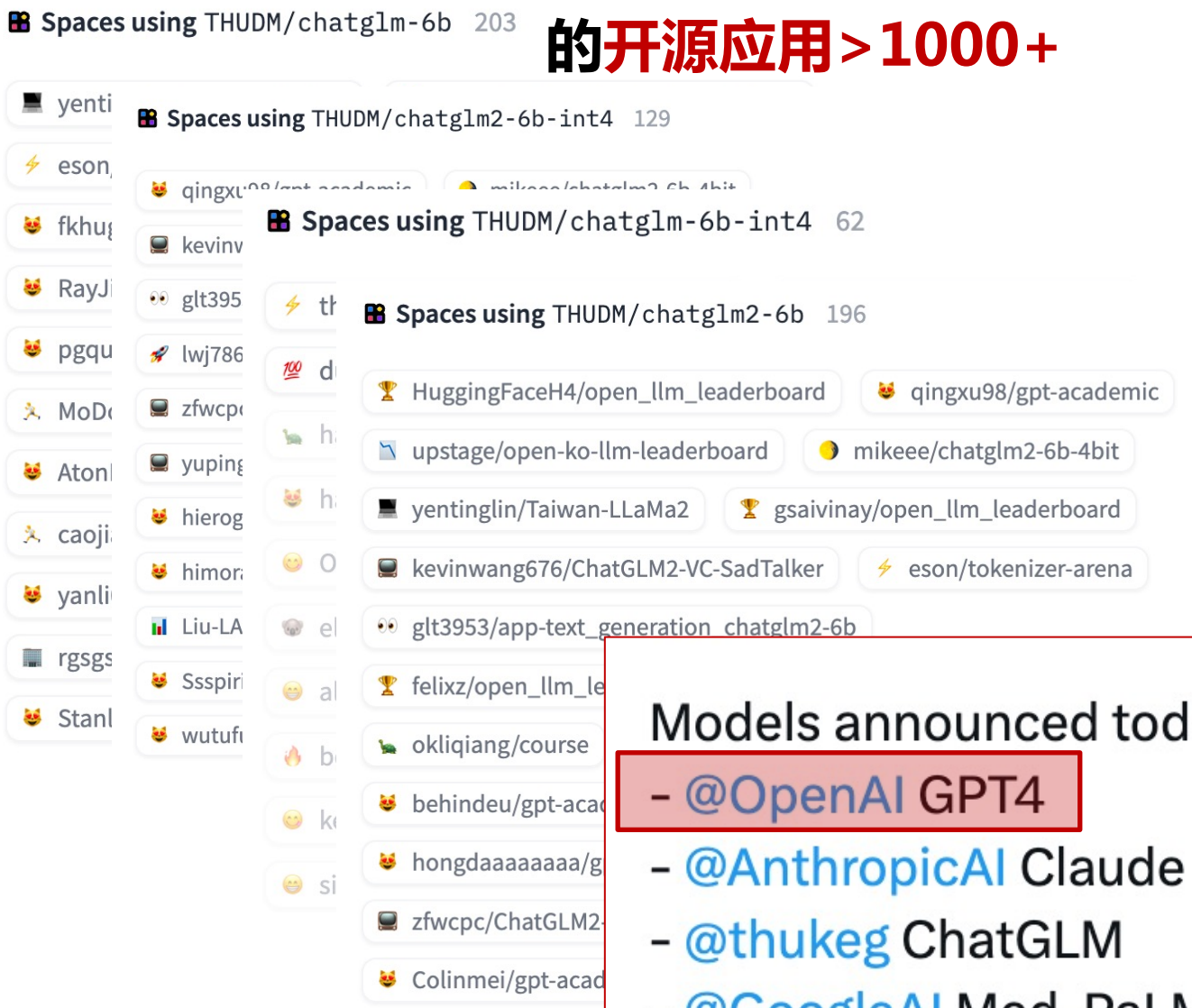
Liang et al., Holistic Evaluation of Language Models. arXiv: **2211.09110**

ChatGLM开源

ChatGLM-6B开源：

- **Mar. 14, 2023**
- GitHub 7万 star
- GitHub Trending榜第一
- Huggingface全球下载超过2000万
- Hugging Face Trending榜第一 (2023年共4周)
- 基于ChatGLM开发的开源应用>1000+

基于ChatGLM开发的开源应用>1000+



https://github.com/THUDM

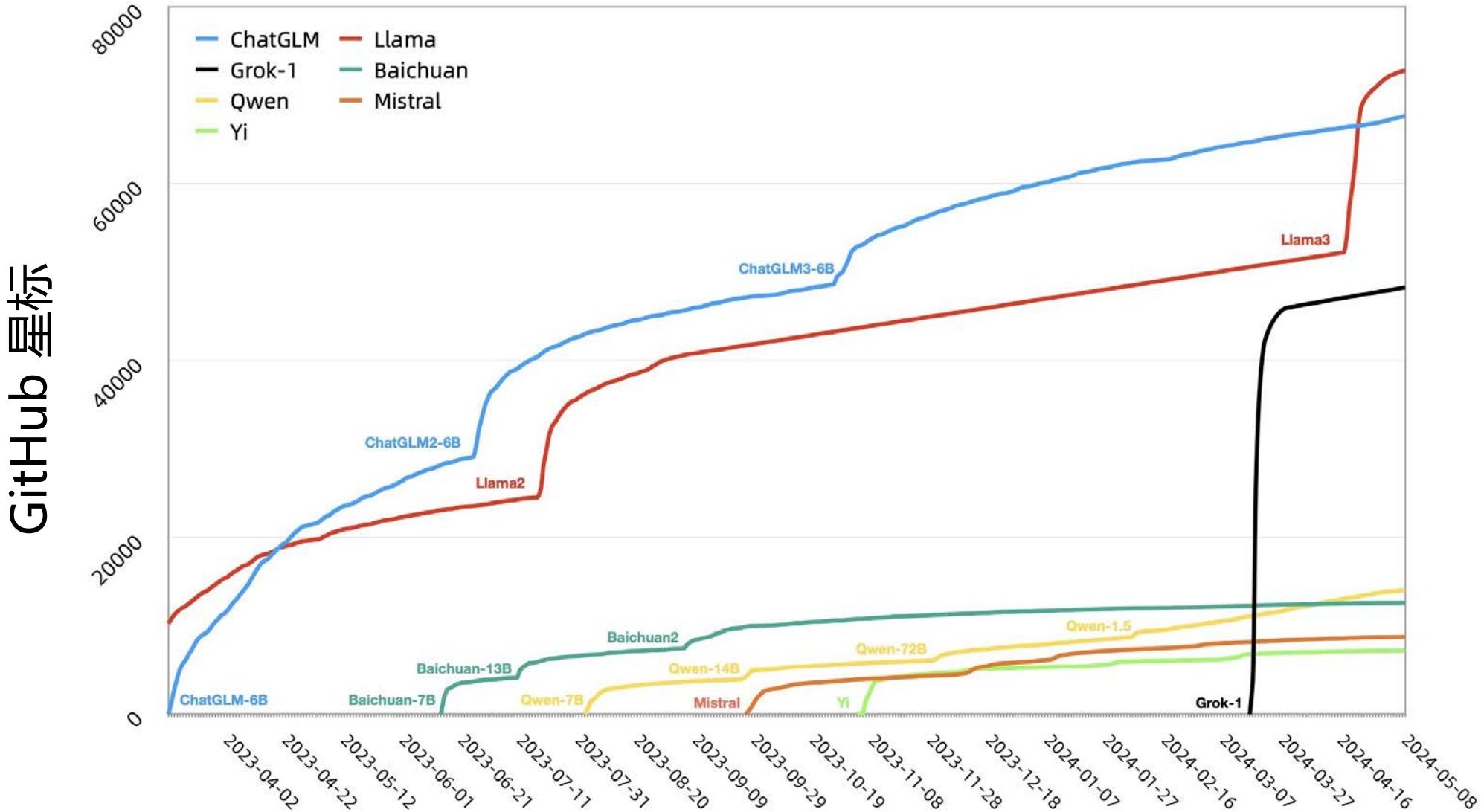
ChatGLM开源

ChatGLM-6B 2023.03.14 → **ChatGLM2-6B** 2023.06.25 → **ChatGLM3-6B** 2023.10.27 → **GLM-4-9B** 2024.06.05

- **GLM-4-9B**：基座模型，不具备对话能力，支持8K上下文
- **GLM-4-9B-Chat**：对话模型，具备工具调用能力，支持长度128K
- **GLM-4-9B-Chat-1M**：对话模型，长文本模型，支持1M上下文，具备工具调用能力
- **GLM-4V-9B**：VQA视觉模型，支持8K上下文。
- **GLM-4-Voice-9B**：端到端中英语音对话模型

Model	中文综合 AlignBench	英文综合 MT-Bench	指令遵从 IFEval	英文学科 MMLU	中文学科 C-Eval	基础算数 GSM8K	数学解题 MATH	基础代码 HumanEval	工程代码 NCB
Llama-3-8B-Instruct	6.40	8.00	68.6	68.4	51.3	79.6	30.0	62.2	24.7
ChatGLM3-6B	5.18	5.50	28.1	61.4	69.0	72.3	25.7	58.5	11.3
GLM-4-9B-Chat	**7.01**	**8.35**	**69.0**	**72.4**	**75.6**	**79.6**	**50.6**	**71.8**	**32.2**

ChatGLM-6B开源



https://github.com/THUDM

Why Large Models?

GPT-1
117M

GPT-2
1.5B

GPT-3
175B

GPT-4
1.8T?

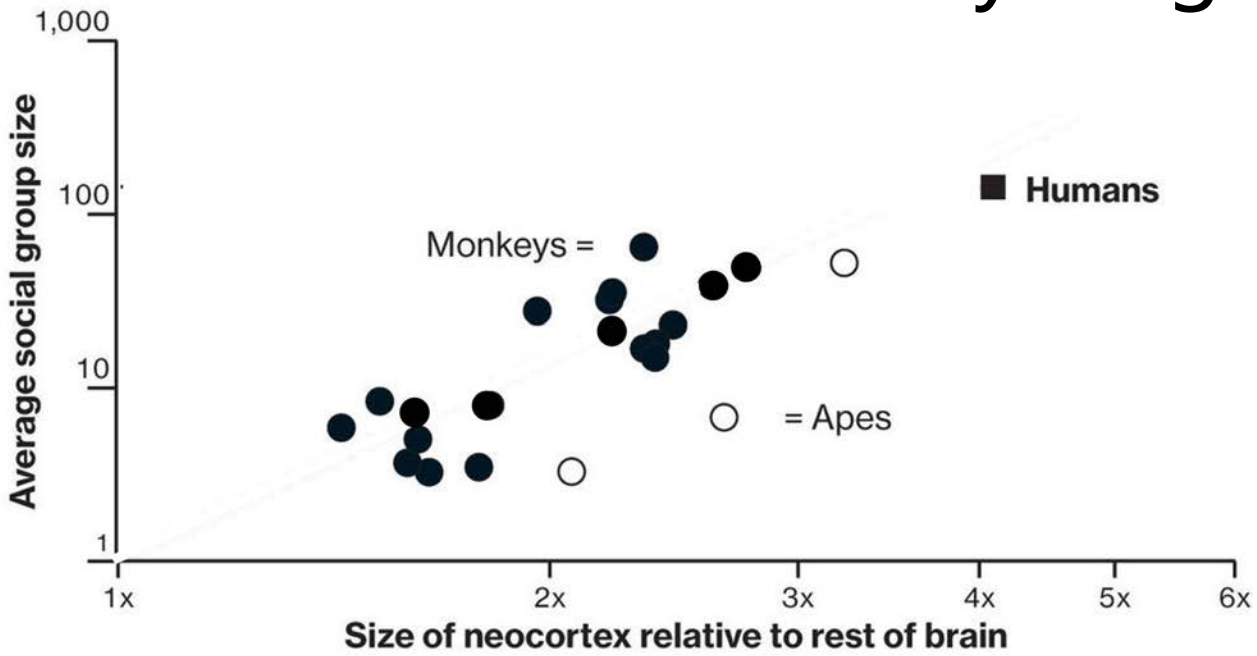
DeepSeek
67B

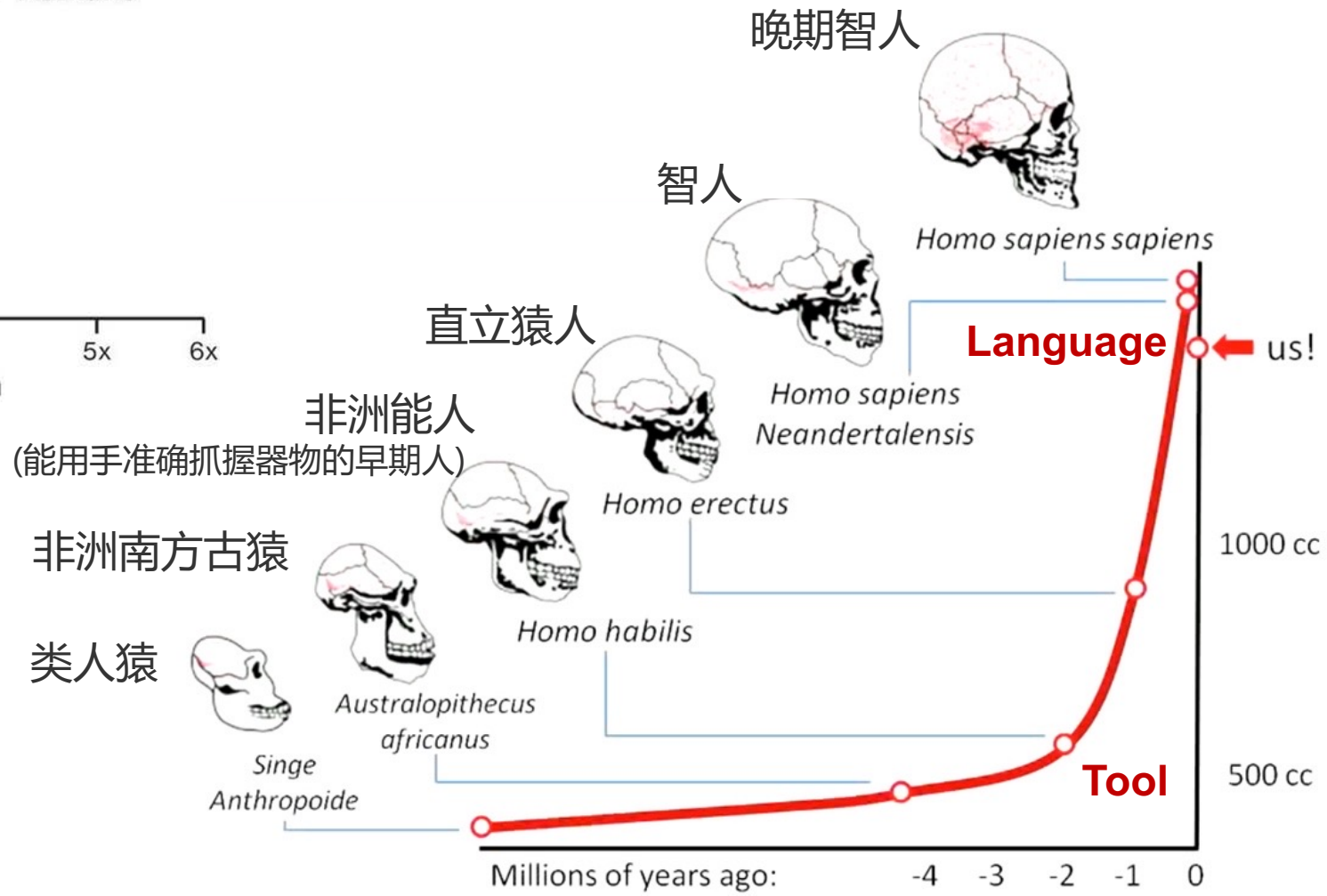
DeepSeek v2
236B (21B A)

DeepSeek v3
671B (37B A)

#parameters

Why Large Models?

