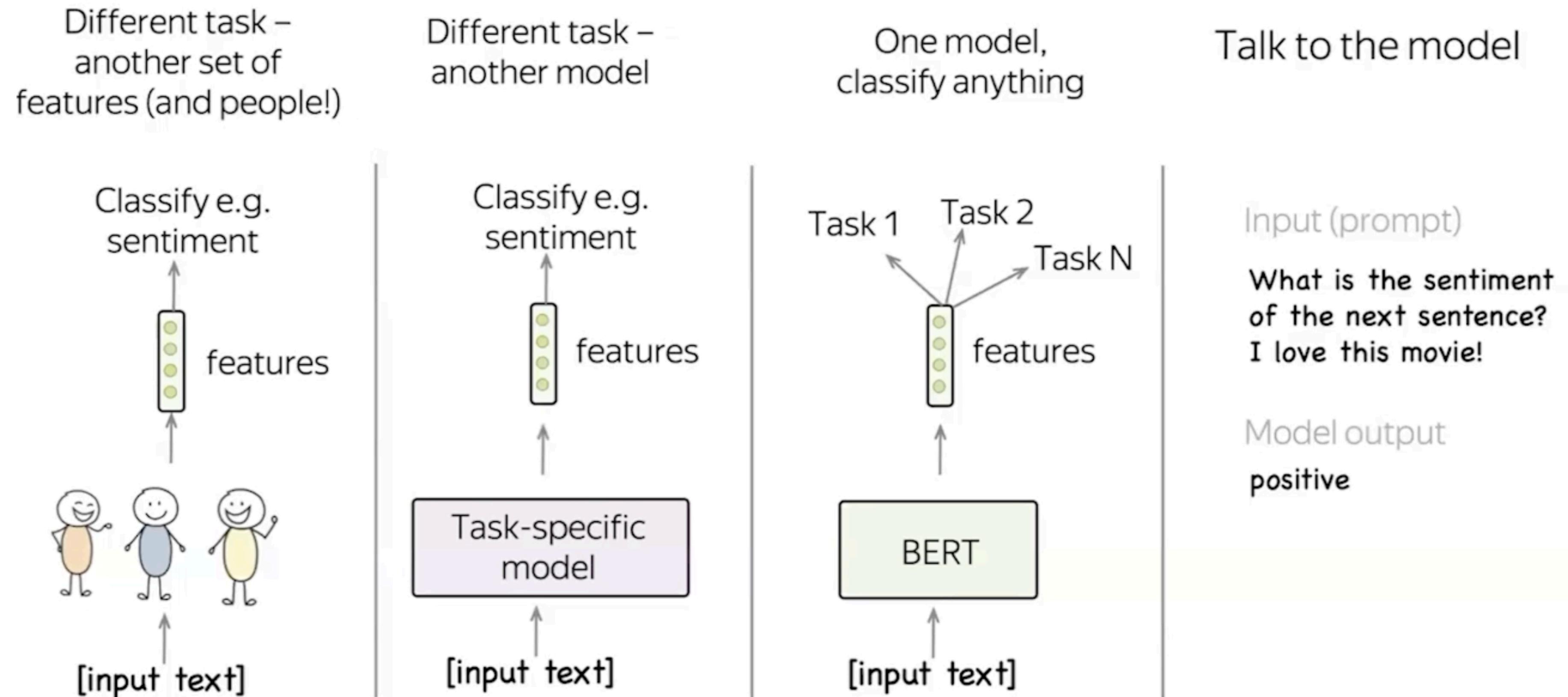


# **LLMs**

“Prompt engineering guide”

# Rough plan for the lecture

- LMs
  - basic idea, objective, results
- Large models, scaling laws
- In-context learning (1-2 slides)
  - Control sampling
  - Zero-shot, Few-shot, Chain-of-Thought
  - Instability of in-context learning (techniques to fix it)
- RLHF
  - Instruction tuning (RLHF, Supervised) orcas, alpacas and stuff
  - Examples from Jeremy with recall to previous stages
- Agents
  - Tool use: Retrieval, Interpreter/Functions (Examples from Jeremy), ReAct
  - Example usage (write basic code here during the lecture)
- Current state of the world
  - Software, services, how to stay (why?) on top of things
  - Benchmarks, Model capabilities, limitations