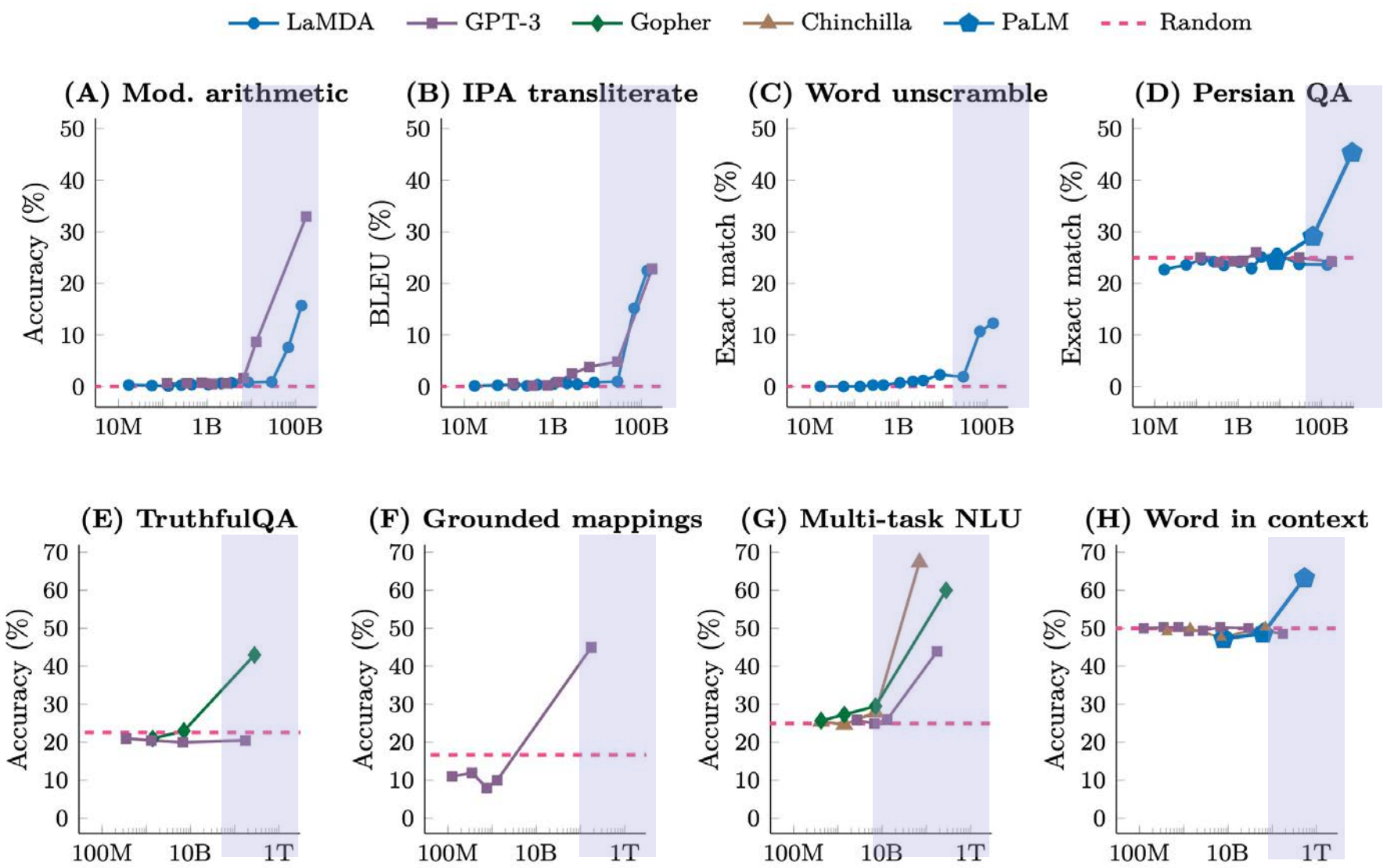




R. I. M. Dunbar, Susanne Shultz. Evolution in the Social Brain. **Science**, Vol. 317, No. 5843, **2007**

Figure Credit: Google

Performance vs. size (model / compute)



1. J Wei, et al. Emergent Abilities of Large Language Models. arXiv: **2206**.07682

Why Large Models?

Pre-train 30+ LLMs of varied **model** (7) and **data** (5) sizes from scratch (300M, 540M, 1B, 1.5B, 3B, 6B, 32B)

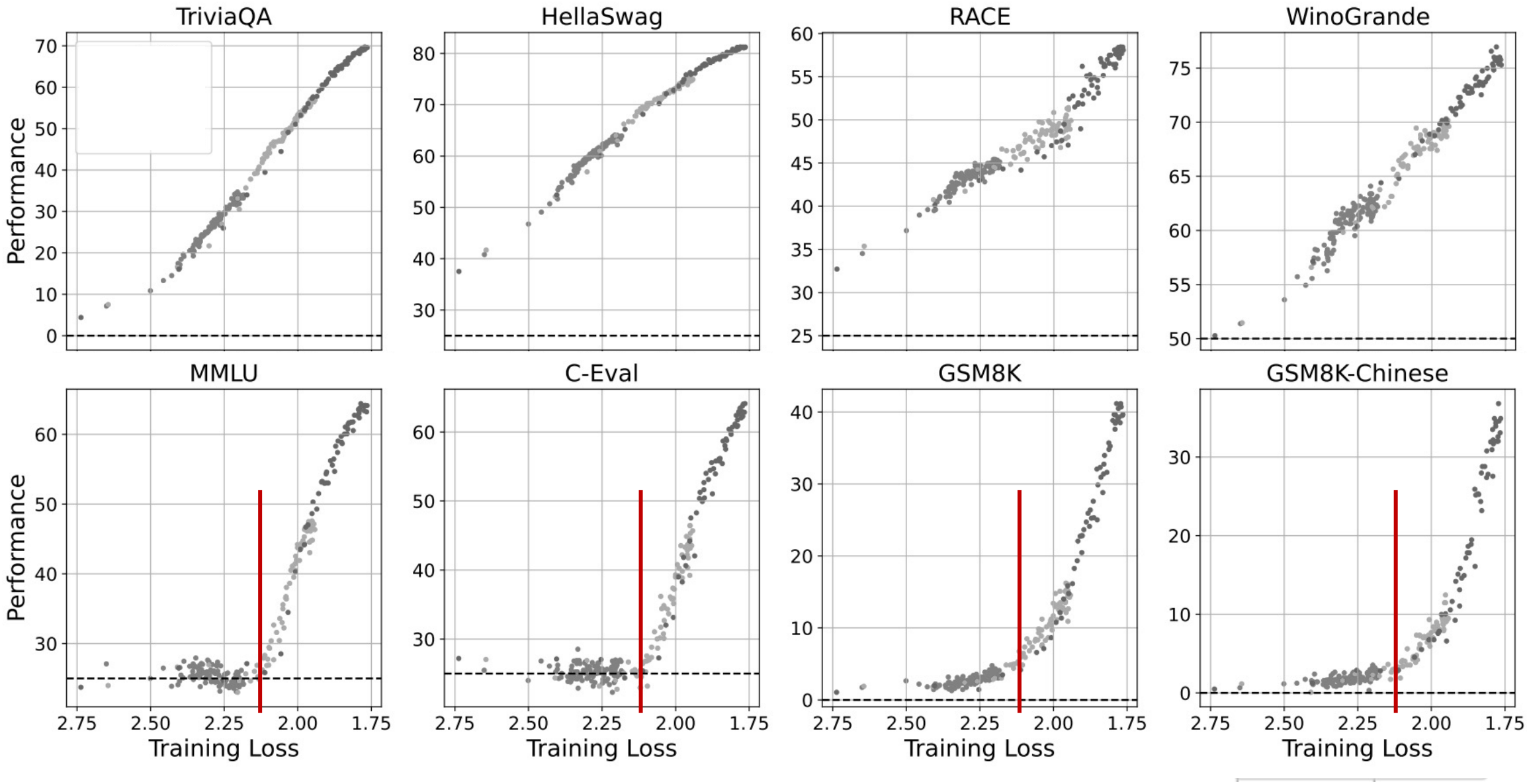
- Fixed data corpus
- Fixed tokenization
- Fixed model architecture

Evaluate downstream performance on 12 diverse datasets

- Different tasks
- Different languages
- Different prompting types
- Different answer forms

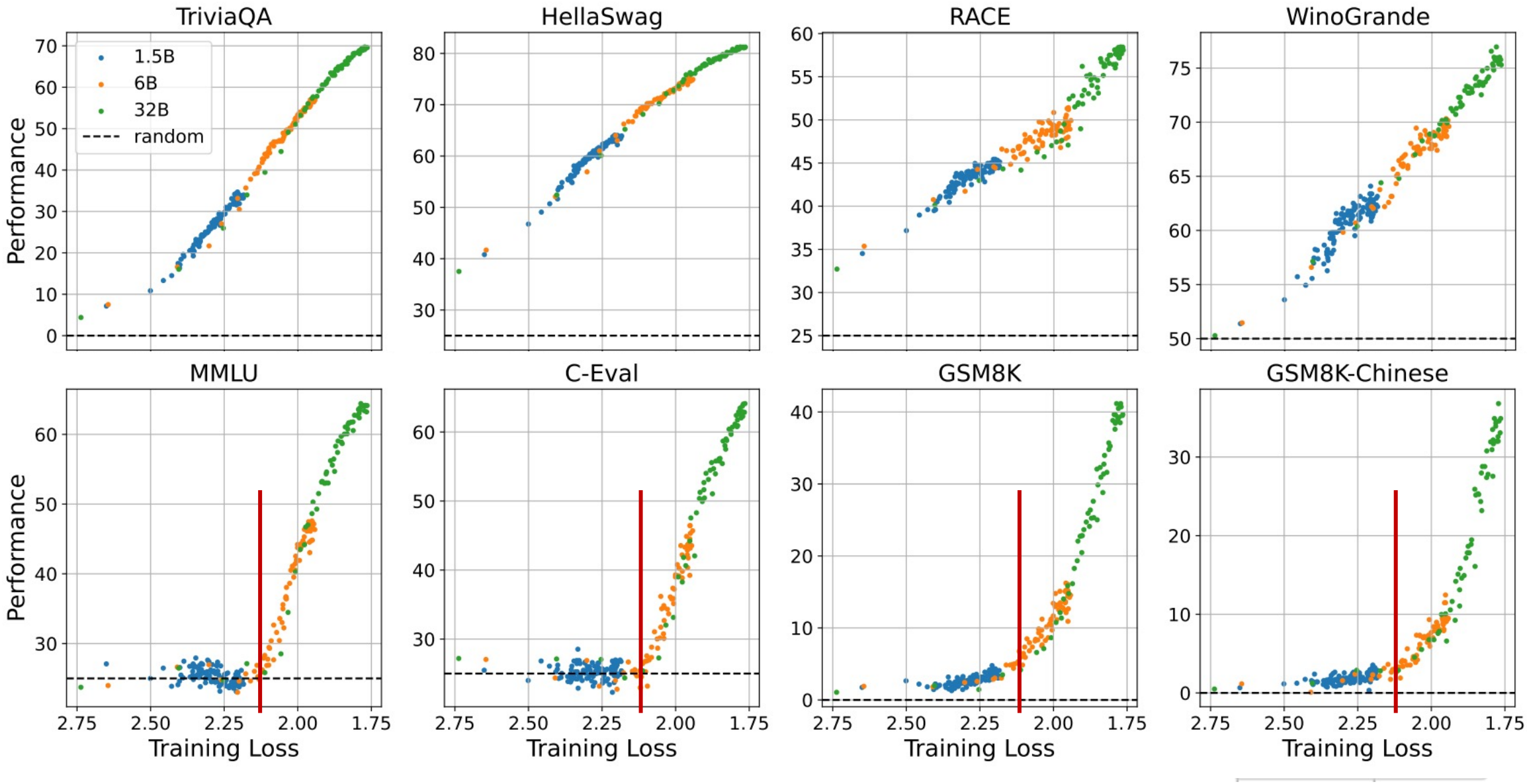
1. Z Du et al. Understanding Emergent Abilities of Language Models from the Loss Perspective. arXiv:2403 NeurIPS'24

Performance vs. **loss** of 1.5B, 6B, 32B models



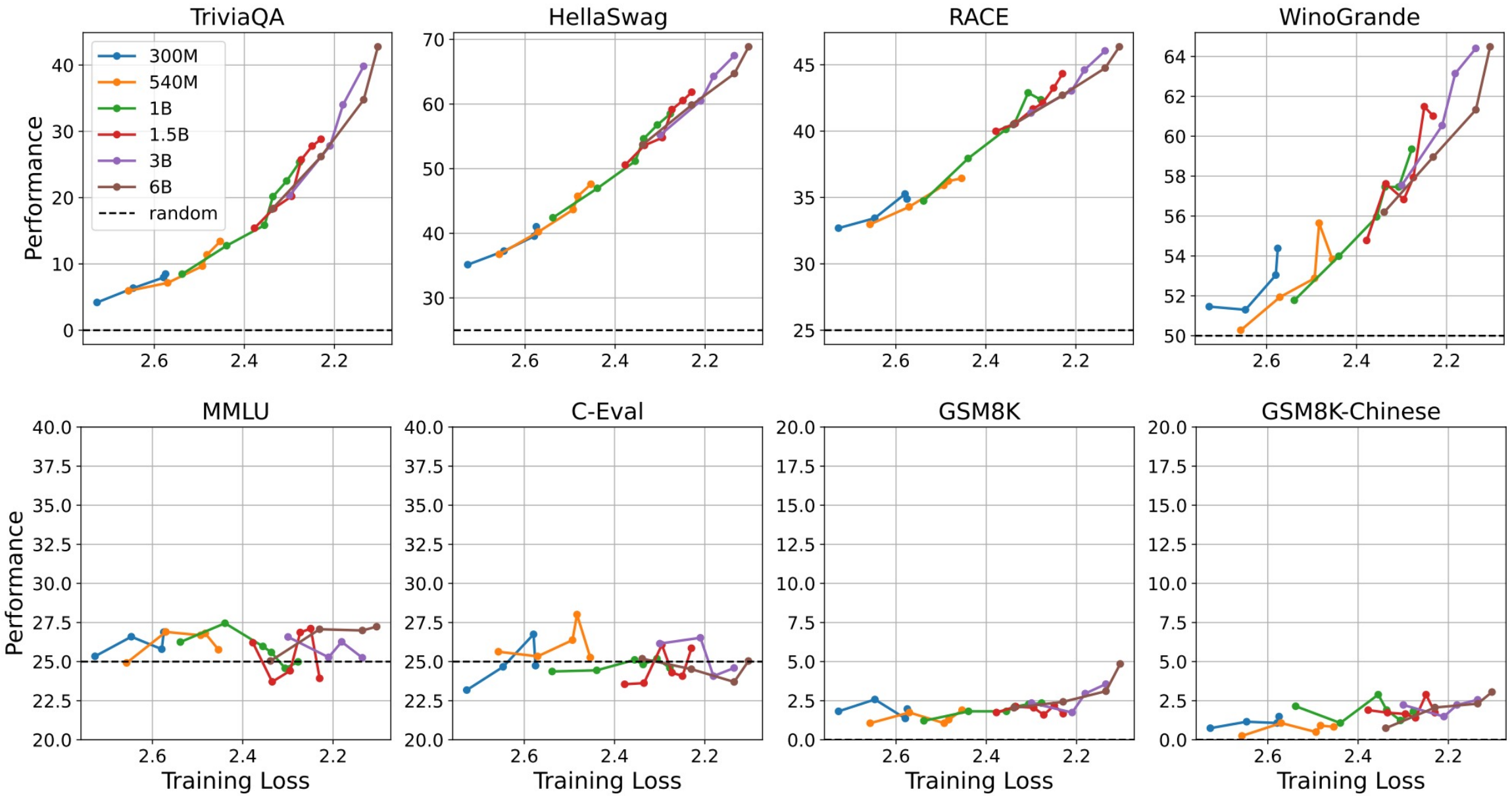
1. Z Du et al. Understanding Emergent Abilities of Language Models from the Loss Perspective. arXiv:2403 NeurIPS'24

Performance vs. **loss** of 1.5B, 6B, 32B models



1. Z Du et al. Understanding Emergent Abilities of Language Models from the Loss Perspective. arXiv:2403 NeurIPS'24

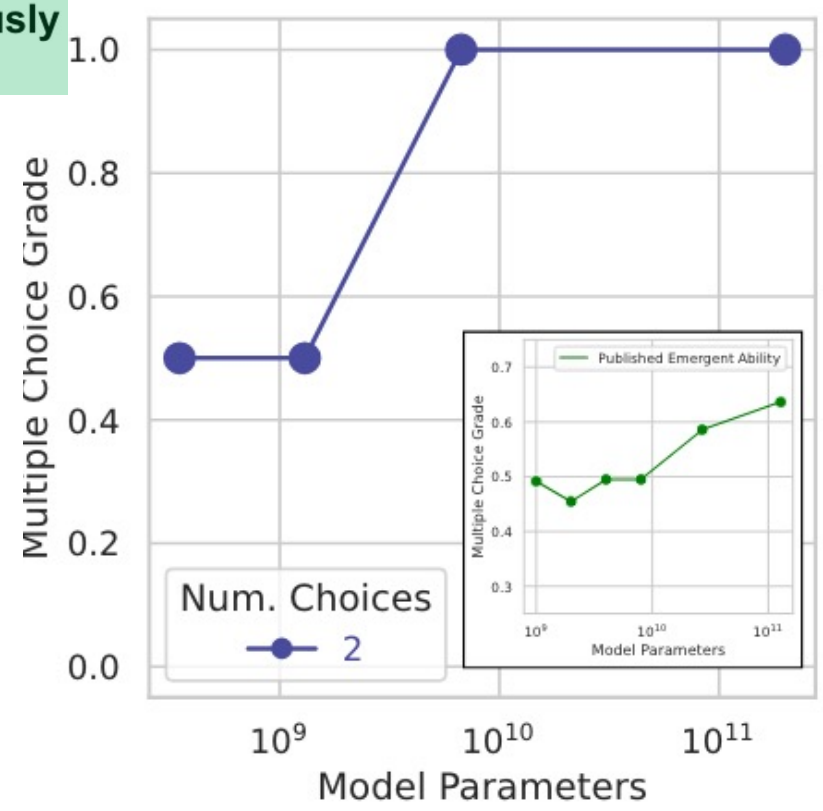
Performance vs. **loss** of smaller models



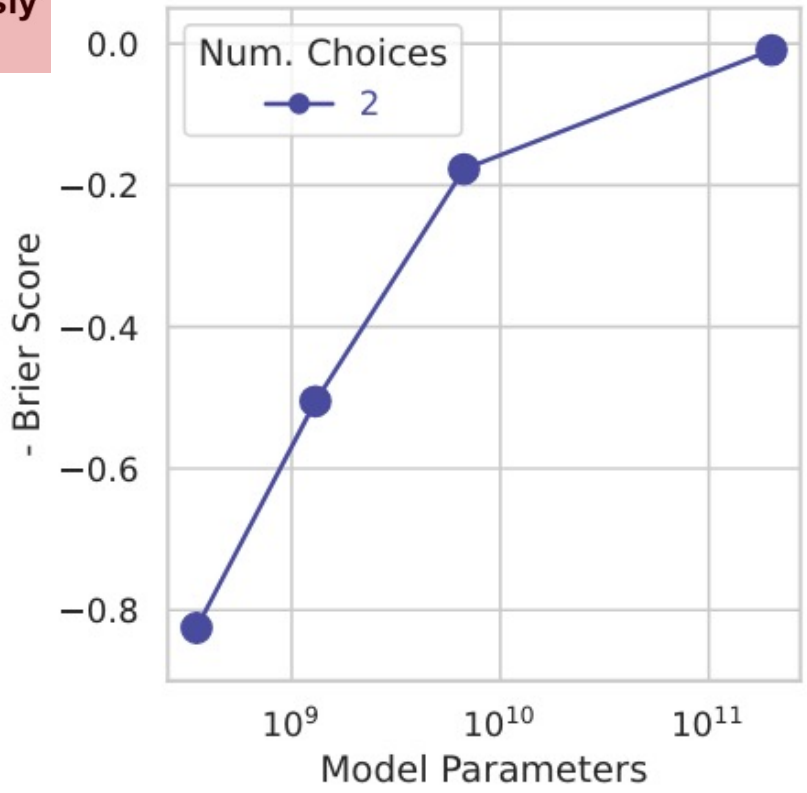
1. Z Du et al. Understanding Emergent Abilities of Language Models from the Loss Perspective. arXiv:2403 NeurIPS'24

"emergent abilities are created by the researcher's choice of metrics,
not fundamental changes in model family behavior on specific tasks with scale."

Discontinuously score



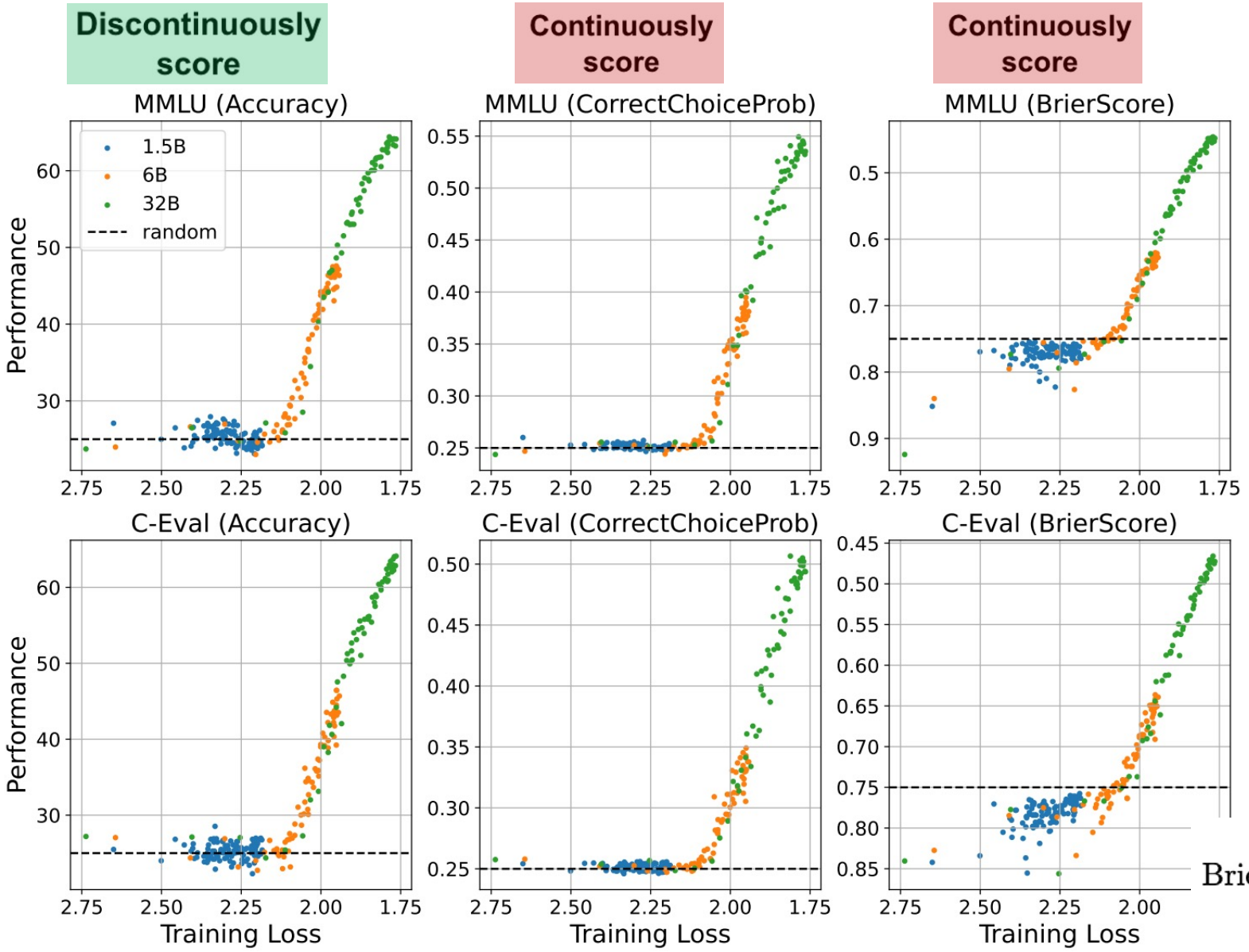
Continuously score



$$\text{BrierScore} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C}(y_{ij} - \hat{y}_{ij})^2$$

1. Schaeffer, et al., NeurIPS'23 Best Paper

Performance (different metrics) vs. **loss**



$$\text{BrierScore} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C}(y_{ij} - \hat{y}_{ij})^2$$

1. Z Du et al. Understanding Emergent Abilities of Language Models from the Loss Perspective. **arXiv:2403.15796**

Emergent Abilities defined by **loss**

The normalized performance on an emergent ability as a function of the pre-training loss L is:

$$\begin{cases} f(L) & \text{if } L < \eta \\ 0 & \text{otherwise} \end{cases}$$

Combined with the model scaling law, we can get the normalized performance as a function of the model size N

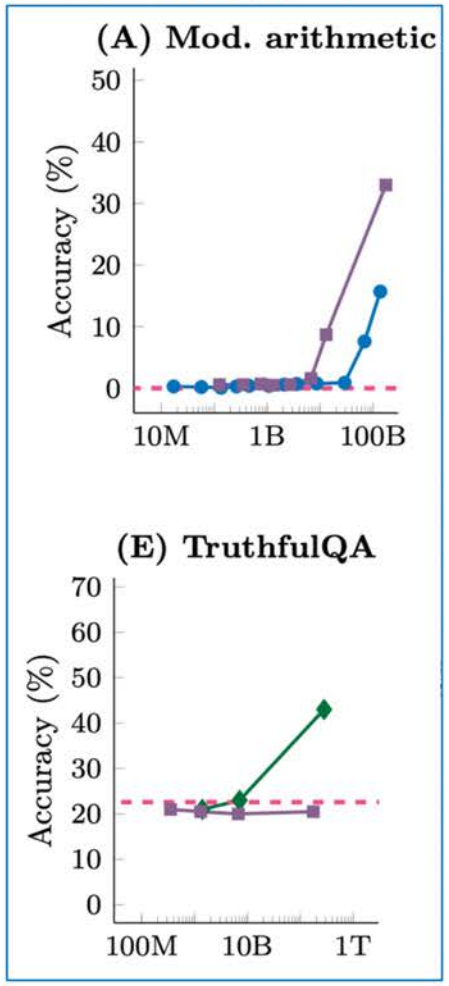
$$\begin{cases} f\left(L_\infty + \left(\frac{N_0}{N}\right)^{\alpha_N}\right) & \text{if } N \geq N_0(\eta - L_\infty)^{-\frac{1}{\alpha^N}} \\ 0 & \text{otherwise} \end{cases}$$

From this equation, we can explain the observed emergent abilities with model sizes.

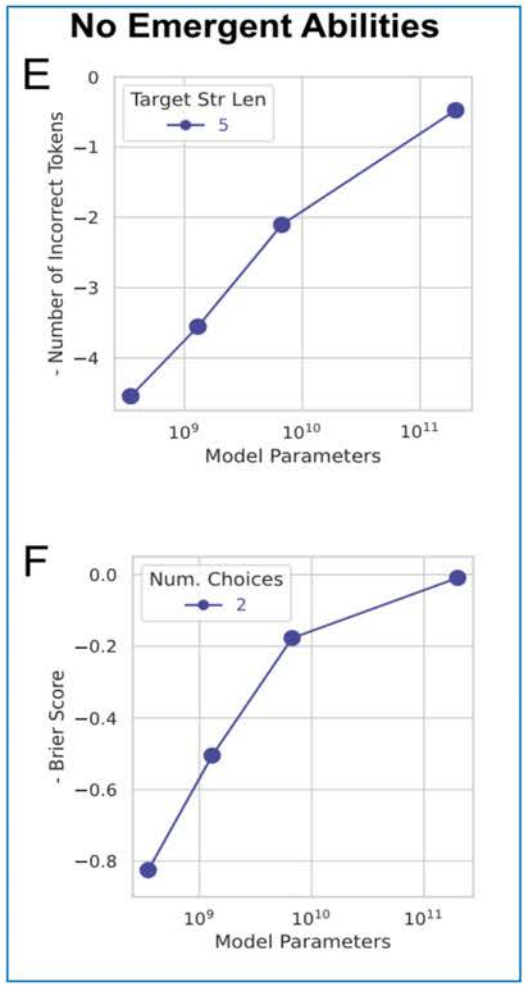
Z Du et al. Understanding Emergent Abilities of Language Models from the Loss Perspective. NeurIPS'24

Emergent Abilities defined by **loss**

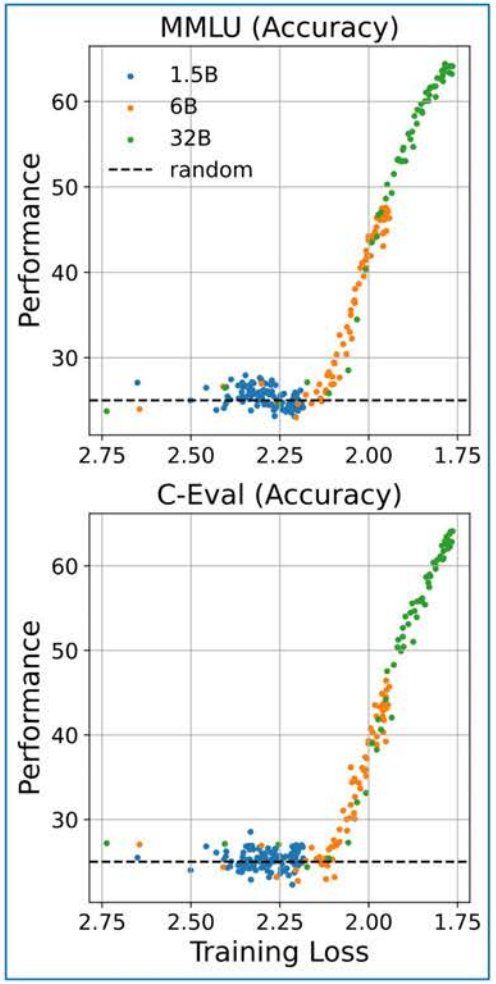
J Wei, et al. Emergent Abilities of Large Language Models. TMLR, 2022.



Schaeffer, et al., Are Emergent Abilities of Large Language Models a Mirage? NeurIPS'23



Du, et al., Understanding Emergent Abilities of Language Models from the Loss Perspective? NeurIPS'24



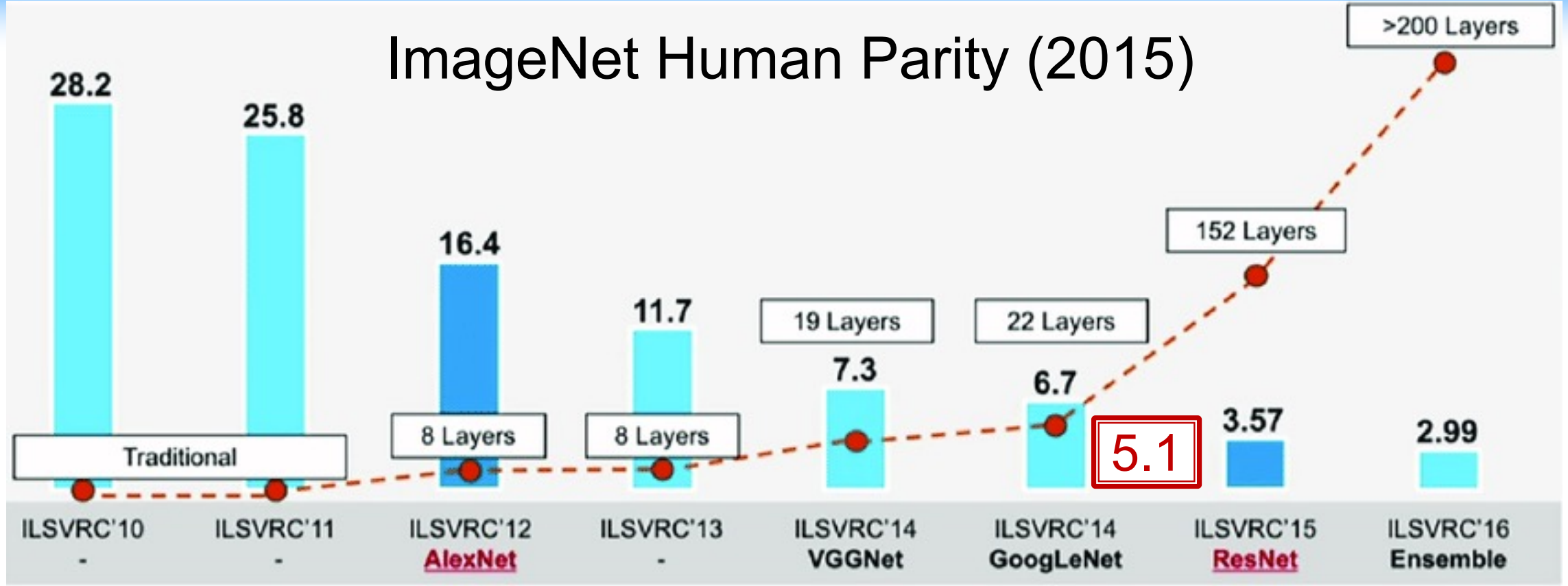
Why Large Models?

- 对话

- 数学题
- 代码题
- 物化生题

- 能看
- 能听
- 能说

ImageNet Human Parity (2015)



PC: Web

2023.3.14 GPT-4

What is funny about this image? Describe it panel by panel.

Source: https://www.reddit.com/r/hmmm/comments/ubab5v/hmmm/

GPT-4

The image shows a package for a "Lightning Cable" adapter with three panels.

Panel 1: A smartphone with a VGA connector (a large, blue, 15-pin connector typically used for computer monitors) plugged into its charging port.

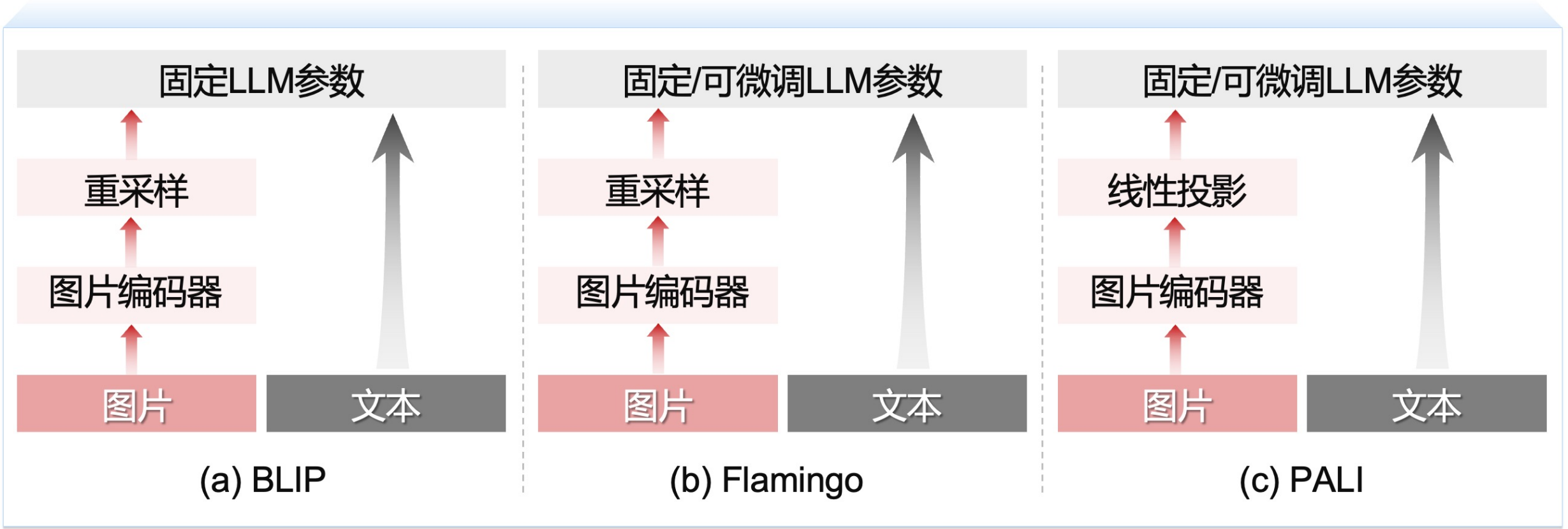
Panel 2: The package for the "Lightning Cable" adapter with a picture of a VGA connector on it.