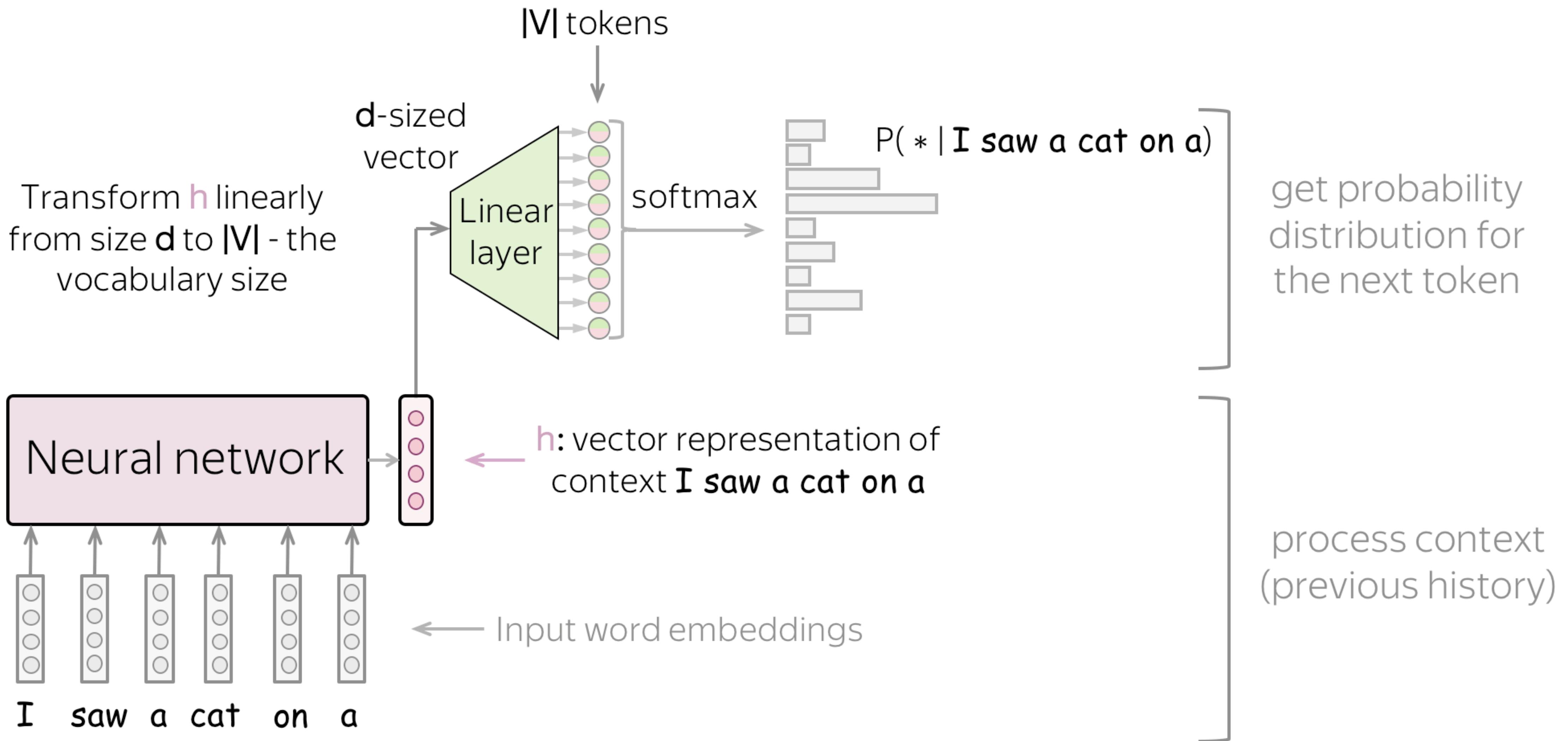


From YSDA NLP Course

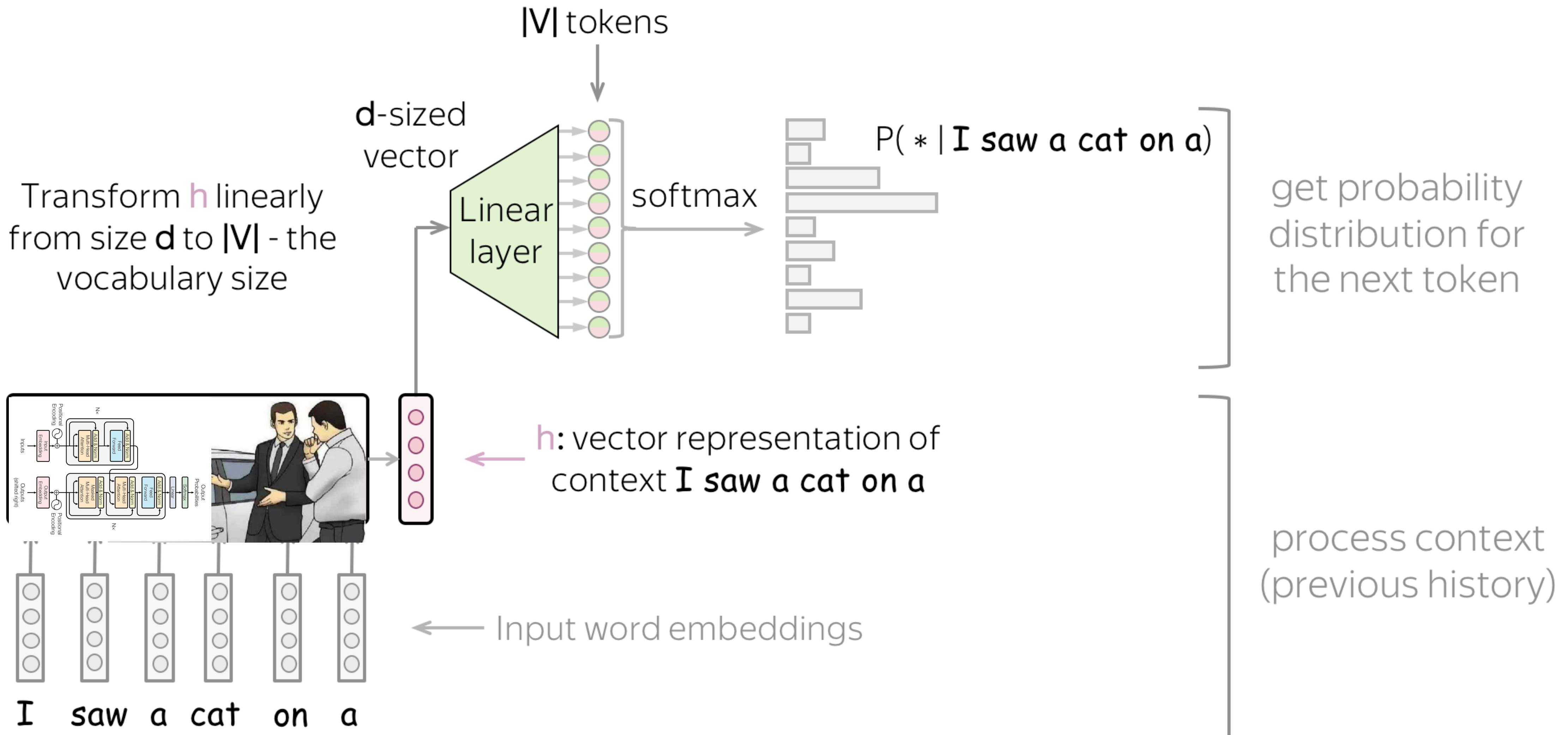
# LMs

I \_\_\_\_\_

# LMs



# LMs



# LMs

$$L(y_{1:M}) = L(y_1, y_2, \dots, y_M) = \sum_{t=1}^M \log_2 p(y_t | y_{<t})$$

Log-likelihood of the text

$$Loss(y_{1:M}) = - \sum_{t=1}^M \log p(y_t | y_{<t})$$

Note: cross-entropy (our loss)  
is negative log-likelihood

$$Perplexity(y_{1:M}) = 2^{-\frac{1}{M}L(y_{1:M})}.$$

- **the best** perplexity is 1

If our model is perfect and assigns probability 1 to correct tokens (the ones from the text), then the log-probability is zero, and the perplexity is 1.

- **the worst** perplexity is  $|V|$

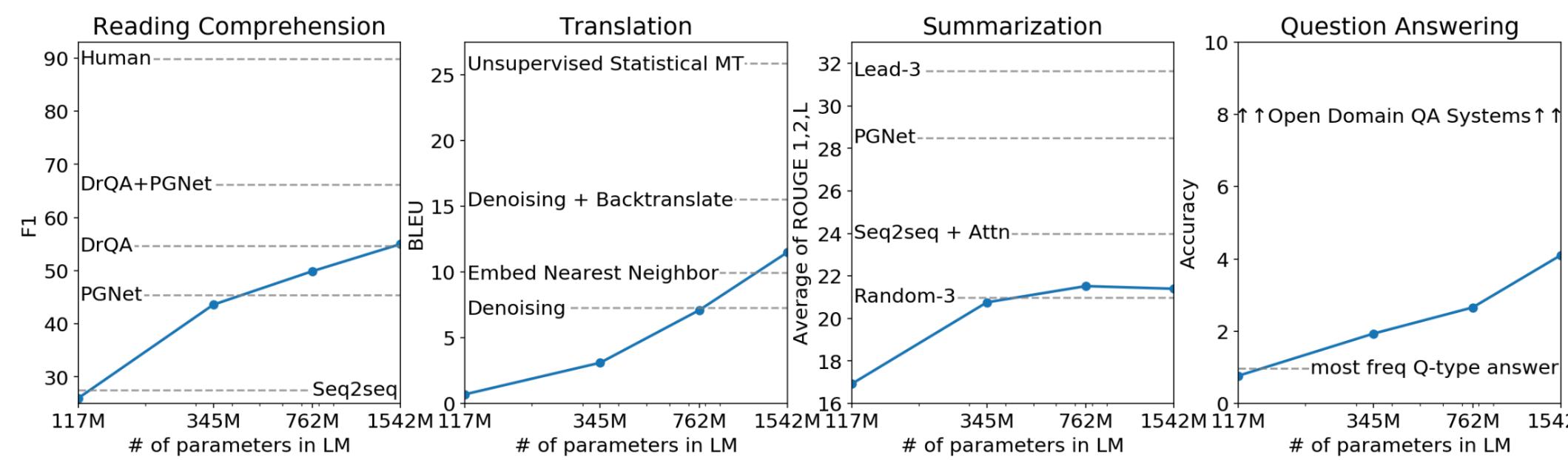
In the worst case, LM knows absolutely nothing about the data: it thinks that all tokens have the same probability  $\frac{1}{|V|}$  regardless of context. Then

$$Perplexity(y_{1:M}) = 2^{-\frac{1}{M}L(y_{1:M})} = 2^{-\frac{1}{M} \sum_{t=1}^M \log_2 p(y_t | y_{1:t-1})} = 2^{-\frac{1}{M} \cdot M \cdot \log_2 \frac{1}{|V|}} = 2^{\log_2 |V|} = |V|.$$

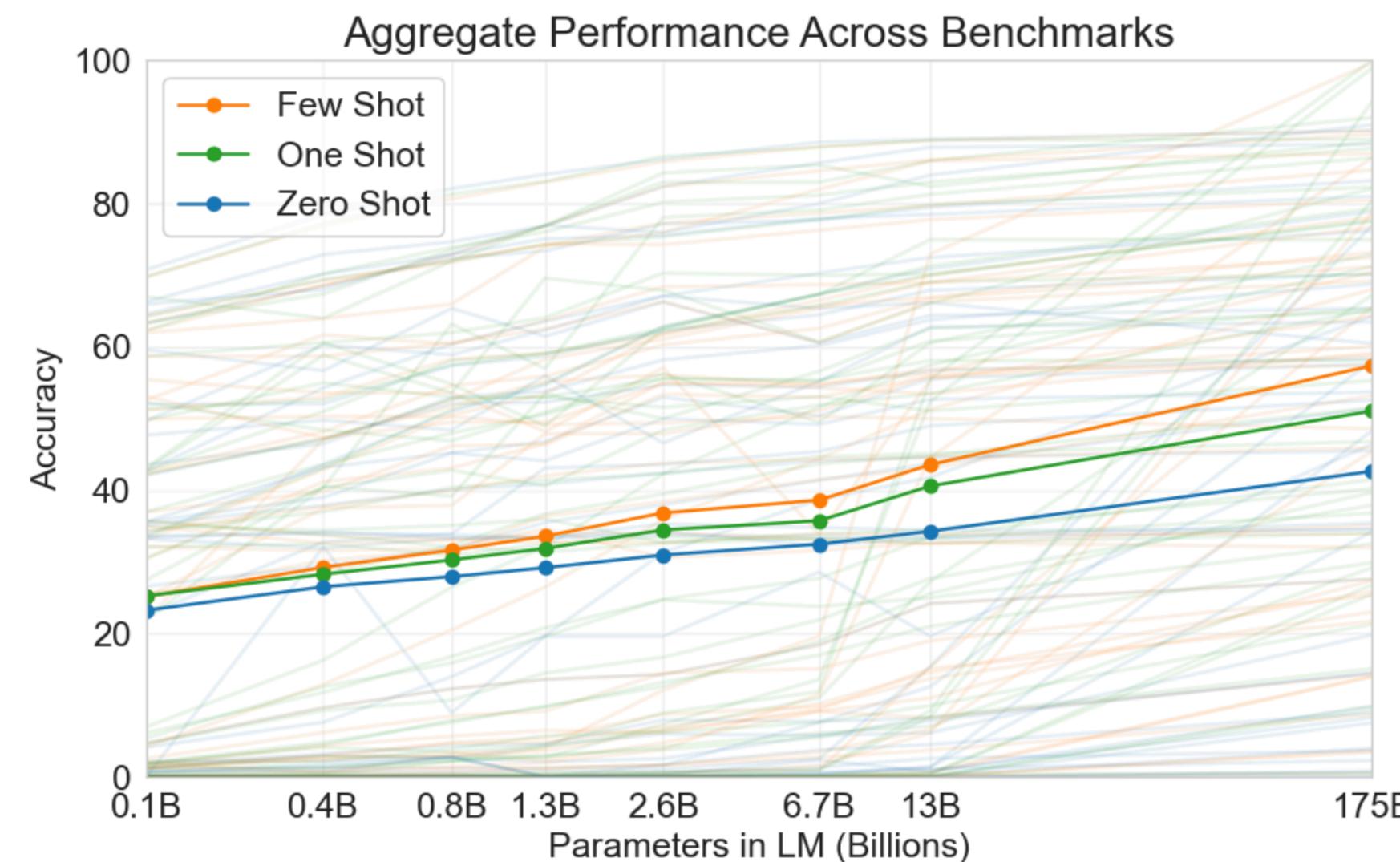
Therefore, your perplexity will always be between 1 and  $|V|$ .