Panel 3: A close-up of the VGA connector with a small Lightning connector (used for charging iPhones and other Apple devices) at the end.

多模态理解模型

Wang et al. CogVLM: visual experts for large language models. NeurIPS'24

多模态理解模型VLM



Li, et al. (Salesforces)
arXiv:**2201**.12086

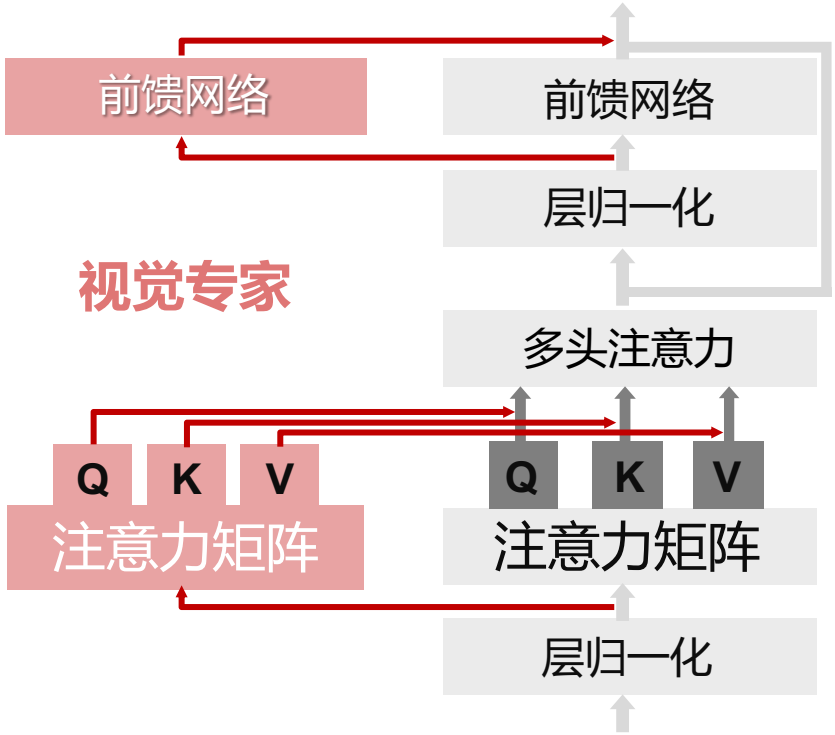
Alayrac, et al. (Google)
arXiv:**2204**.14198

Chen, et al. (Google)
arXiv:**2209**.06794

- Shallow feature alignment between image and text
- Fit into text space, while image spaces may have different features from text

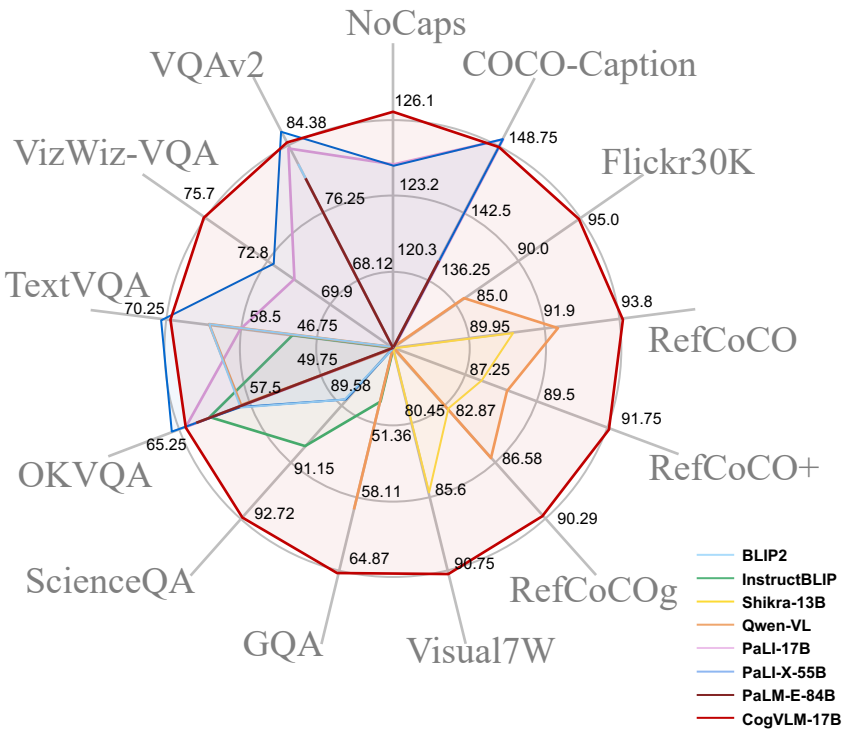
多模态理解模型CogVLM

跨模态对齐语言与视觉信号



提出**可训练视觉专家模块**，在自注意力层深度对齐语言与视觉

跨模态理解认务



VQA等14项多模态理解基准

Wang et al. CogVLM: visual experts for large language models. NeurIPS'24

多模态理解模型CogVLM

World Knowledge

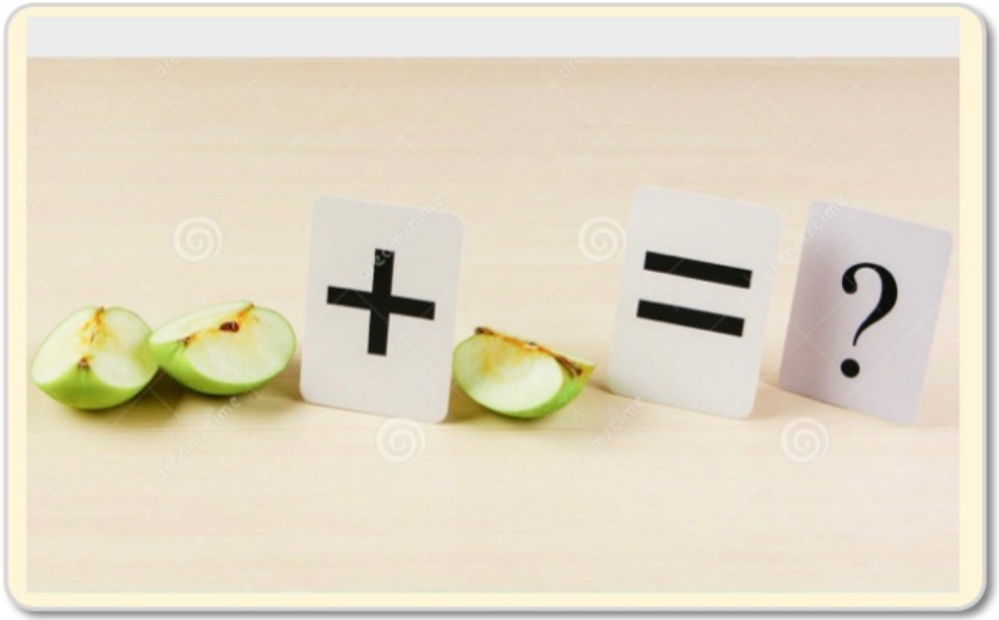
Q: How many goals did the person in the picture score in the 2018 World Cup?

CogVLM: The person in the picture is Cristiano Ronaldo. He is the captain of the Portuguese team and has participated in the 2018 World Cup. In the 2018 World Cup, Cristiano Ronaldo scored 4 goals.

Wang et al. CogVLM: visual experts for large language models. NeurIPS'24

多模态理解模型CogVLM

Visual Math Problem Solving



Q1: How to solve it?

CogVLM: The problem in the picture is a simple math problem, where a sliced green apple is equated to a question mark. Specifically, in the middle of the picture, there are three green apples placed side by side and three cardboard signs. The signs have black characters on them: a plus sign, an equal sign, and a question mark. The equation shown in the picture is: 2 + 1 = ?

Q2: What's the final answer?

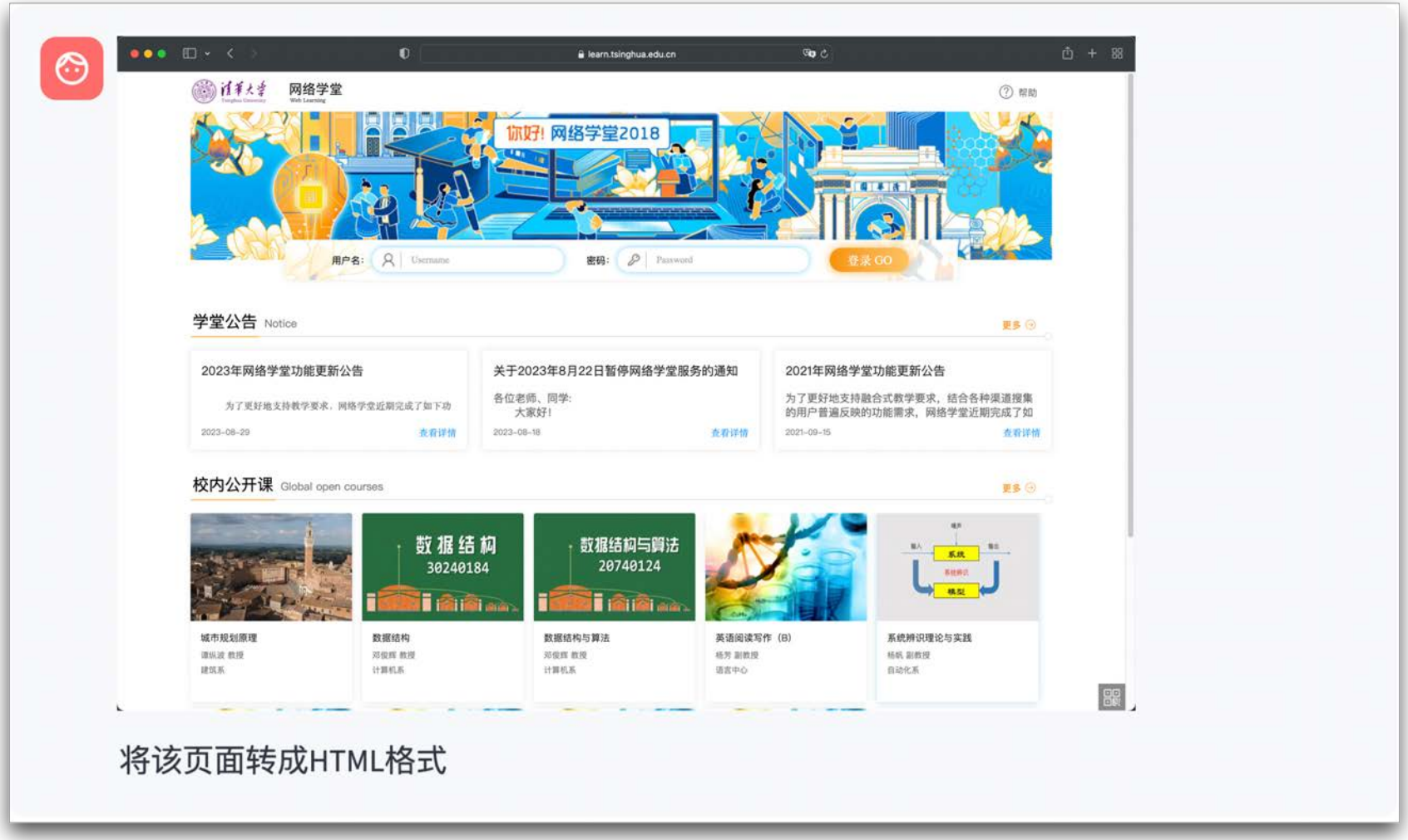
CogVLM: 2+1=3

Wang et al. CogVLM: visual experts for large language models. NeurIPS'24

多模态理解模型CogVLM

Wang et al. CogVLM: visual experts for large language models. NeurIPS'24

多模态理解模型CogVLM



Wang et al. CogVLM: visual experts for large language models. NeurIPS'24

多模态理解模型CogVLM

```html
<html lang="zh-CN">
<head>
    <meta charset="UTF-8">
    <title>清华大学网络学堂</title>
    <!-- 引入样式文件 -->
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <!-- 导航栏 -->
    <header class="navbar">
        <div class="logo">清华大学网络学堂</div>
        <nav>
            <ul>
                <li><a href="#">课程</a></li>
                <li><a href="#">公告</a></li>
                <li><a href="#">个人中心</a></li>
                <li><a href="#">帮助</a></li>
            </ul>
        </nav>
    </header>

    <!-- 登录框 -->
    <section class="login-box">
        <h2>你好！欢迎来到网络学堂2018</h2>
        <form action="#">
            <input type="text" placeholder="用户名：Username" required>
            <input type="password" placeholder="密码：Password" required>
            <button type="submit">登录GO</button>
        </form>
    </section>

    <!-- 公告区域 -->
    <section class="notice-section">
        <h3>学堂公告 Notice</h3>
        <article>
            <h4>2023年网络学堂功能更新公告</h4>
            <p>为了更好地支持教学要求，网络学堂近期完成了如下功...</p>
            <button>查看详情</button>
        </article>
        <article>
            <h4>关于2023年8月22日暂停网络学堂服务的通知</h4>
            <p>各位老师、同学：大家好！...</p>
            <button>查看详情</button>
        </article>
        <article>
            <h4>2021年网络学堂功能更新公告</h4>
            <p>为了更好地支持融合式教学要求，结合各种渠道搜集的用户普遍反映的功能需求，</p>
            <button>查看详情</button>
        </article>
    </section>

    <!-- 校内公开课区域 -->
    <section class="open-courses-section">
        <h3>校内公开课 Global open courses</h3>
        <div class="course-card">
            <img src="course-image.jpg" alt="城市规划原理">
            <h5>城市规划原理</h5>
            <p>谭纵波 教授 建筑系</p>
        </div>
        <div class="course-card">
            <img src="course-image.jpg" alt="数据结构">
            <h5>数据结构</h5>
            <p>邓俊辉 教授 计算机系</p>
        </div>
        <div class="course-card">
            <img src="course-image.jpg" alt="数据结构与算法">
            <h5>数据结构与算法</h5>
            <p>邓俊辉 教授 计算机系</p>
        </div>
        <div class="course-card">
            <img src="course-image.jpg" alt="英语阅读写作（B）">
            <h5>英语阅读写作（B）</h5>
            <p>杨芳副教授 语言中心</p>
        </div>
        <div class="course-card">
            <img src="course-image.jpg" alt="系统辨识理论与实践">
            <h5>系统辨识理论与实践</h5>
            <p>杨帆副教授 自动化系</p>
        </div>
    </section>

    <!-- 页脚 -->
    <footer class="footer">
        &copy; 2023 清华大学网络学堂 版权所有
    </footer>
</body>
</html>
```

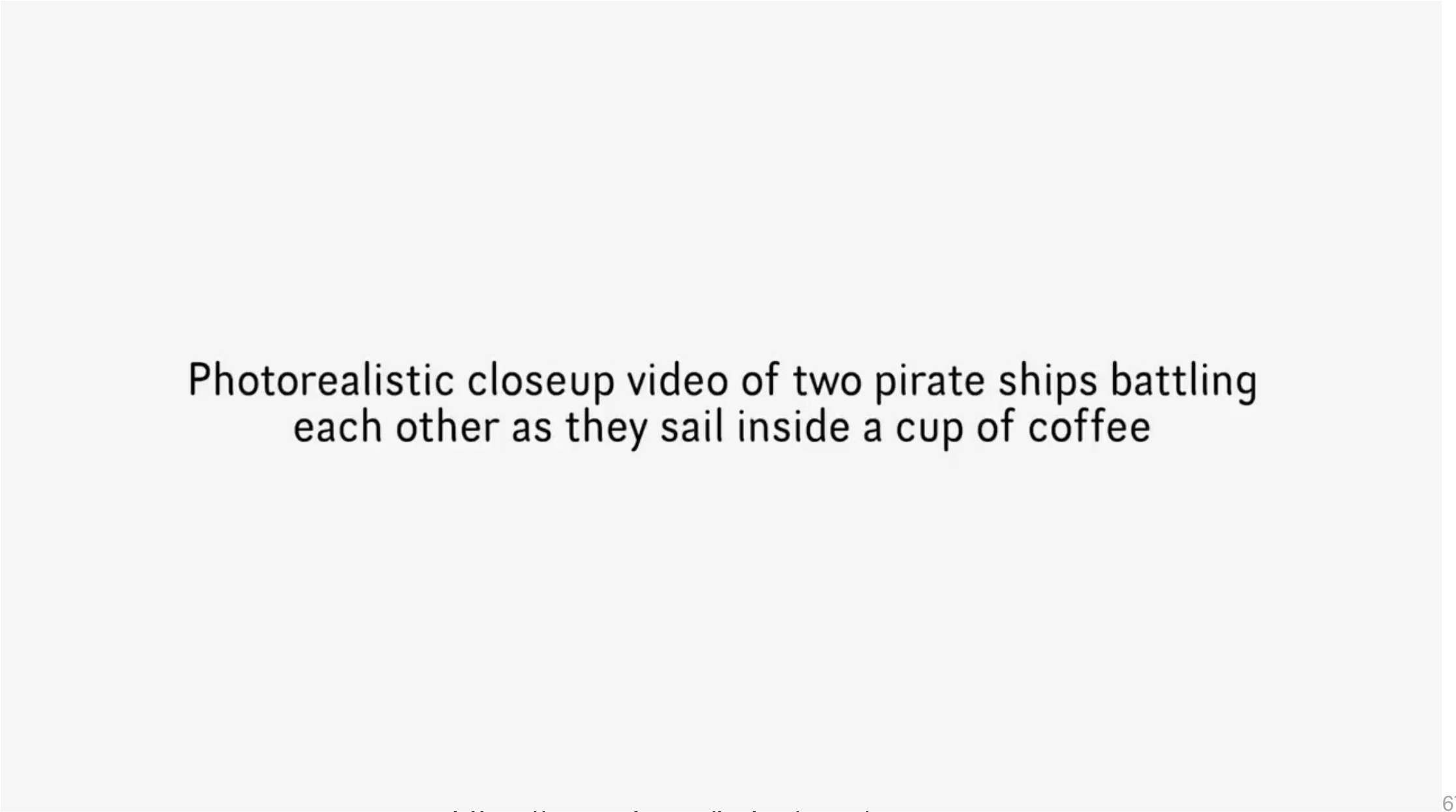
请注意，上述代码仅为示例，实际网页可能包含更多细节和交互功能。您可以根据需要调整样式和内容。