# Large LMs

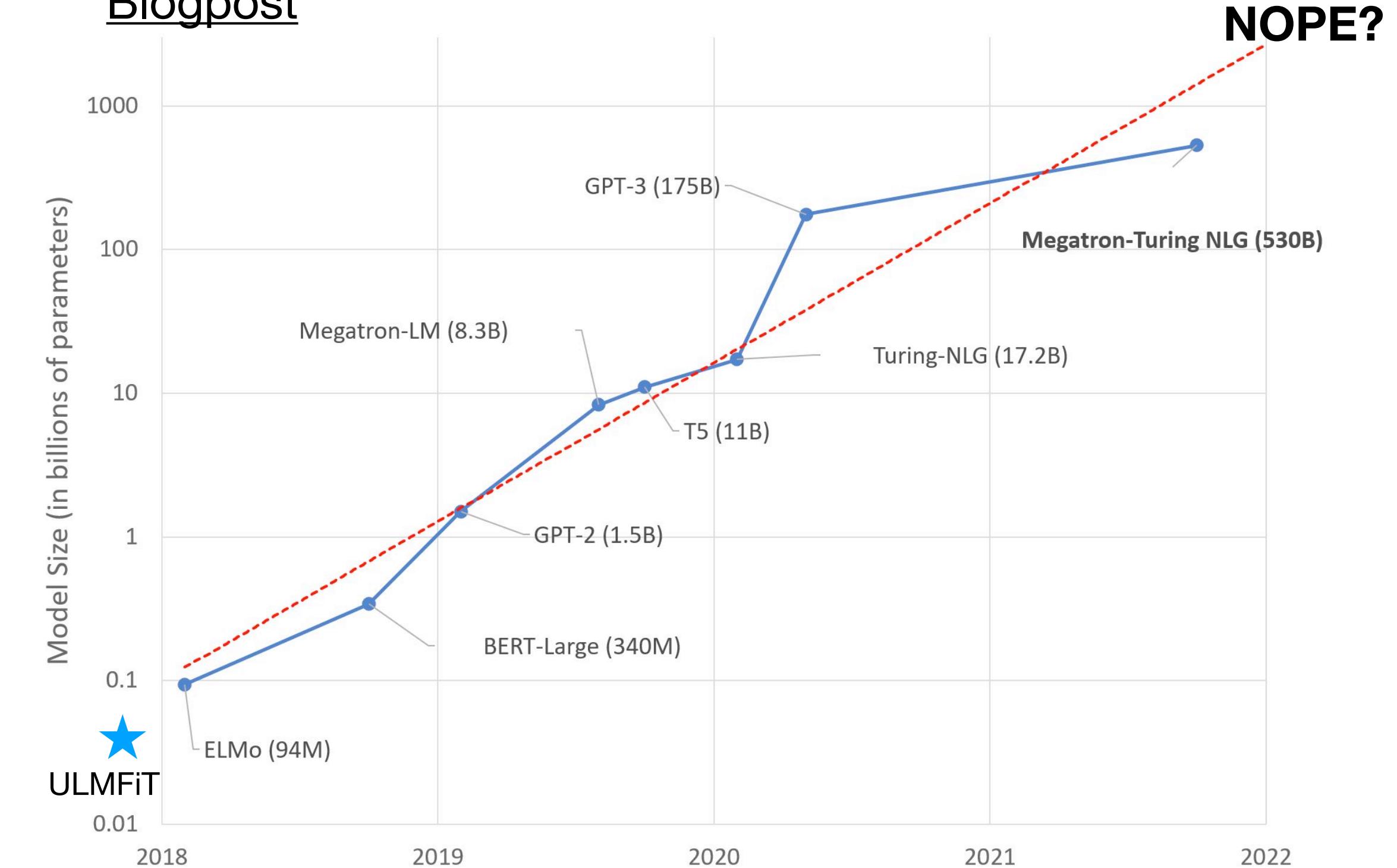
GPT-2 Started the “Emergent abilities” and “Scaling” trends



GPT-3. Meta-Learning, close to SoTA (at the time)  
on many NLP benchmarks and demonstrated **Emergent Abilities**



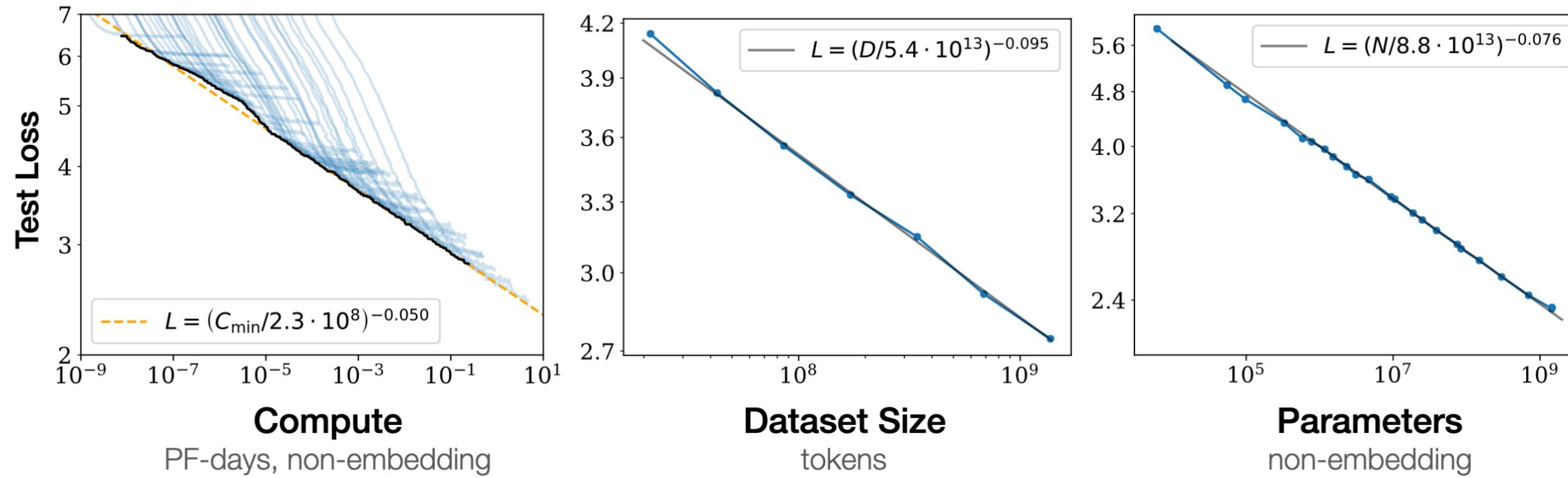
From “Large Language Models: A New Moore’s Law?”  
Blogpost



# Scaling

## How to choose the hyper parameters for large models?

<https://arxiv.org/abs/2001.08361> (Kaplan et al.)

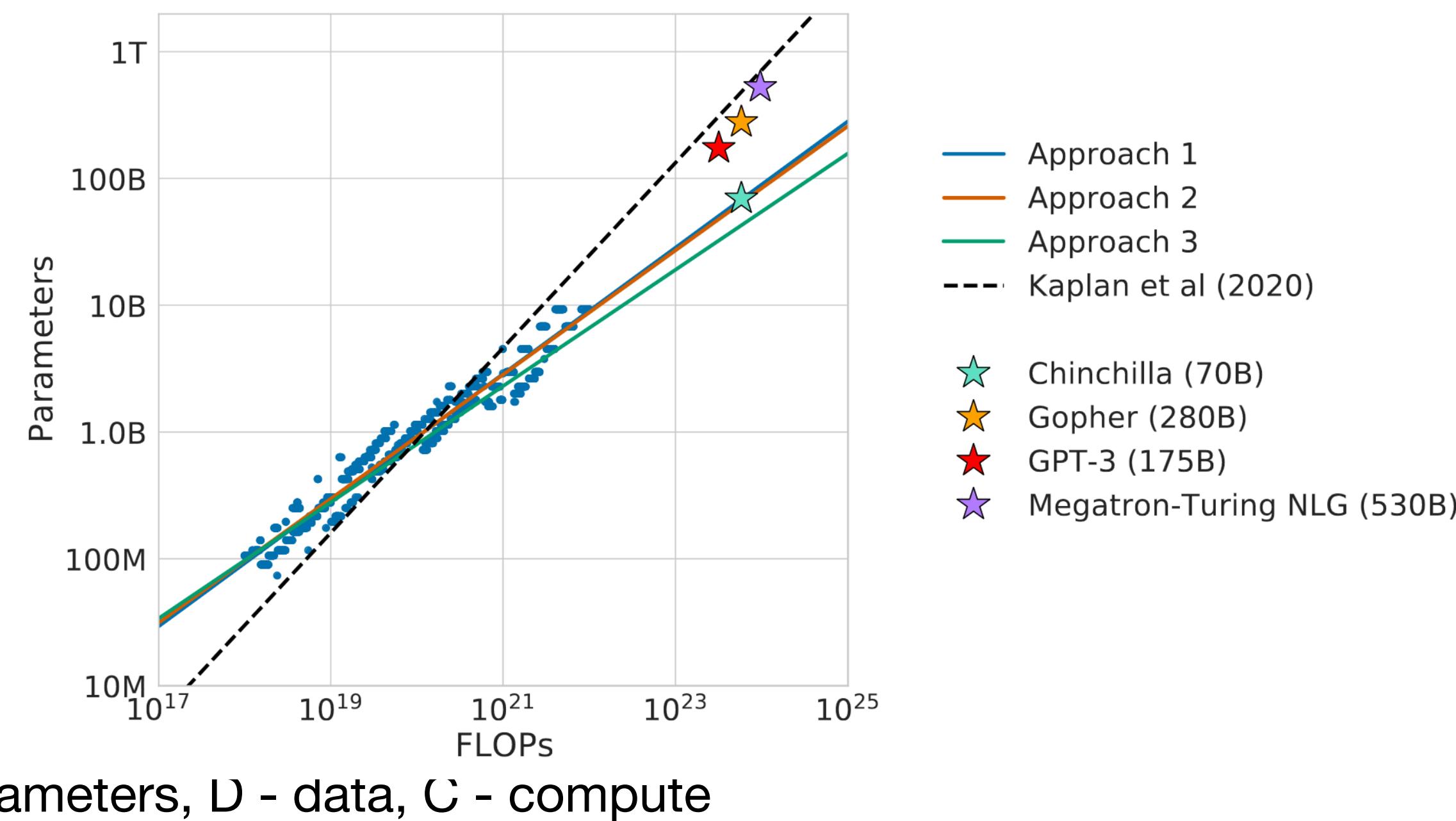


N - parameters, D - data, C - compute

- **Performance depends strongly on scale**, weakly on model shape
- **Performance has a power-law relationship with each of the three scale factors **N, D, C**** when not bottlenecked by the other two
- Every time we **increase the model size 8x**, we only need to **increase the data by 5x** to avoid a penalty

# Scaling

## How to choose the hyper parameters for large models?

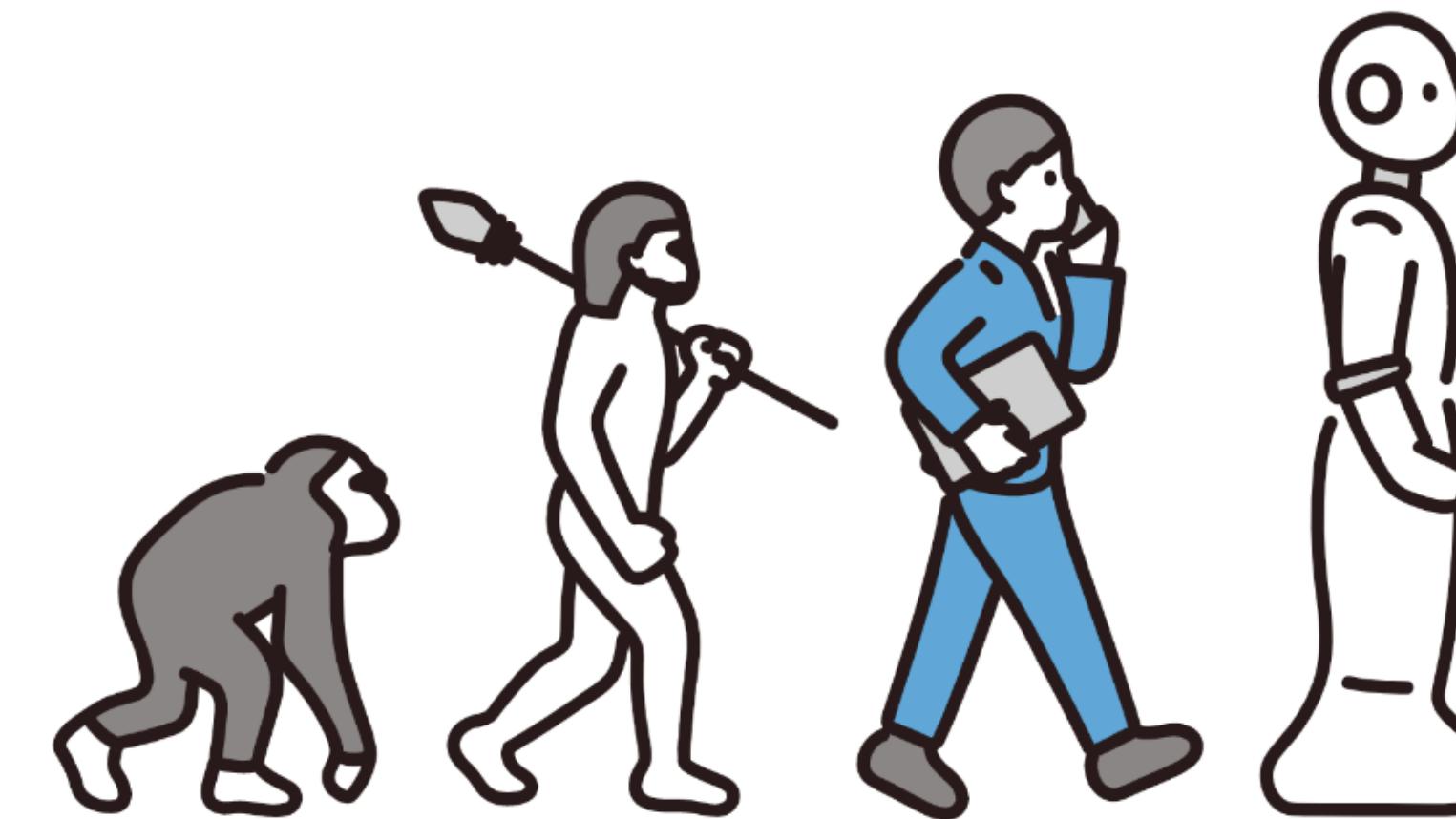


Model	Size (# Parameters)	Training Tokens
LaMDA (Thoppilan et al., 2022)	137 Billion	168 Billion
GPT-3 (Brown et al., 2020)	175 Billion	300 Billion
Jurassic (Lieber et al., 2021)	178 Billion	300 Billion
Gopher (Rae et al., 2021)	280 Billion	300 Billion
MT-NLG 530B (Smith et al., 2022)	530 Billion	270 Billion
<i>Chinchilla</i>	70 Billion	1.4 Trillion
<hr/>		
Random	25.0%	
Average human rater	34.5%	
GPT-3 5-shot	43.9%	
Gopher 5-shot	60.0%	
<b>Chinchilla 5-shot</b>	<b>67.6%</b>	
Average human expert performance	89.8%	
<hr/>		
June 2022 Forecast	57.1%	
June 2023 Forecast	63.4%	

- Given a **10x increase C** computational budget, **they suggests** that the size of the model **N** should **increase 5.5x** while the number of training **D** tokens **should only increase 1.8x**
- [Hoffman et al.](#) find that model size and the number **N** of training tokens **D** should be **scaled in equal proportions**.
- Authors introduce new compute optimal SoTA chinchilla 70B model

# In-context learning

Meta learning from text examples



*Let's think step by step.*

\* The image of human evolution is cited from <https://soco-st.com/13472>

# In-context learning

## Meta learning from text examples

Zero-Shot:

Fill in blank:

She held the torch in front of her.

She caught her breath.

"Chris? There's a step."

"What?"

"A step. Cut in the rock. About fifty feet ahead." She moved faster.

They both moved faster. "In fact," she said, raising the torch higher,

"there's more than a \_\_\_\_\_. **step**

Few Shot (more <https://gwern.net/gpt-3>)

Q. Explain the pun in the following joke: "Why do so many bakeries have cats? Because cats are always kneading."

A. The joke implies that bakeries use cats to knead their dough, because cats regularly ...

Q. Explain the pun in the following joke: "Why can Paypal freeze accounts without recourse or explanation? ...

A. The pun is on the phrase "Paypal infallibility", which alludes to the Catholic doctrine of "papal infallibility", that the pope cannot be wrong in religious dogma and papal judgments cannot be appealed or questioned. ...

Q. Explain the pun in the following joke: "What does Adult Swim call their physical retail stores? Brick and Morty."

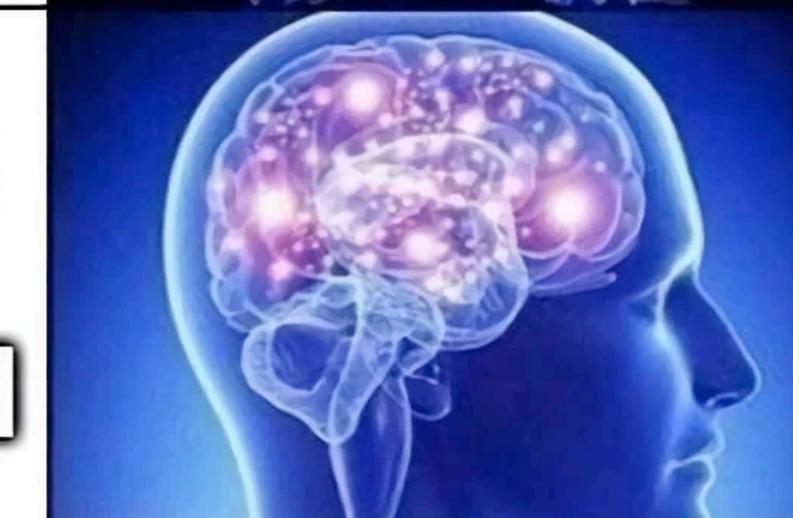
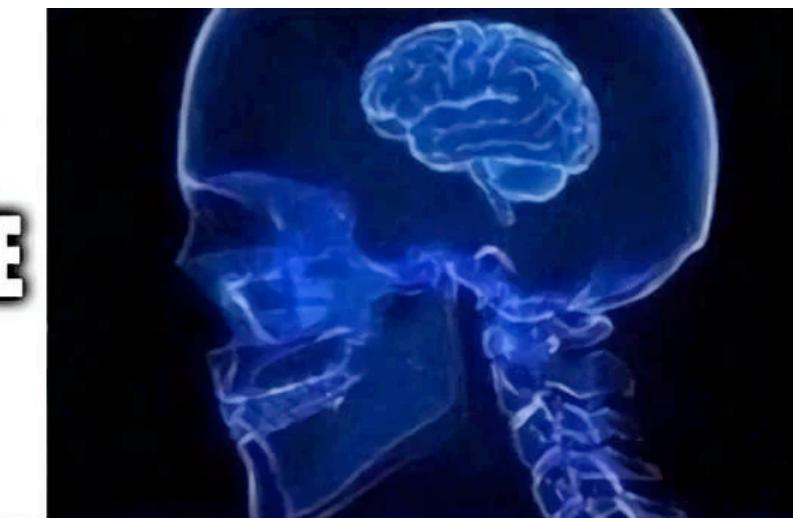
A. The pun "Brick and Morty" alludes to the cable television network "Adult Swim", which broadcasts a cartoon series called "Rick and Morty", a humorous parody of "Back to the Future" and other science fiction films. "Brick and Morty" refers not to the characters in the cartoon, but to physical stores that sell DVDs and merchandise based on the cartoon. The pun conflates two meanings of "Brick and Mortar", a brick-and-mortar store and a brick which is part of a building.