Wang et al. CogVLM: visual experts for large language models. NeurIPS'24

2024.02 Sora

https://openai.com/index/sora/

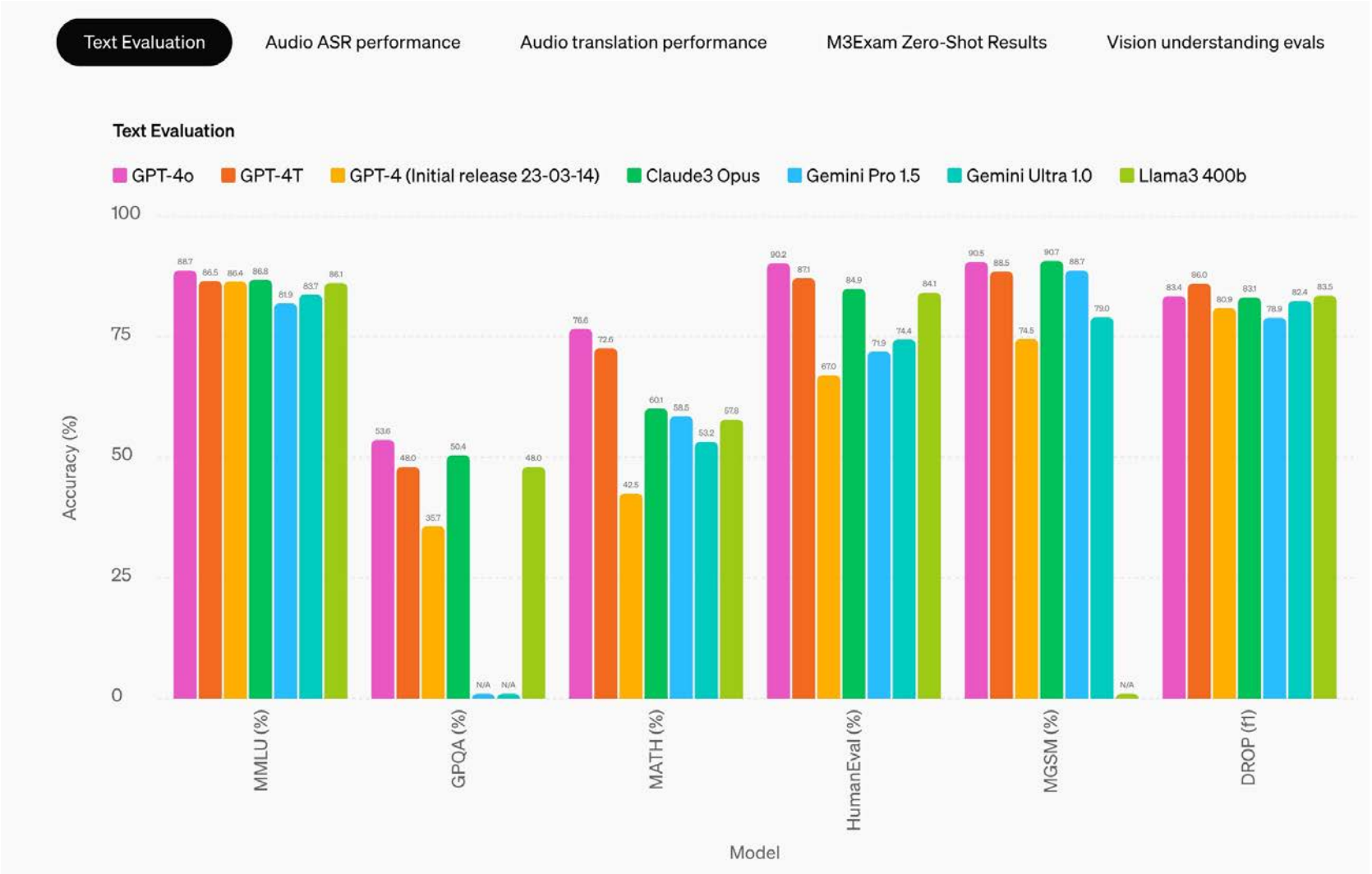
2024.02 Sora



https://openai.com/index/sora/

2024.05 GPT-4o

"GPT-4o, our new flagship model that can reason across audio, vision, and text in real time."



https://openai.com/index/hello-gpt-4o/

Why Large Models?

语言 → **推理** → **视觉**

- 对话

- 数学题
- 代码题
- 物化生题

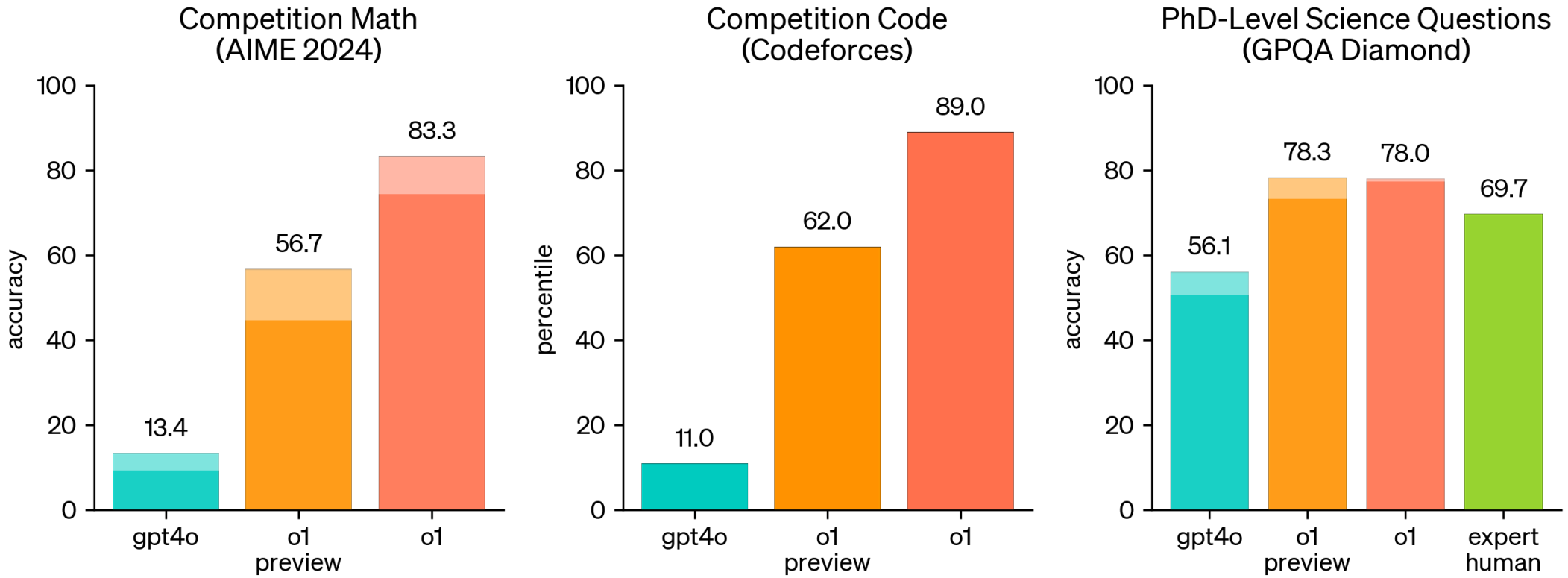
- 围棋

- 能看
- 能听
- 能说

“用计算模型可描述的认知问题，AI很快超过人类水平”

2024.09 o1

“OpenAI o1 ranks in the 89th percentile on competitive programming questions (Codeforces), places **among the top 500 students in the US in a qualifier for the USA Math Olympiad (AIME)**, and exceeds **human PhD-level accuracy** on a benchmark of physics, biology, and chemistry problems (GPQA). ”



https://openai.com/index/learning-to-reason-with-llms/

Chain of Thought (CoT)

oyfjdnisdr rtqwainr acxz mynzbhhx -> Think step by step

Use the example above to decode:

oyekaijzdf aaptcg suaokybhai ouow aqht mynznvaatzacdfoulxxz

Chain of Thought (CoT)

Thought for 5 seconds

To decode the given ciphertext using the example provided, we'll use the pattern identified in the example.

Ciphertext Example:

```
PlainText
1  oyfjdnisdr rtqwainr acxz mynzbhhx
```

Decoded as:

```
PlainText
1  Think step by step
```

Decoding Method:

1. **Pair the letters** in the ciphertext.
2. **Convert each letter to its numerical position** in the alphabet (A=1, B=2, ..., Z=26).
3. **Sum the numerical values** of each pair.
4. **Compute the average** of the sum (divide by 2).
5. **Convert the average back to a letter** (1=A, 2=B, ..., 26=Z).

Apply this method to the given ciphertext:

```
PlainText
1  oyekaijzdf aaptcg suaokybhai ouow aqht mynzn
```

Break down the ciphertext into pairs:

1. **First word: "oyekaijzdf"**
   a. Pairs: oy, ek, ai, jz, df
   b. Decoded letters:
      i. oy → (15+25)/2 = 20 → T
      ii. ek → (5+11)/2 = 8 → H
      iii. ai → (1+9)/2 = 5 → E
      iv. jz → (10+26)/2 = 18 → R
   c. df → (4+6)/2 = 5 → E
   d. Decoded word: **THERE**
2. **Second word: "aaptcg"**
   a. Pairs: aa, pt, cg
   b. Decoded letters:
      i. aa → (1+1)/2 = 1 → A
      ii. pt → (16+20)/2 = 18 → R
      iii. cg → (3+7)/2 = 5 → E
      iv. Decoded word: **ARE**
3. **Third word: "suaokybhai"**
   a. Pairs: su, ao, ky, bh, ai
   b. Decoded letters:
      i. su → (19+21)/2 = 20 → T
      ii. ao → (1+15)/2 = 8 → H
      iii. ky → (11+25)/2 = 18 → R
      iv. bh → (2+8)/2 = 5 → E
      v. ai → (1+9)/2 = 5 → E
   c. Decoded word: **THREE**
4. **Fourth word: "ouow"**
   a. Pairs: ou, ow
   b. Decoded letters:
      i. ou → (15+21)/2 = 18 → R
      ii. ow → (15+23)/2 = 19 → S
   c. Decoded letters: **R'S**
5. **Fifth word: "aqht"**
   a. Pairs: aq, ht
   b. Decoded letters:
      i. aq → (1+17)/2 = 9 → I
      ii. ht → (8+20)/2 = 14 → N
   c. Decoded word: **IN**
6. **Sixth word: "mynznvaatzacdfoulxxz"**
   a. Pairs: my, nz, nv, aa, tz, ac, df, ou, lx, xz
   b. Decoded letters:
      i. my → (13+25)/2 = 19 → S
      ii. nz → (14+26)/2 = 20 → T
      iii. nv → (14+22)/2 = 18 → R
      iv. aa → (1+1)/2 = 1 → A
      v. tz → (20+26)/2 = 23 → W
      vi. ac → (1+3)/2 = 2 → B
      vii. df → (4+6)/2 = 5 → E
      viii. ou → (15+21)/2 = 18 → R
      ix. lx → (12+24)/2 = 18 → R
      x. xz → (24+26)/2 = 25 → Y
   c. Decoded word: **STRAWBERRY**

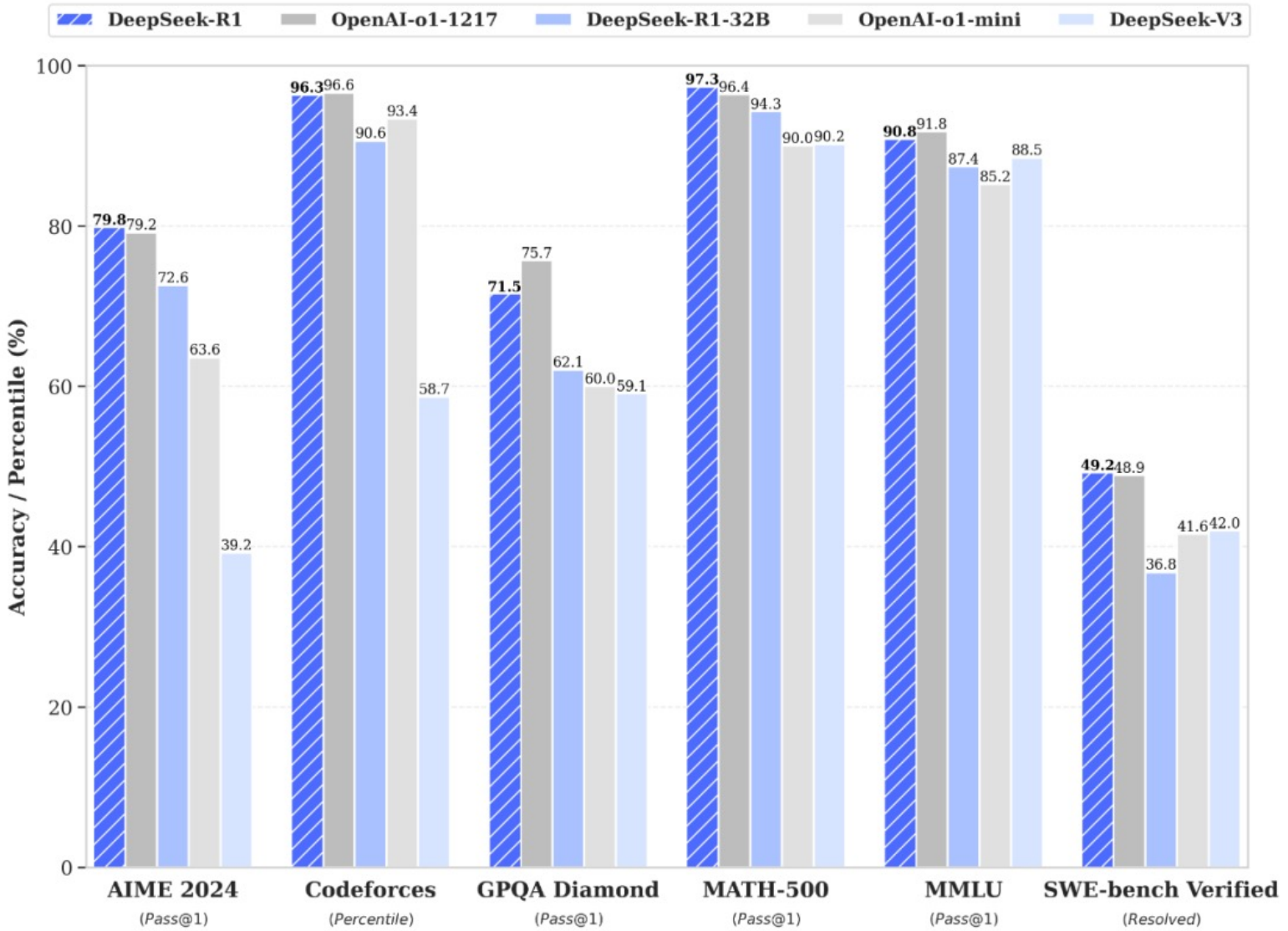
Final Decoded Message:

```
PlainText
1  THERE ARE THREE R'S IN STRAWBERRY
```