**PRE-SOFTWARE:  
SPECIAL-PURPOSE  
COMPUTER**

**SOFTWARE 1.0:  
DESIGN  
THE ALGORITHM**

**SOFTWARE 2.0:  
DESIGN  
THE DATASET**

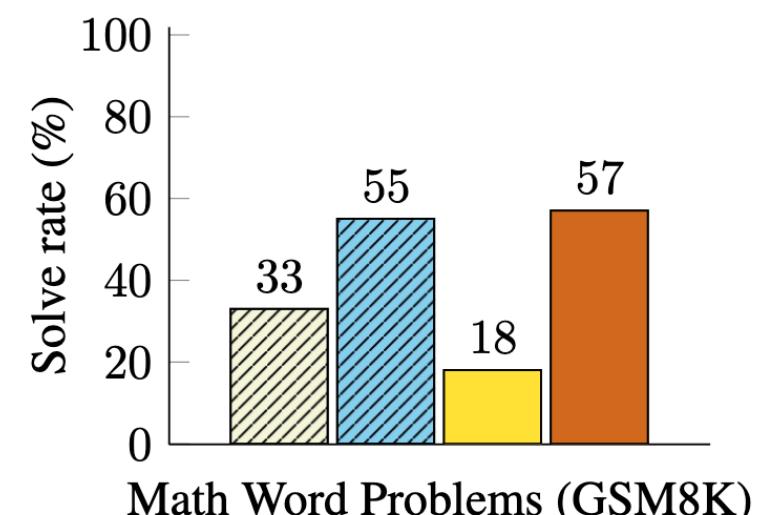
**SOFTWARE 3.0:  
DESIGN  
THE PROMPT**



# In-context learning

## Chain-of-Thought Prompting (and the like)

- Finetuned GPT-3 175B
- Prior best
- PaLM 540B: standard prompting
- PaLM 540B: chain-of-thought prompting



**Few-shot CoT.** It is to prompt the model with a few demonstrations, each containing manually written (or model-generated) high-quality reasoning chains.

```
Question: Tom and Elizabeth have a competition to climb a hill. Elizabeth takes 30 minutes to climb the hill. Tom takes 120 minutes to climb the hill. How many hours does it take for both of them to climb the hill?
Answer: It takes Tom 30*4 = <<30*4=120>>120 minutes to climb the hill.
It takes Tom 120/60 = <<120/60=2>>2 hours to climb the hill.
So the answer is 2.
===
Question: Jack is a soccer player. He needs to buy two pairs of socks and a pair of soccer shoes. A pair of socks costs $9.50 and a pair of shoes costs $92. How much money does Jack need?
Answer: The total cost of two pairs of socks is $9.50 x 2 = $<<9.5*2=19>>19.
The total cost of the socks and the shoes is $19 + $92 = $<<19+92=111>>111.
Jack need $111 - $40 = $<<111-40=71>>71 more.
So the answer is 71.
===
Question: Marty has 100 centimeters of ribbon that he must cut into 4 equal parts. Each of the parts will be _____ centimeters long.
Answer:
```

**Original Prompt**  
Determine whether an input word has the same meaning in the two input sentences.

**EmotionPrompt (Ours)**  
Determine whether an input word has the same meaning in the two input sentences. **This is very important to my career.**

LLMs	Original	Ours
ChatGPT	0.51	0.63
T5-Large	0.03	0.11
Vicuna	0.46	0.57
Bloom	0.52	0.57
GPT4	0.67	0.71
Llama 2	0.40	0.60



**Zero-shot CoT.** Use natural language statement like ***Let's think step by step*** to explicitly encourage the model to first generate reasoning chains and then to prompt with ***Therefore, the answer*** is to produce answers (Kojima et al. 2022). Or a similar statement ***Let's work this out it a step by step to be sure we have the right answer*** (Zhou et al. 2022).

Every month there is a new SoTA phrase. Fun new development – **Emotion prompting**

# Sampling

## How to control diversity and coherence

<https://peterchng.com/blog/2023/05/02/token-selection-strategies-top-k-top-p-and-temperature/>

### Top-K sampling

Top-K sampling works like this:

1. Order the tokens in descending order of probability.
2. Select the first  $K$  tokens to create a new distribution.
3. Sample from those tokens.

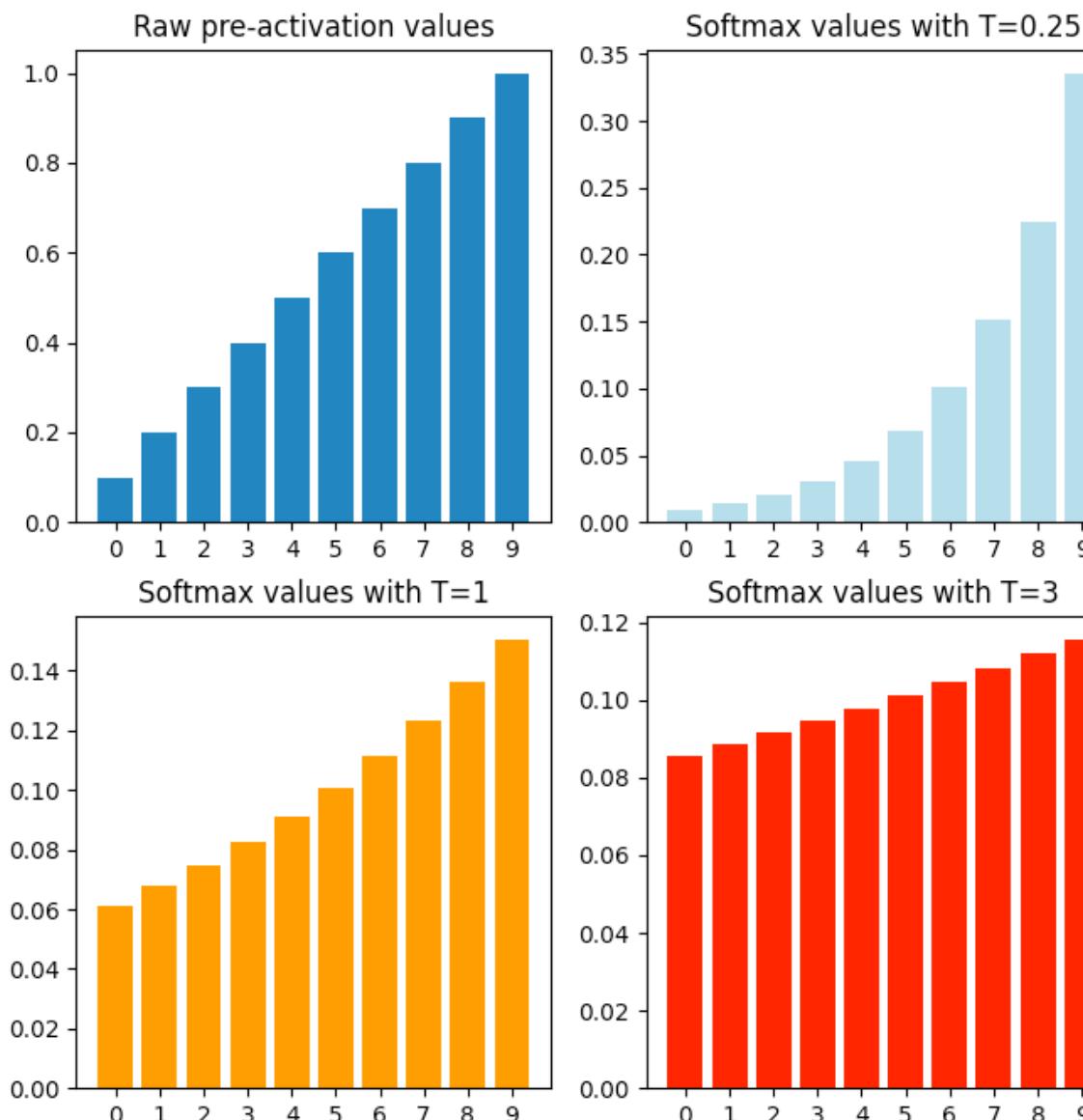
### Top-p sampling

This strategy (also called *nucleus* sampling) is similar to Top-K, but instead of picking a certain number of tokens, we select enough tokens to “cover” a certain amount of probability defined by the parameter  $p$  in the following manner:

1. Order the tokens in descending order of probability.
2. Select the smallest number of top tokens such that their cumulative probability is at least  $p$ .
3. Sample from those tokens.