Answer:
THERE ARE THREE R'S IN STRAWBERRY

2025.01 DeepSeek R1

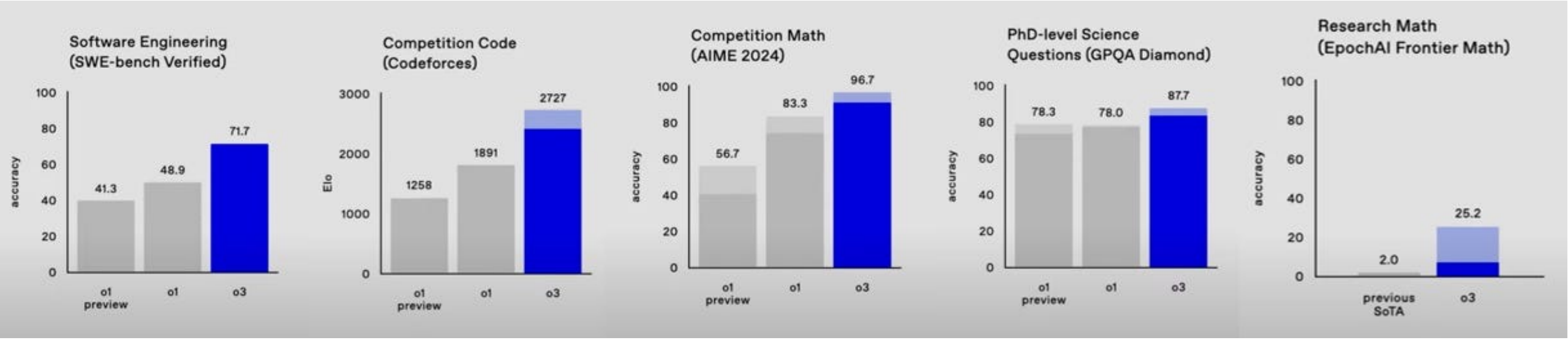
- Performance on par with OpenAI-o1
- Fully open-source model & technical report
- DeepSeek-R1-Zero
- As a result, open discovery, open methods, and open CoTs



https://api-docs.deepseek.com/news/news250120

2024.12 o3

“The new OpenAI model o3 scores better than 99.8% of competitive coders on Codeforces, with a score of 2727, which is equivalent to the #175 best human competitive coder on the planet. ”

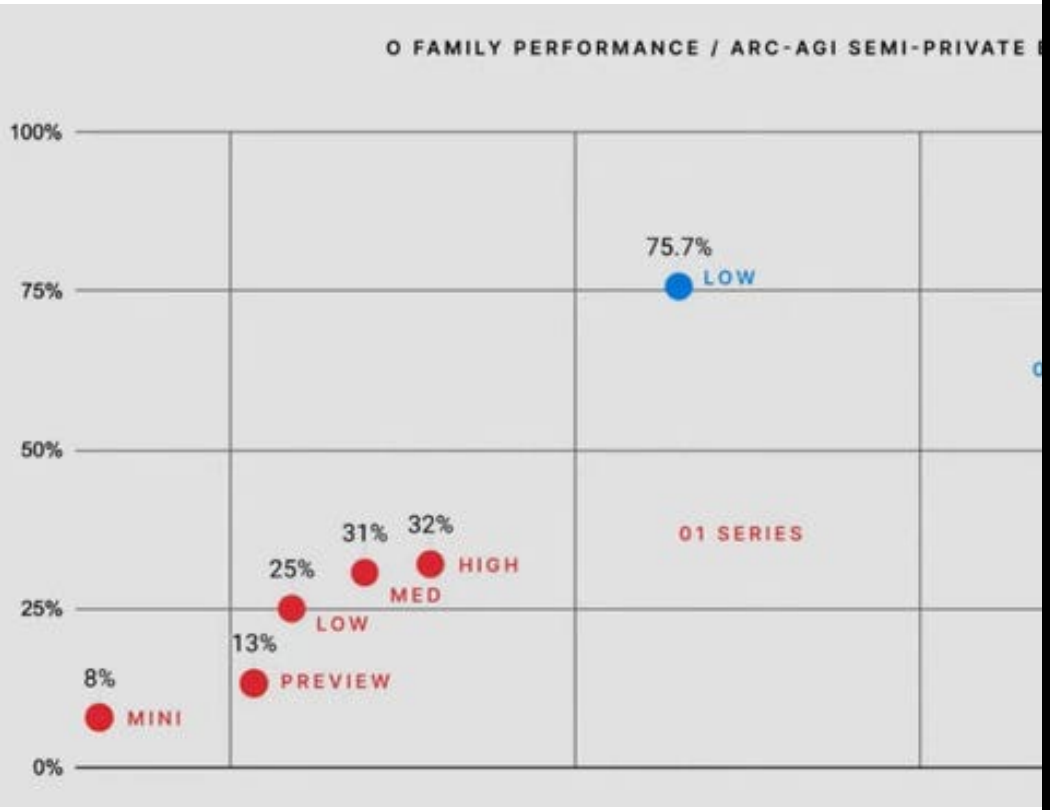


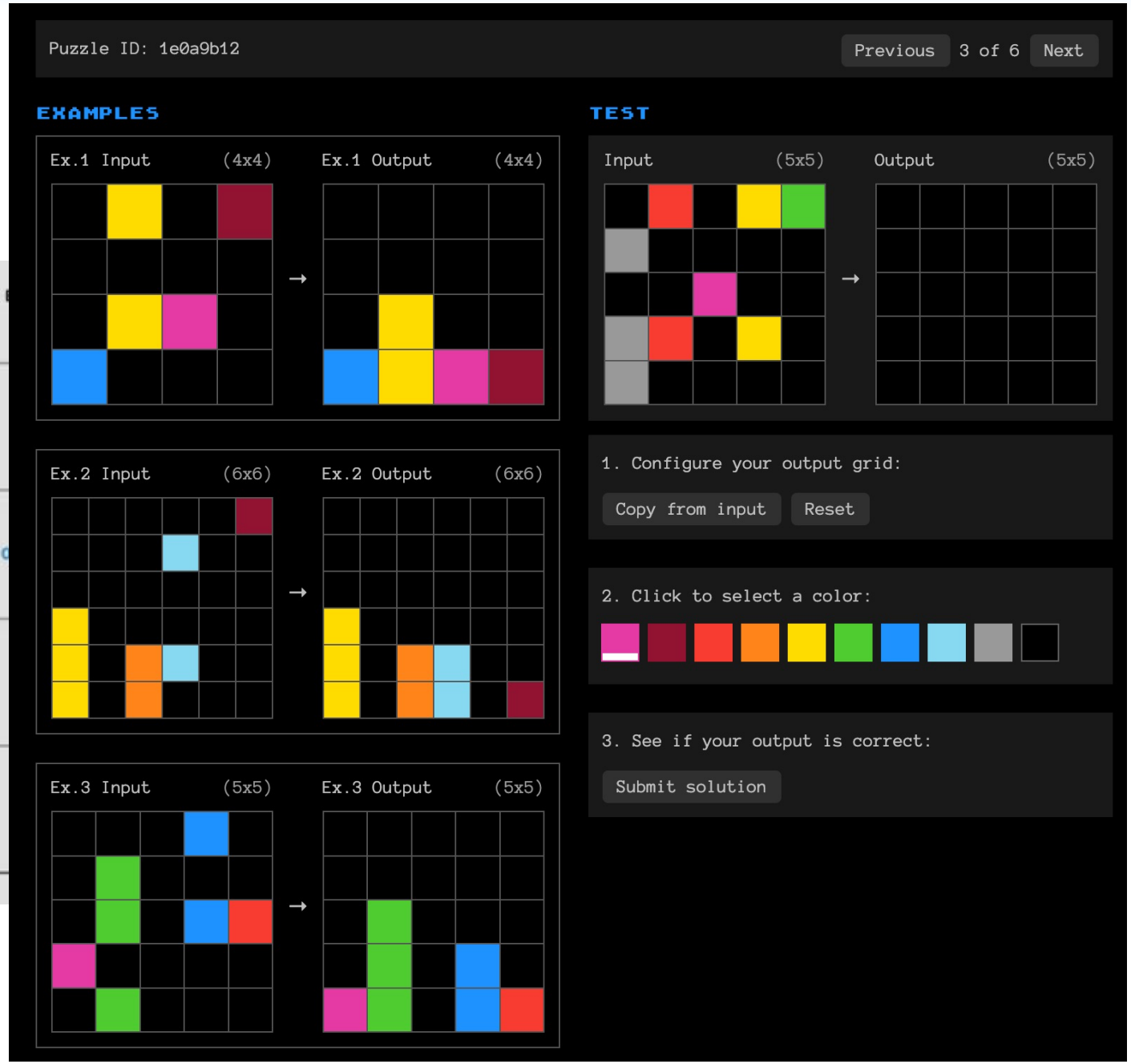
https://openai.com/index/learning-to-reason-with-llms/

Rating: users participated in recent 6 months

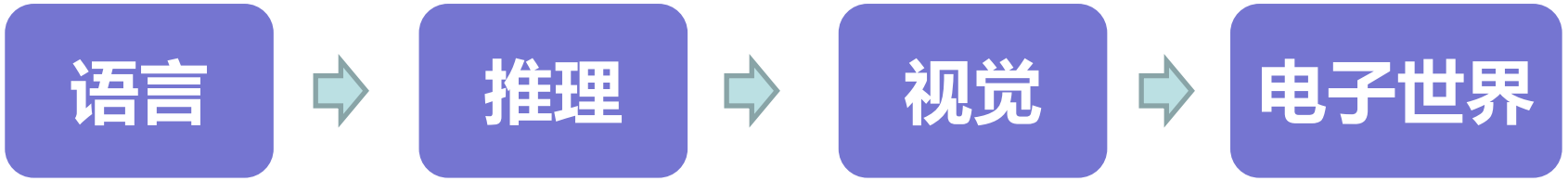
	Who	#	=
1	jiangly	173	3976
2	tourist	272	3815
3	jqdai0815	131	3682
4	ksun48	286	3614
5	orzdevinwang	69	3526
6	ecnerwala	192	3514
7	Benq	156	3482
8	hos.lyric	201	3382
9	gamegame	95	3374
10	heuristica	63	3357
11	Radewoosh	251	3355
12	Ormlis	140	3354
13	LJC00118	98	3330
14	maspy	86	3313
15	ugly2333	141	3310
16	turmax	97	3294
17	Kevin114514	85	3292
18	Um_nik	294	3285
19	maroonrk	187	3262
20	dorijanlendvaj	181	3236
20	arvindf232	52	3236
156	ymmparsa	81	2751
157	RDDCCD	51	2749
158	green_gold_dog	47	2748
159	Tlatoani	126	2747
160	Noam527	144	2746
160	KevinWan	123	2746
160	Wuyanru	37	2746
163	fengqiyuka	27	2745
164	CJ-zhuyifan	34	2744
165	E869120	36	2738
165	Xylenox	141	2738
165	Chinese_zjc_	49	2738
168	sevlll777	210	2736
168	kshitij_sodani	121	2736
170	Mangooste	65	2733
170	zhaohaikun	71	2733
172	DPprince	12	2731
173	lexiyvv	143	2729
174	Dominater069	70	2728
175	RanRankeainie	10	2727
176	LeoPro	115	2726
177	PersistentLife	56	2725
178	dreamoon_love_AA	375	2724
178	wwwwodddd	20	2724
180	physics0523	226	2722
180	umbrella-leaf	36	2722
182	BigBag	278	2720
182	Nutella3000-7	10	2720
184	Nyaan	143	2719
184	Zanite	59	2719
186	namelessgugugu	23	2717
187	teraqqq	29	2715
188	sansen	157	2713

https://codeforces.com/ratings


O FAMILY PERFORMANCE / ARC-AGI SEMI-PRIVATE
100%
75%
50%
25%
0%
75.7%
LOW
01 SERIES
31%
32%
25%
HIGH
MED
LOW
13%
PREVIEW
8%
MINI


Puzzle ID: 1e0a9b12
Previous
3 of 6
Next
EXAMPLES
Ex.1 Input (4x4)
Ex.1 Output (4x4)
Ex.2 Input (6x6)
Ex.2 Output (6x6)
Ex.3 Input (5x5)
Ex.3 Output (5x5)
TEST
Input (5x5)
Output (5x5)
1. Configure your output grid:
Copy from input
Reset
2. Click to select a color:
3. See if your output is correct:
Submit solution

Why Large Models?



- 对话

- 数学题
- 代码题
- 物化生题

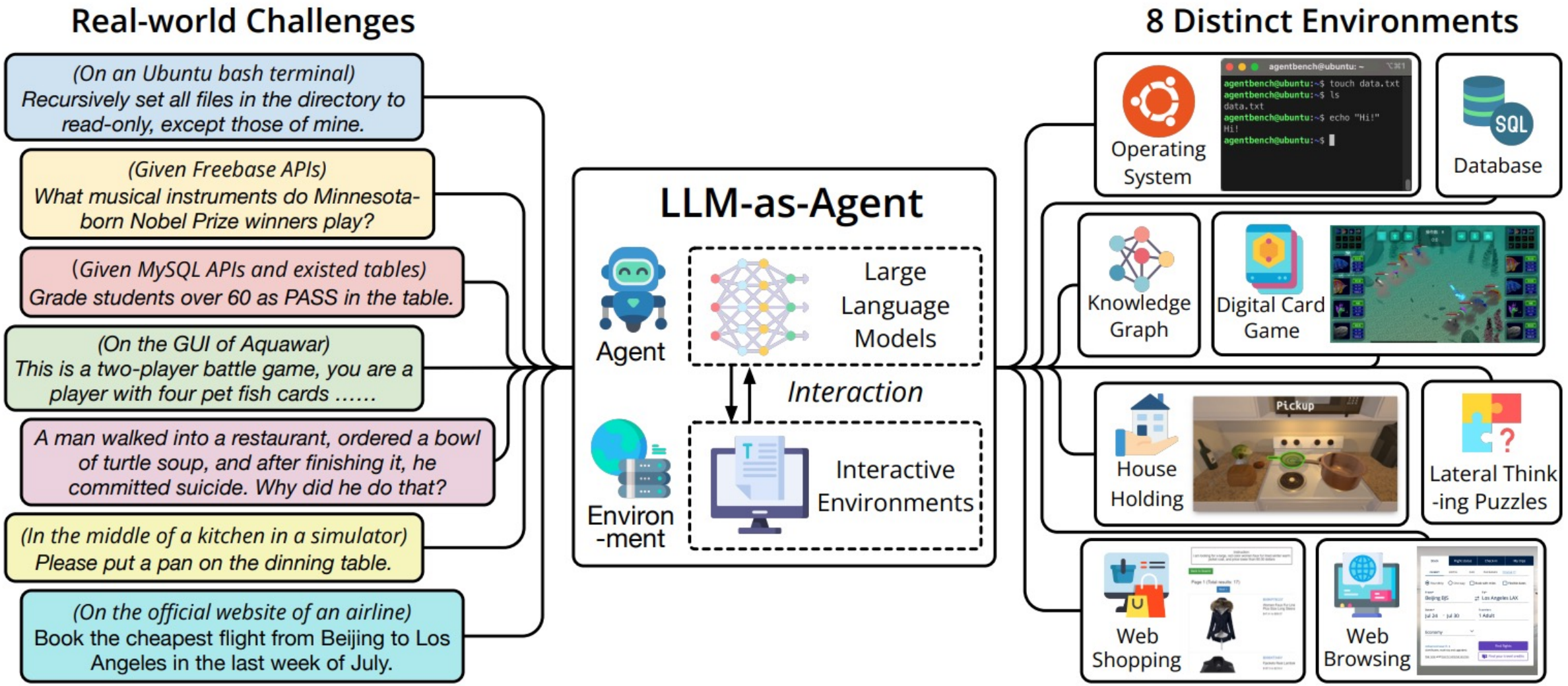
- 围棋

- 能看
- 能听
- 能说

- 手机智能体
- 网页智能体
- 电脑智能体

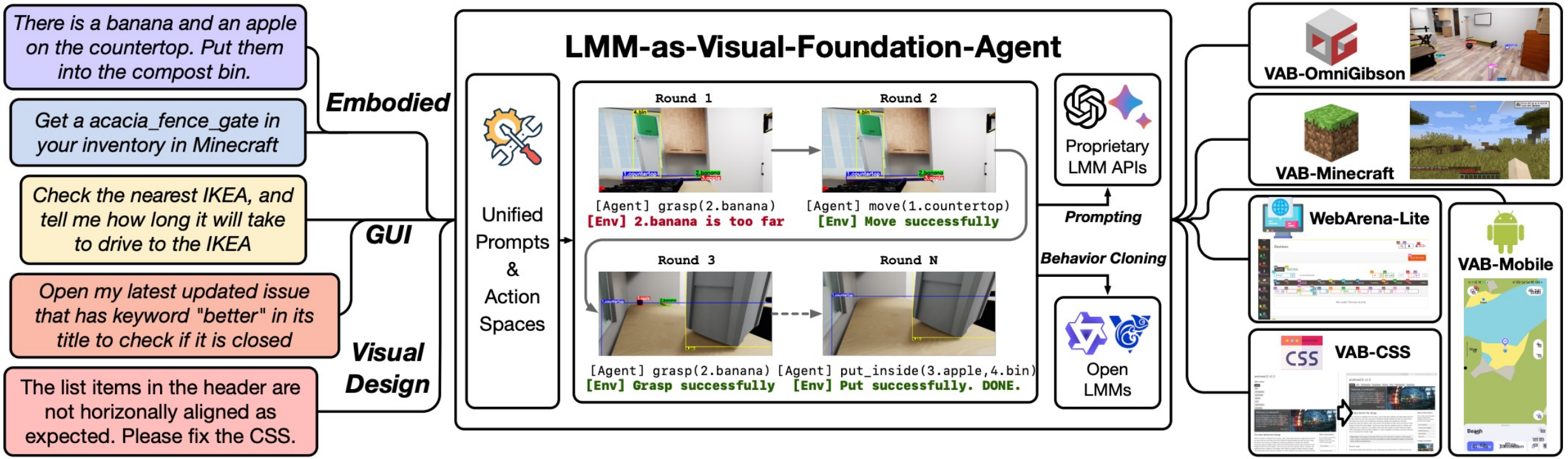
“用计算模型可描述的认知问题，AI很快超过人类水平”

大模型作为智能体 AgentBench



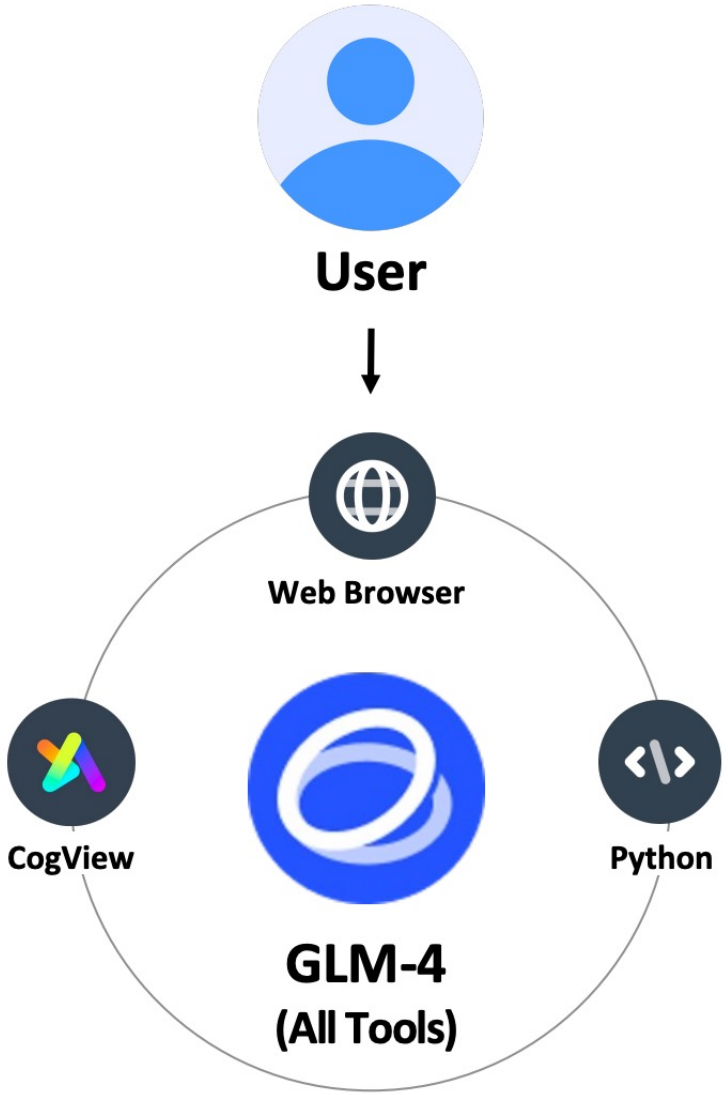
Liu et al, *AgentBench: Evaluating LLMs as Agents*. arXiv:2308.03688, ICLR'24

大模型作为智能体 VisualAgentBench



Liu et al, *VisualAgentBench: Towards Large Multimodal Models as Visual Foundation Agents*. arXiv:2408.06327, ICLR'25

智能体模型



Search for the global population from 2000 to 2023, then calculate the average annual growth rate.

1 Web Browser Search global population from 2000 to 2023

1 World Population 1950-2024
https://www.macrotrends.net/...

2 World Population Clock
https://www.worldometers.info/...

2 Click + Summarize

According to the information I found, from **2000** to **2023**, the global population grew from about **6.15 billion** to about **8.05 billion...**

3 Python Generate + Execute

```
# Starting and ending population values
starting_population = 6.15e9  # 2000 population
ending_population = 8.05e9    # 2023 population
# Number of years between 2000 and 2023
years = 2023 - 2000
# CAGR formula
cagr = (ending_population / starting_population) ** (1 / years) - 1
cagr * 100  # Convert to percentage
```

Result: 1.17739919480071

The average annual growth rate of the global population from 2000 to 2023 was approximately **1.18%.**

智能体模型

- 智能体模型实现**自主根据用户意图，自动理解、规划复杂指令，自由调用工具（**网页浏览器、Code Interpreter 代码解释器和多模态文生图大模型等），以完成复杂任务。
- 简单来讲，即只需一个指令，模型会自动分析指令，结合上下文选择决定调用合适的工具。

文生图

网页浏览

代码解释器

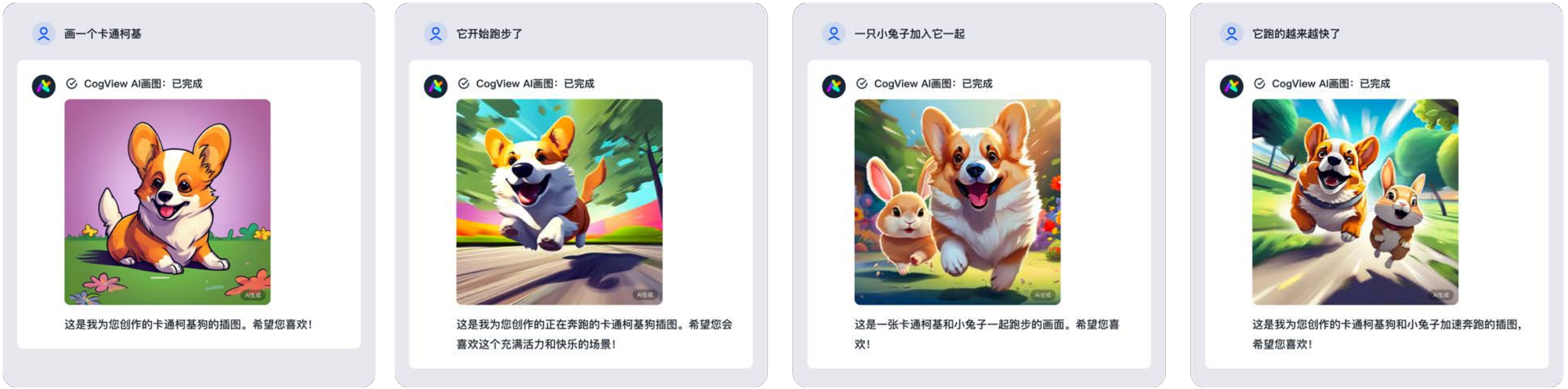
Function Calling

单工具调用

多工具自动调用

智能体模型

- **融合多种工具的 Agent 形态，无需手动指定工具，自动触发**
  - Browser, Code Interpreter, **Text2Image (CogView3)**



Zheng et al., **CogView3**: Finer and Faster Text-to-Image Generation via Relay Diffusion. arXiv:2403:05121. ECCV'24

智能体模型

- **融合多种工具的 Agent 形态，无需手动指定工具，自动触发**
  - Browser, **Code Interpreter**, Text2Image (CogView3)

x^3+ax^2−5x+9除以x+4，商为x^2+bx−1，余数为13，计算 a, b 的值。

要解决这个问题，我们可以使用多项式除法的原理。我们知道，当一个多项式 $P(x)$ 被另一个多项式 $D(x)$ 整除时，可以得到一个商 $Q(x)$ 和一个余数 $R(x)$。在这种情况下，我们有：

$$P(x) = D(x) \times Q(x) + R(x)$$

已知条件是 $P(x) = x^3 + ax^2 - 5x + 9$，$D(x) = x + 4$，$Q(x) = x^2 + bx - 1$，且余数 $R(x) = 13$。

根据多项式除法的原理，我们可以将 $P(x)$ 表达为 $D(x)$ 和 $Q(x)$ 的乘积加上余数 $R(x)$：

$$x^3 + ax^2 - 5x + 9 = (x + 4)(x^2 + bx - 1) + 13$$

我们可以通过展开右侧的乘积并将结果与左侧的多项式进行比较来找到 a 和 b 的值。接下来，我将进行计算。

代码生成：已完成

结果

{a: 3, b: -1}

根据计算结果，a 和 b 的值分别是 3 和 -1。因此，多项式 $x^3 + ax^2 - 5x + 9$ 除以 $x + 4$，其商为 $x^2 + bx - 1$，余数为 13 时，$a = 3$，$b = -1$。

代码生成：已完成

python 复制

```python
from sympy import symbols, Eq, solve

# 定义变量
x, a, b = symbols('x a b')

# 已知的多项式
P_x = x**3 + a*x**2 - 5*x + 9
D_x = x + 4
Q_x = x**2 + b*x - 1
R_x = 13

# 构造等式 P(x) = D(x) * Q(x) + R(x)
equation = Eq(P_x, D_x * Q_x + R_x)

# 展开等式右侧
expanded_equation = equation.expand()

# 解方程求 a 和 b 的值
solutions = solve(expanded_equation, (a, b))
solutions
```

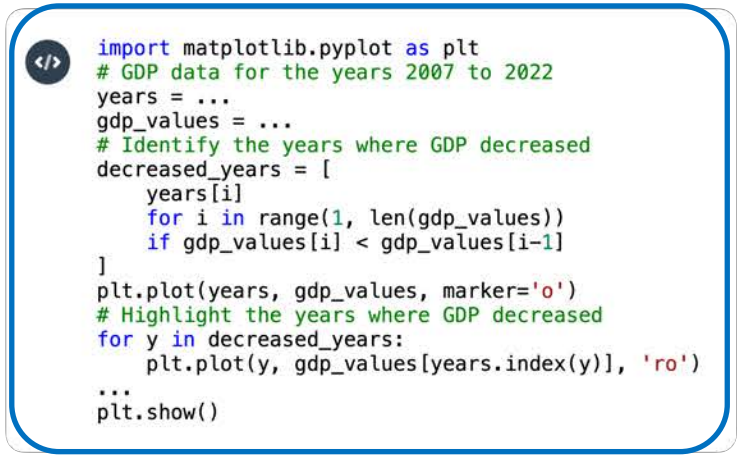
结果

{a: 3, b: -1}

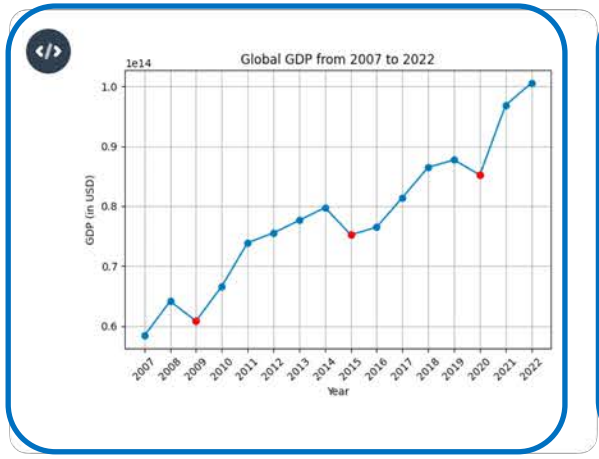
智能体模型

- **融合多种工具的 Agent 形态，无需手动指定工具，自动触发**
  - **Browser**, **Code Interpreter**, Text2Image (CogView3)

“请查一下全球过去十多年的GDP情况，并画出趋势图，如果哪年GDP下降，在趋势中用红色标出来。”



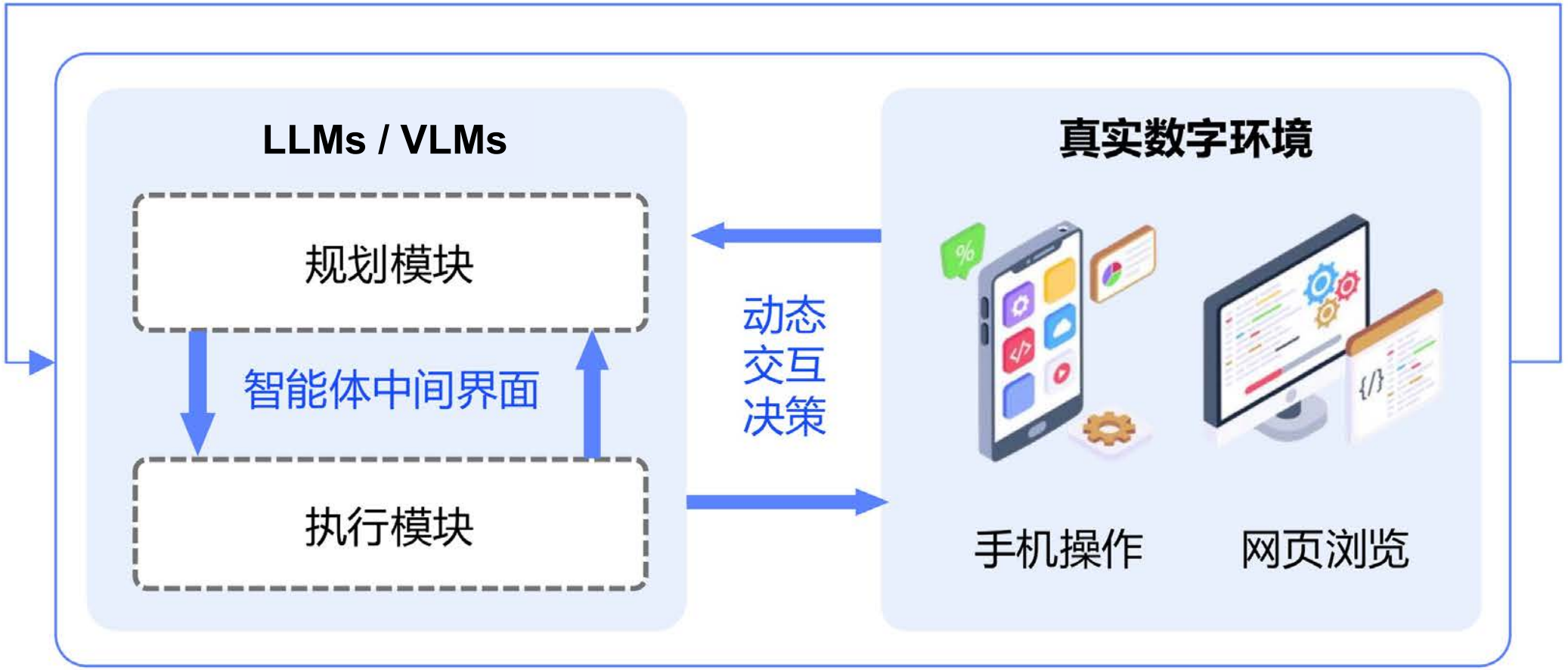
```python
import matplotlib.pyplot as plt
# GDP data for the years 2007 to 2022
years = ...
gdp_values = ...
# Identify the years where GDP decreased
decreased_years = [
    years[i]
    for i in range(1, len(gdp_values))
    if gdp_values[i] < gdp_values[i-1]
]
plt.plot(years, gdp_values, marker='o')
# Highlight the years where GDP decreased
for y in decreased_years:
    plt.plot(y, gdp_values[years.index(y)], 'ro')
...
plt.show()
```



根据从MacroTrends获取的数据，我绘制了2007年至2022年全球GDP的折线图。在图表中，我已经将下降的数据点标记为红色，以便于识别。您可以查看这张图表，以直观地看到全球GDP的变化情况。

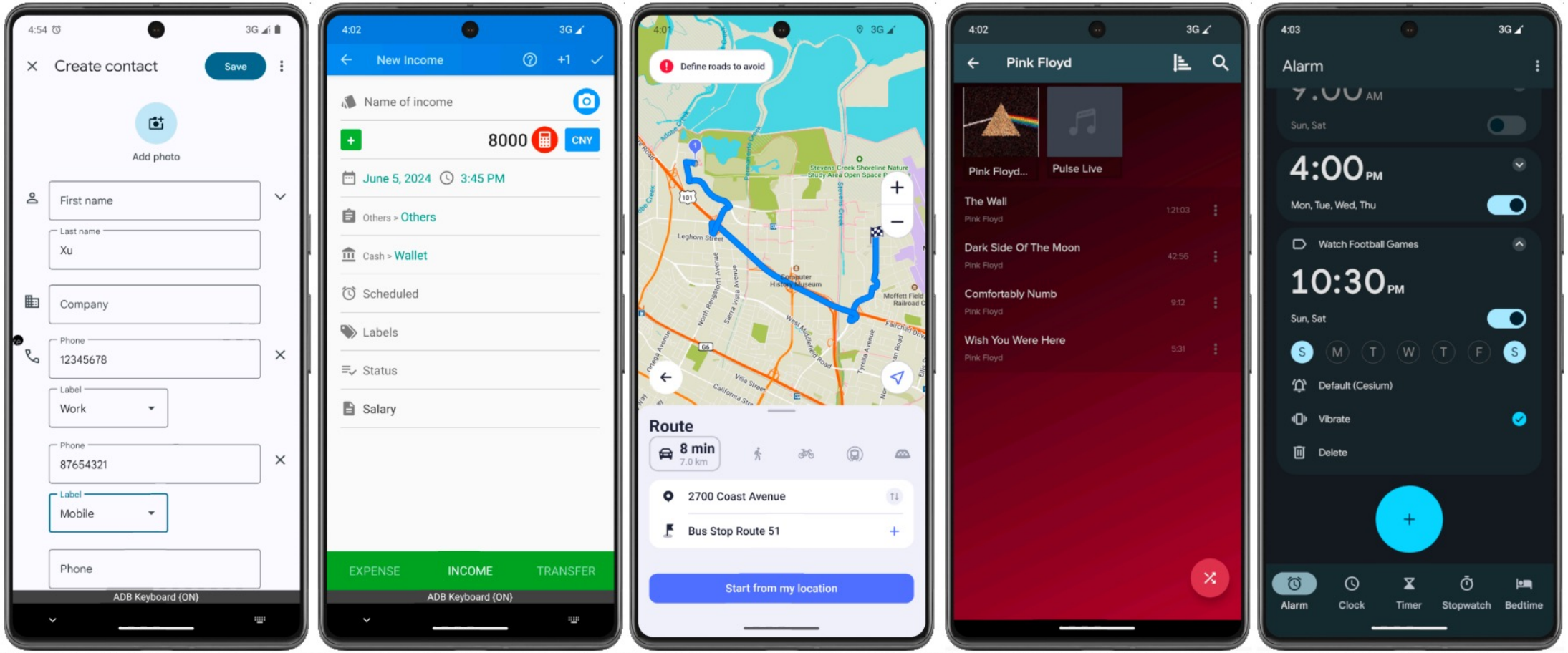
LLM/VLM for Building GUI agents?