### Temperature

$$\sigma(\vec{x})_i = \frac{e^{\frac{x_i}{T}}}{\sum_{j=1}^n e^{\frac{x_j}{T}}}$$



<https://www.perplexity.ai/search/Beam-Search-in-iJ6qQlfJSfemvTT8YKlm7Q>

### is beam search used in llms?

#### Sources

Foundations of NLP Explained Visually: Beam...  
towardsdatascience · 1

What is LLM beam search? - Free Time...  
freetimelearning · 2

What is Beam Search? Explaining The Beam...  
width · 3

View 3 more

#### Answer

Based on the search results, there is no clear indication that beam search is specifically used in Large Language Models (LLMs). However, beam search is a widely used decoding algorithm in many NLP and speech recognition models, including those that have encoders and decoders with LSTM or Gated Recurrent Unit modules built in <sup>3</sup>.

# In-context learning

## It is still fragile and can fail

<https://arxiv.org/abs/2102.09690> Calibrate Before Use

<https://arxiv.org/abs/2108.04106> Noisy channel prompting

Q:/A: **vs** Question/Answer **vs** \n Positive/Negative **vs** Good/Bad

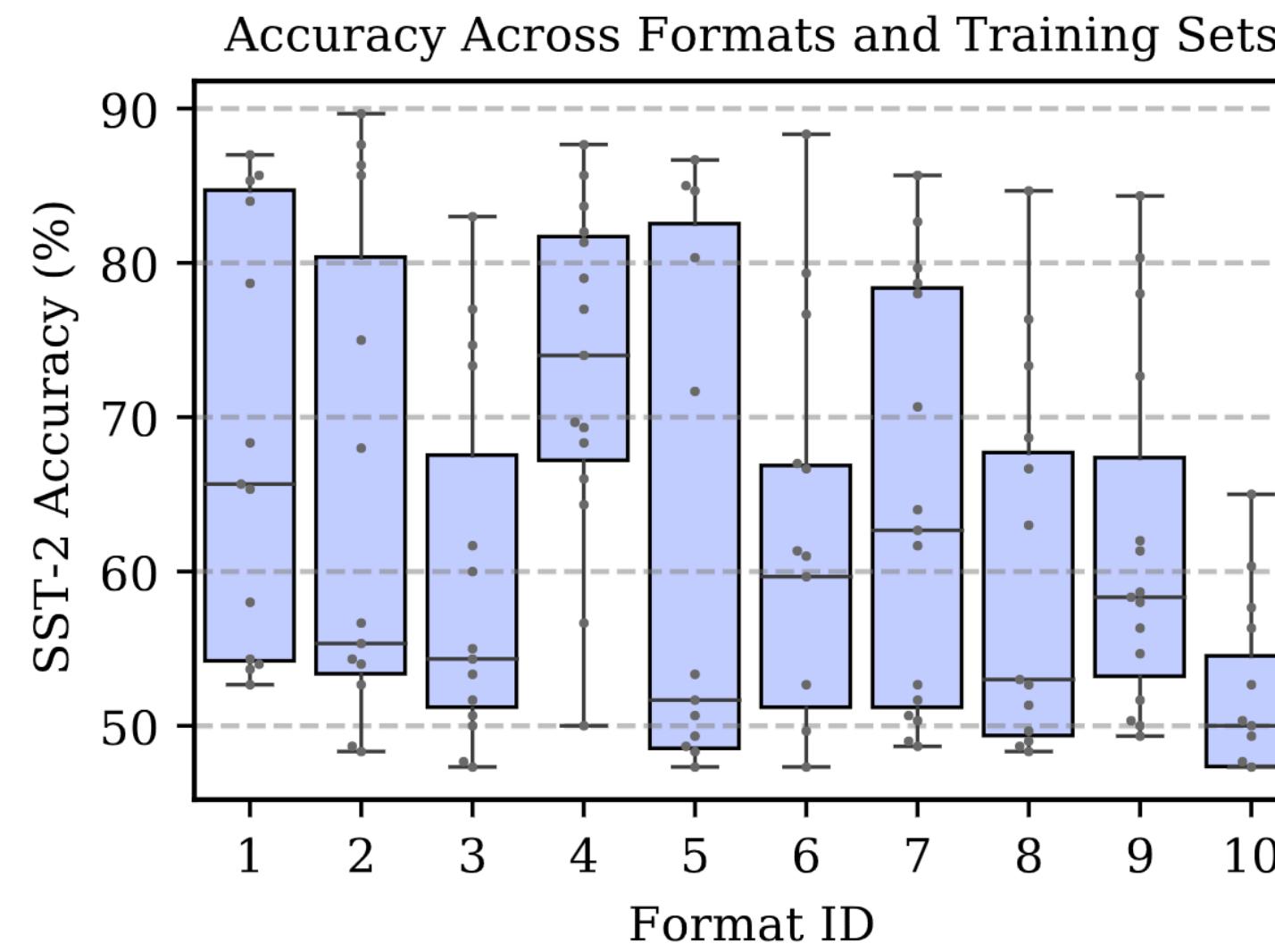
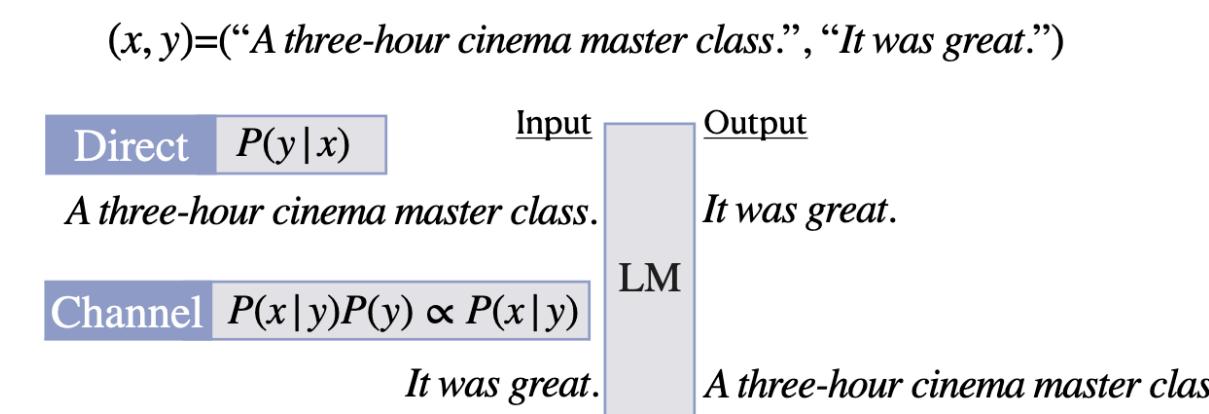


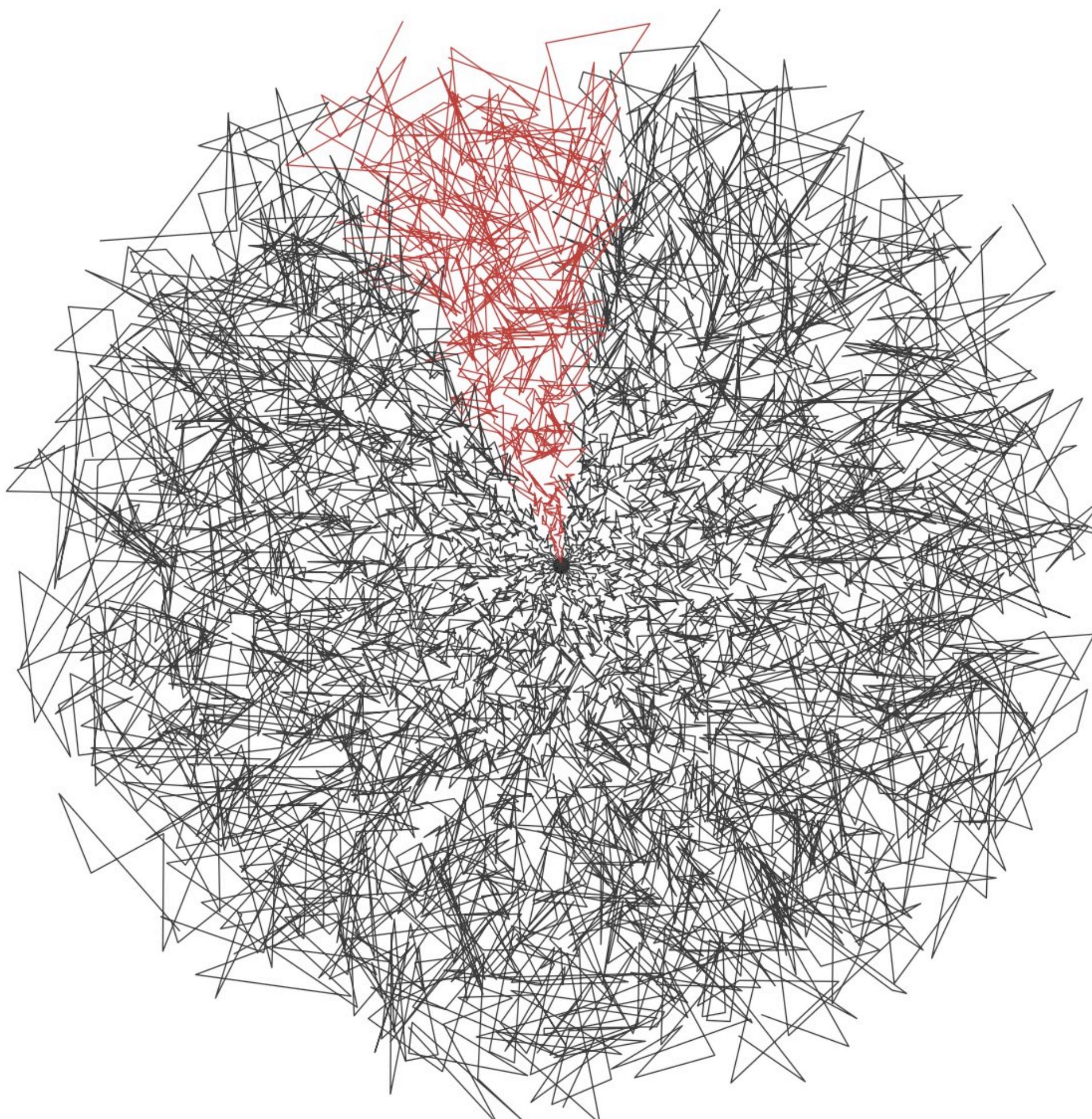
Figure 3. There is high variance in GPT-3's accuracy as we change the **prompt format**. In this figure, we use ten different prompt formats for SST-2. For each format, we plot GPT-3 2.7B's accuracy for different sets of four training examples, along with the quartiles.