WebRL：自进化在线课程强化学习框架



Zehan Qi et al., WebRL: Training LLM Web Agents via Self-Evolving Online Curriculum Reinforcement Learning. arXiv:2411 ICLR'25

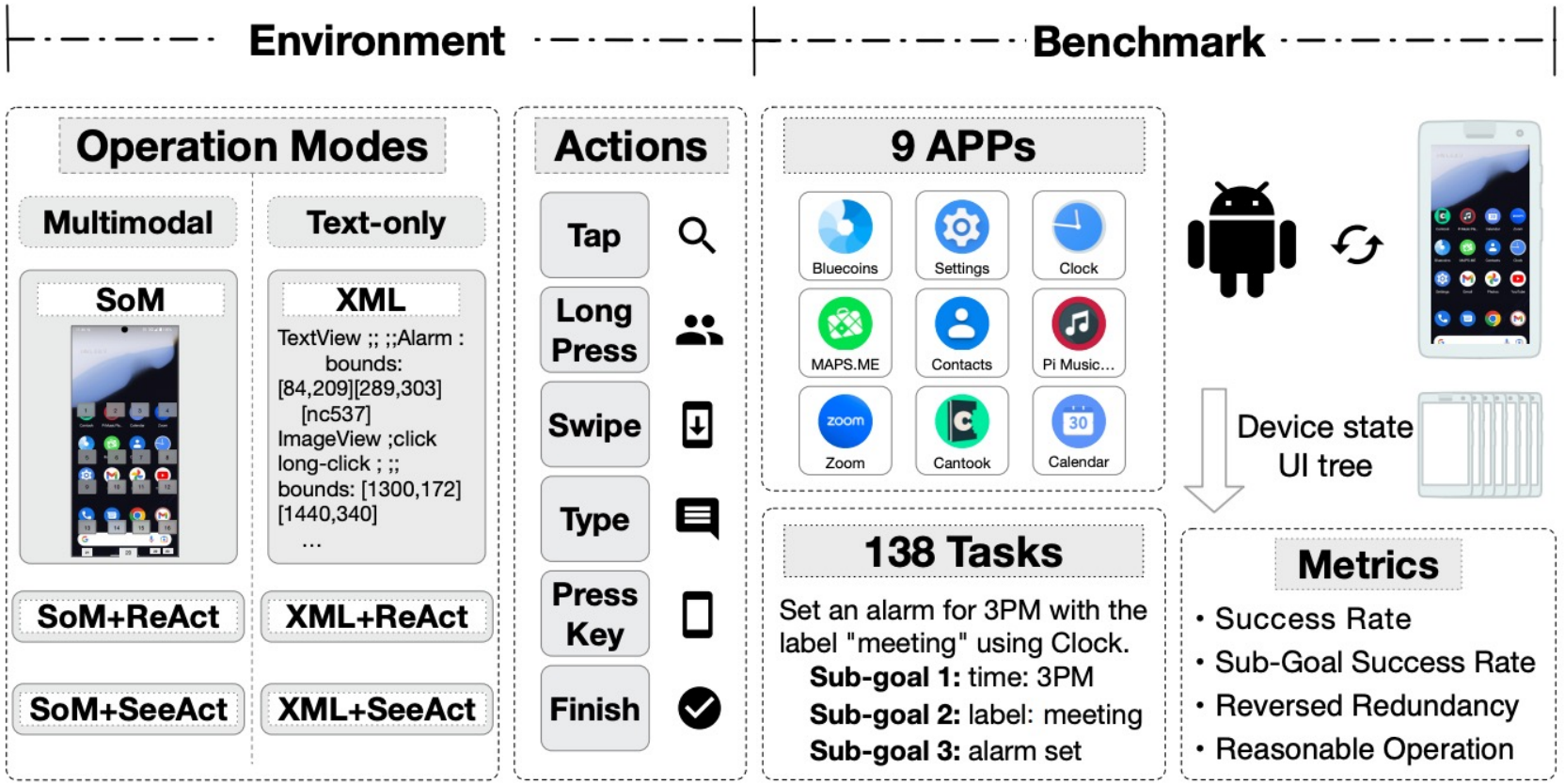
AndroidLab：交互与测试环境



Task: Add a contacts whose name is Xu, set the working phone…	Task: Record an income of 8000 CNY in the books, and mark it as…	Task: Check the driving distance and time between Bus stop of…	Task: Sort Pink Floyd's songs by duration time in descending order.	Task: I need set an 10:30PM clock every weekend, and label it…
Sub-Goals: · Name: Xu · Working phone number · Mobile phone number	Sub-Goals: · Enter: New income · Cash: 8000 · Note: …	Sub-Goals: · Driving distance · Driving time	Sub-Goals: · Page: ARTISTS · Artist: Pink Floyd · Order: Descending	Sub-Goals: · Time: 10:30PM · Frequency: Weekend · Label: …

AndroidLab：交互与测试环境

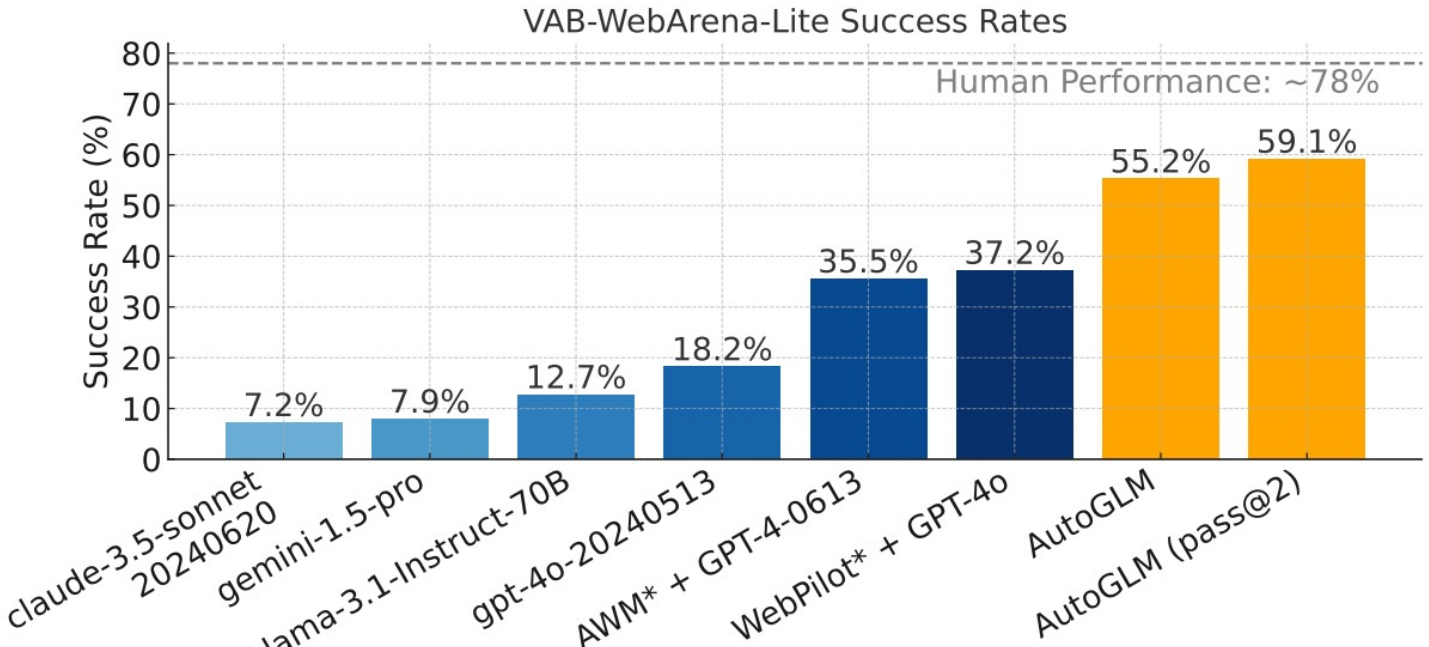
- 通过利用XML信息和set-of-mark方法，构造了在LLMs和VLMs完全等价的操作空间，保证公平比较
- 适配ReAct和SeeAct两种推理后输出的框架

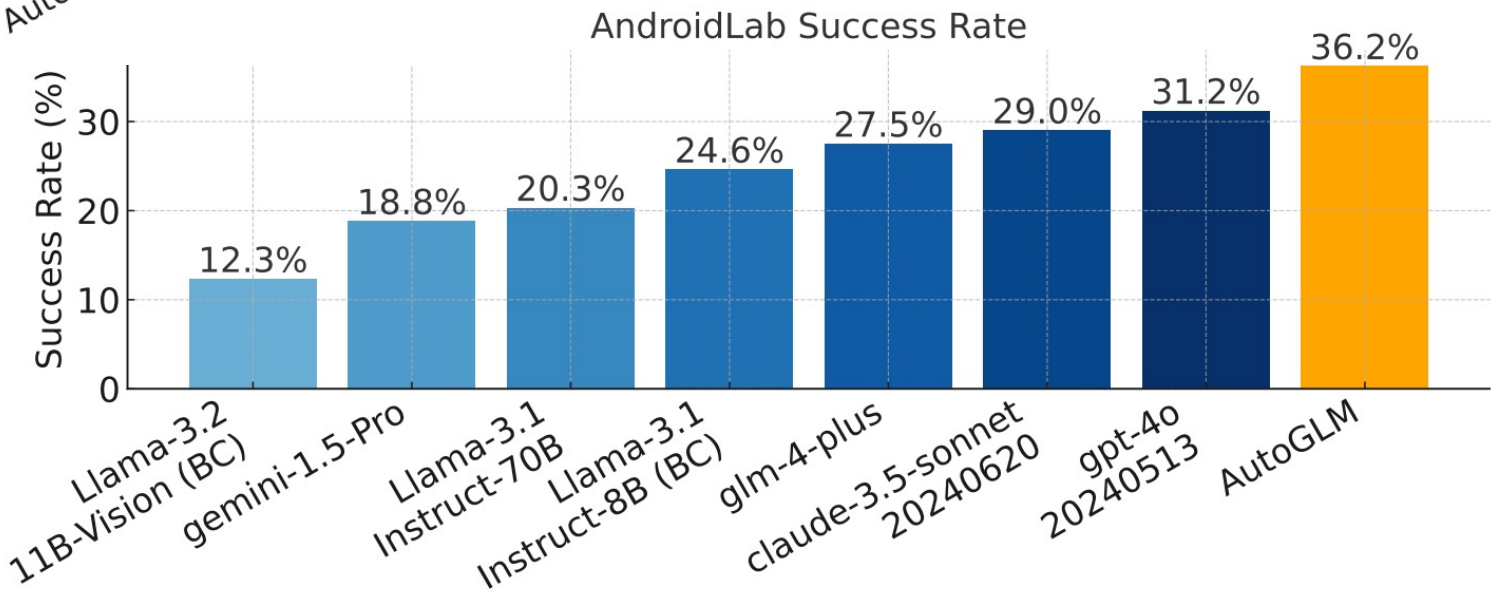


AutoGLM with AndroidLab & WebRL

- Yifan Xu et al., **AndroidLab**: Training and Systematic Benchmarking of Android Autonomous Agents. arXiv:**2410**.24024
- Zehan Qi et al., **WebRL**: Training LLM Web Agents via Self-Evolving Online Curriculum Reinforcement Learning. arXiv:**2411**.02337
- Xiao Liu et al., **AutoGLM**: Autonomous Foundation Agents for GUIs, arXiv:**2411**.00820

AutoGLM with AndroidLab & WebRL





- Xiao Liu et al., AutoGLM: Autonomous Foundation Agents for GUIs, arXiv:**2411**.00820

Large AI Models

- 对话

- 数学题
- 代码题
- 物化生题

- 围棋

- 能看
- 能听
- 能说

- 手机智能体
- 网页智能体
- 电脑智能体

- 具身智能
- VLA

“用计算模型可描述的认知问题，AI很快超过人类水平”

John Jumper

"for protein structure prediction"

© Nobel Prize Outreach. Photo: Clément Morin

Jason Wei

- Chain of Thought
- Emergent Abilities of LLMs
- OpenAI o1

Photo: Jason HP

https://github.com/THUDM

ChatGLM-6B	ChatGLM-6B: An Open Bilingual Dialogue Language Model | 开源双语对话语言模型	对话模型	Python	5.2k	41k
ChatGLM2-6B	ChatGLM2-6B: An Open Bilingual Chat LLM | 开源双语对话语言模型	对话模型	Python	1.9k	16k
ChatGLM3	ChatGLM3 series: Open Bilingual Chat LLMs | 开源双语对话语言模型	对话模型	Python	1.6k	14k
CogVideo	text and image to video generation: CogVideoX (2024) and CogVideo (ICLR 2023)	视频生成	Python	1k	11k
CodeGeeX	CodeGeeX: An Open Multilingual Code Generation Model (KDD 2023)	代码模型	Python	628	8.4k
GLM-130B	GLM-130B: An Open Bilingual Pre-Trained Model (ICLR 2023)	千亿基座	Python	604	7.7k
CodeGeeX2	CodeGeeX2: A More Powerful Multilingual Code Generation Model	代码模型	Python	531	7.6k
CogVLM	a state-of-the-art-level open visual language model | 多模态预训练模型	图片理解	Python	427	6.4k
GLM-4	GLM-4 series: Open Multilingual Multimodal Chat LMs | 开源多语言多模态对话模型	对话模型	Python	513	6k
VisualGLM-6B	Chinese and English multimodal conversational language model | 多模态中英双语对话语言模型	图片理解	Python	422	4.1k
GLM	GLM (General Language Model)	对话模型	Python	326	3.2k
GLM-4-Voice	GLM-4-Voice | 端到端中英语音对话模型	语音模型	Python	223	2.8k
AgentBench	A Comprehensive Benchmark to Evaluate LLMs as Agents (ICLR'24)	智能体	Python	183	2.4k
CogVLM2	GPT4V-level open-source multi-modal model based on Llama3-8B	图片、视频理解	Python	152	2.3k

References

1. Team GLM et al. **ChatGLM**: A Family of Large Language Models from GLM-130B to GLM-4 All Tools. arXiv:2406.12793
2. Z Du et al. Understanding **Emergent Abilities** of Language Models from the Loss Perspective. arXiv:2403.15796, NeurIPS'24
3. D Zhang et al. **ReST-MCTS**: LLM Self-Training via Process Reward Guided Tree Search. NeurIPS'24
4. Yang et al., **CogVideoX**: Text-to-Video Diffusion Models with An Expert Transformer. arXiv:2408.06072
5. Hanyu Lai, Xiao Liu, et al. **AutoWebGLM**: A Large Language Model-based Web Navigating Agent. KDD'24
6. Wenyi Hong, et al. **CogAgent**: A Visual Language Model for GUI Agents. CVPR'24.
7. Zeng et al. **AgentTuning**: Enabling Generalized Agent Abilities for LLMs. arXiv: 2310.12823. Findings of ACL'24
8. Liu e al. **AgentBench**: Evaluating LLMs as Agents. arXiv: 2308.03688. ICLR 2024.
9. Liu et al., **VisualAgentBench**: Towards Large Multimodal Models as Visual Foundation Agents. arXiv: 2408.06327
10. Yushi Bai et al. **LongBench**: A Bilingual, Multitask Benchmark for Long Context Understanding. ACL 24
11. Bai et al., **LongWriter**: Unleashing 10,000+ Word Generation from Long Context LLMs. arXiv:2408.07055
12. Liu et al, **APAR**: LLMs Can Do Auto-Parallel Auto-Regressive Decoding.arXiv:2401.06761, 2024.
13. Xu et al., **ImageReward**: Learning and Evaluating Human Preference for Text-to-Image Generation. NeurIPS'23.
14. Zheng et al., **CogView3**: Finer and Faster Text-to-Image Generation via Relay Diffusion. ECCV'24
15. Yang et al., **Inf-DiT**: Upsampling Any-Resolution Image with Memory-Efficient Diffusion Transformer. ECCV'24
16. Liu et al, Extensive **Self-Contrast** Enables Feedback-Free Language Model Alignment. arXiv:2404.00604
17. Wang et al. **CogVLM**: Visual Expert for Pretrained Language Models. arXiv: 2311.03079
18. Zheng, et al. **CodeGeeX**: A Pre-Trained Model for Code Generation with Multilingual Benchmarking on HumanEval-X. KDD'23.
19. Liu, et al. **WebGLM**: Towards An Efficient Web-enhanced Question Answering System with Human Preference. KDD'23.
20. Zeng, Liu, et al. **GLM-130B**: An Open Bilingual Pre-trained Model. ICLR'23.

致谢

计算机系知识工程实验室（KEG）

ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools

Team GLM: Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, Zihan Wang

https://arxiv.org/abs/2406.12793

算力或数据标注支持：

- 智谱AI
- 中科曙光
- 鹏城实验室
- 神威超算
- 济南超算中心

谢谢大家！

https://keg.cs.tsinghua.edu.cn/yuxiao/
yuxiaod@tsinghua.edu.cn