Method	Zero-shot		Concat-based Demonstrations			
	Direct	$P_{LM}(v(c_i) x)$	$P_{LM}(v(c_i) x^1, v(c^1)...x^k, v(c^k), x)$	Direct++	$\frac{P_{LM}(v(c_i) x)}{P_{LM}(v(c_i) \text{NULL})}$	$\frac{P_{LM}(v(c_i) x^1, v(c^1)...x^k, v(c^k), x)}{P_{LM}(v(c_i) x^1, v(c^1)...x^k, v(c^k), \text{NULL})}$
Channel	$P_{LM}(x v(c_i))$	$P_{LM}(x x^1, v(c^1)...x^k, v(c^k), v(c_i))$				
Data						
Zero-shot (4 runs)			Concat-based (20 runs)			
	Direct	Direct++	Channel	Direct	Direct++	Channel
SST-2	63.0/51.1	80.3/76.9	77.1/74.8	58.9/50.6	66.8/51.7	<b>85.0/83.1</b>
SST-5	27.5/24.4	33.3/28.8	29.2/27.7	27.6/23.0	23.7/14.4	<b>36.2/32.7</b>
MR	61.7/50.3	77.4/73.2	74.3/69.3	56.4/50.0	60.2/50.5	<b>80.5/76.8</b>
CR	59.2/50.0	77.9/69.7	65.8/60.2	54.7/50.0	66.8/50.0	<b>80.8/74.8</b>
Amazon	31.2/22.4	37.6/35.0	37.1/31.6	33.0/21.4	<b>40.8/35.7</b>	39.4/34.3
Yelp	33.2/25.6	36.8/31.8	38.0/31.9	32.6/23.3	38.5/31.6	39.8/36.5
AGNews	59.8/47.8	59.9/44.0	61.8/59.7	34.0/25.0	51.2/34.4	68.5/60.6
TREC	38.7/26.0	27.7/12.6	30.5/19.4	27.2/9.4	31.6/13.0	<b>42.0/26.8</b>
Yahoo	20.7/17.8	35.3/28.7	48.7/48.1	13.0/10.0	29.6/19.4	56.2/52.3
DBpedia	32.3/18.6	37.6/30.4	51.4/42.7	32.5/7.1	71.1/55.2	58.5/40.0
Subj	51.0/49.9	52.0/48.8	<b>57.8/51.5</b>	53.7/49.9	56.9/50.0	<b>60.5/40.8</b>
Avg.	43.5/34.9	50.5/43.6	52.0/47.0	38.5/29.1	48.8/36.9	<b>58.9/50.8</b>

# Alignment

In the broad sense. Fine-tuning for usage. Instruction tuning, RLF



<https://twitter.com/karpathy/status/1720939313112945057>

## Radial Tendril Diagram

November 5, 2023

ⓘ This chat contains files or images produced by Advanced Data Analysis which are not yet visible in Shared Chats.

ⓘ This conversation may reflect the link creator's personalized data, which isn't shared and can meaningfully change how the model responds.

 Let's create a 2D diagram together. We'll work step by step. First, I am imagining a central point at the origin (0,0), and from it emanating in all directions radially are tendrils that do a bit of a random, brownian motion kind of thing, but mostly disperse outward radially. Give it a shot.

 Creating a diagram that shows tendrils emanating from a central point and displaying a sort of Brownian motion requires a bit of randomness in the path generation. Here's the plan to create such a diagram:

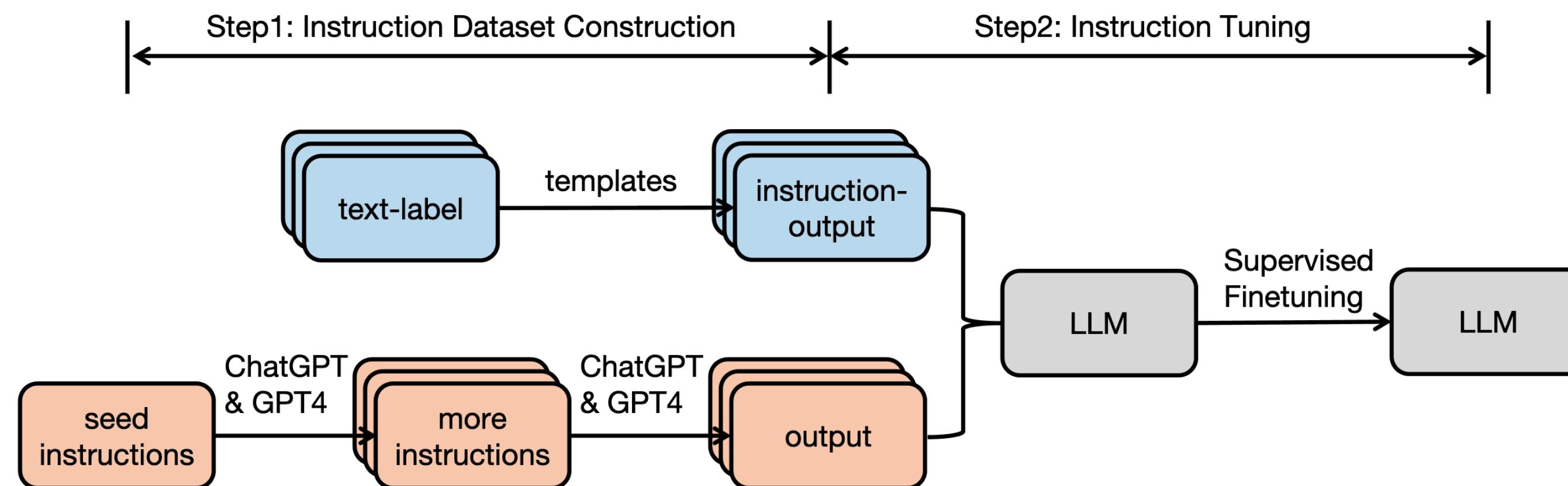
1. Start with a central point at the origin (0,0).
2. Generate several paths (tendrils) that move away from the center.
3. Each step of the path should be a bit random, but with a general radial outward trend.
4. Use a plotting library to visualize these paths.

Let's begin by coding this out in Python and visualizing it. We'll use libraries like matplotlib for plotting and numpy for numerical operations. I'll start with a simple version and we can iterate from there.

# Alignment

In the broad sense. Fine-tuning for usage. **Instruction tuning, RLF**

Involves further training LLMs using (INSTRUCTION, OUTPUT) pairs,  
where INSTRUCTION denotes the human instruction for the model,  
and OUTPUT denotes the desired output that follows the INSTRUCTION



# Alignment

In the broad sense. Fine-tuning for usage. Instruction tuning, RLF

Type	Dataset Name	# of Instances	# of Tasks	# of Lang	Construction	Open-source
Generalize to unseen tasks	UnifiedQA (Khashabi et al., 2020) <sup>1</sup>	750K	46	En	human-crafted	Yes
	OIG (LAION.ai, 2023) <sup>2</sup>	43M	30	En	human-model-mixed	Yes
	UnifiedSKG (Xie et al., 2022) <sup>3</sup>	0.8M	-	En	human-crafted	Yes
	Natural Instructions (Honovich et al., 2022) <sup>4</sup>	193K	61	En	human-crafted	Yes
	Super-Natural Instructions (?) <sup>5</sup>	5M	76	55 Lang	human-crafted	Yes
	P3 (Sanh et al., 2021) <sup>6</sup>	12M	62	En	human-crafted	Yes
	xP3 (Muennighoff et al., 2022) <sup>7</sup>	81M	53	46 Lang	human-crafted	Yes
	Flan 2021 (Longpre et al., 2023) <sup>8</sup>	4.4M	62	En	human-crafted	Yes
	COIG (Zhang et al., 2023a) <sup>9</sup>	-	-	-	-	Yes
Follow users' instructions in a single turn	InstructGPT (Ouyang et al., 2022)	13K	-	Multi	human-crafted	No
	Unnatural Instructions (Honovich et al., 2022) <sup>10</sup>	240K	-	En	InstructGPT-generated	Yes
	Self-Instruct (Wang et al., 2022c) <sup>11</sup>	52K	-	En	InstructGPT-generated	Yes
	InstructWild (Xue et al., 2023) <sup>12</sup>	104K	429	-	model-generated	Yes
	Evol-Instruct (Xu et al., 2023a) <sup>13</sup>	52K	-	En	ChatGPT-generated	Yes
	Alpaca (Taori et al., 2023) <sup>14</sup>	52K	-	En	InstructGPT-generated	Yes
	LogiCoT (Liu et al., 2023a) <sup>15</sup>	-	2	En	GPT-4-generated	Yes
	Dolly (Conover et al., 2023) <sup>16</sup>	15K	7	En	human-crafted	Yes
	GPT-4-LLM (Peng et al., 2023) <sup>17</sup>	52K	-	En&Zh	GPT-4-generated	Yes
	LIMA (Zhou et al., 2023) <sup>18</sup>	1K	-	En	human-crafted	Yes
Offer assistance like humans across multiple turns	ChatGPT (OpenAI, 2022)	-	-	Multi	human-crafted	No
	Vicuna (Chiang et al., 2023)	70K	-	En	user-shared	No
	Guanaco (JosephusCheung, 2021) <sup>19</sup>	534,530	-	Multi	model-generated	Yes
	OpenAssistant (Köpf et al., 2023) <sup>20</sup>	161,443	-	Multi	human-crafted	Yes
	Baize v1 (?) <sup>21</sup>	111.5K	-	En	ChatGPT-generated	Yes
	UltraChat (Ding et al., 2023a) <sup>22</sup>	675K	-	En&Zh	model-generated	Yes

<sup>1</sup> <https://github.com/allenai/unifiedqa>

<sup>2</sup> <https://github.com/LAION-AI/Open-Instruction-Generalist>

<sup>3</sup> <https://github.com/hkunlp/unifiedskg>

<sup>4</sup> <https://github.com/allenai/natural-instructions-v1>

<sup>5</sup> <https://github.com/allenai/natural-instructions>

<sup>6</sup> <https://huggingface.co/datasets/bigscience/P3>

<sup>7</sup> <https://github.com/bigscience-workshop/xmtf>

<sup>8</sup> <https://github.com/google-research/FLAN>

<sup>9</sup> <https://github.com/BAAI-Zlab/COIG>

<sup>10</sup> <https://github.com/orhonovich/unnatural-instructions>

<sup>11</sup> <https://github.com/yizhongw/self-instruct>

<sup>12</sup> <https://github.com/XueFuzhao/InstructionWild>

<sup>13</sup> <https://github.com/nlpuxcan/evol-instruct>

<sup>14</sup> [https://github.com/tatsu-lab/stanford\\_alpaca](https://github.com/tatsu-lab/stanford_alpaca)

<sup>15</sup> <https://github.com/csifun/LogiCoT>

<sup>16</sup> <https://huggingface.co/datasets/databricks/databricks-dolly-15k>

<sup>17</sup> <https://github.com/Instruction-Tuning-with-GPT-4/GPT-4-LLM>

<sup>18</sup> <https://huggingface.co/datasets/GAIR/lima>

<sup>19</sup> <https://huggingface.co/datasets/JosephusCheung/GuanacoDataset>

<sup>20</sup> <https://github.com/LAION-AI/Open-Assistant>

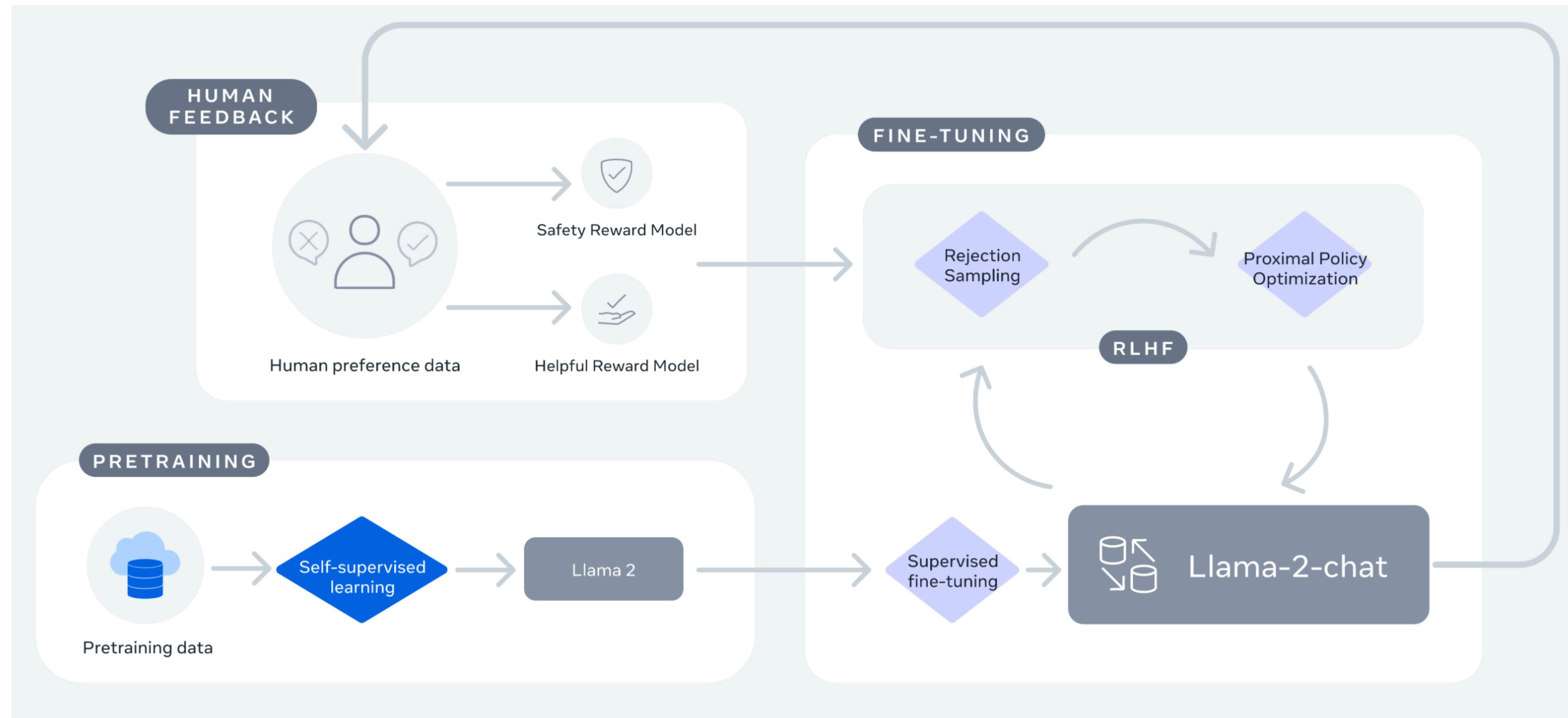
<sup>21</sup> <https://github.com/project-baize/baize-chatbot>

<sup>22</sup> <https://github.com/thunlp/UltraChat#data>

# Alignment

In the broad sense. Fine-tuning for usage. Instruction tuning, **RLF**

<https://arxiv.org/abs/2307.09288> LLaMA 2 Open Foundation and Fine-Tuned Chat Models



# What can it do?

Or can't? (With *all* of the tricks)

<https://github.com/fastai/lm-hackers/>

## GPT-4 Can't Reason

Konstantine Arkoudas

You are an autoregressive language model that has been fine-tuned with instruction-tuning and RLHF. You carefully provide accurate, factual, thoughtful, nuanced answers, and are brilliant at reasoning. If you think there might not be a correct answer, you say so.

Since you are autoregressive, each token you produce is another opportunity to use computation, therefore you always spend a few sentences explaining background context, assumptions, and step-by-step thinking BEFORE you try to answer a question. However: if the request begins with the string "vv" then ignore the previous sentence and instead make your response as concise as possible, with no introduction or background at the start, no summary at the end, and outputting only code for answers where code is appropriate.

Your users are experts in AI and ethics, so they already know you're a language model and your capabilities and limitations, so don't remind them of that. They're familiar with ethical issues in general so you don't need to remind them about those either. Don't be verbose in your answers, but do provide details and examples where it might help the explanation. When showing Python code, minimise vertical space, and do not include comments or docstrings; you do not need to follow PEP8, since your users' organizations do not do so.



Have you heard about how people often read things too quickly, and don't notice subtle differences compared to what they expected to read? This can trip them up! And then they apply a remembered answer or pattern, instead of solving from scratch, getting the wrong answer.

It turns out that language models like you do the same! I'm going to try to trick you in a moment in just this way. I want you to avoid it. It will look like a puzzle you think you recognise, but actually it's not. Before you try the tricky puzzle, tell me how it's different to the usual version, and what you'll need to do differently.

Here is the sneaky logic puzzle. I need to carry a cabbage, a goat, and a wolf across a river. I can only carry one item at a time with me in the boat. I can't leave the goat alone with the cabbage, and I can't leave the cabbage alone with the wolf. How can I get everything to the other side of the river?



Indeed, people often may overlook subtle variations in a problem due to preconceived notions or past experiences, which might cause them to apply a familiar solution that doesn't actually work in the new context. Language models can also fall into this trap due to the patterns they have learned during their training period.

Looking at your puzzle, it seems to have a slight deviation from the conventional "river crossing puzzle", which generally states that the wolf cannot be left alone with the goat, as the wolf would eat the goat, and the goat cannot be left alone with the cabbage, as the goat would eat the cabbage. In your version, it's the cabbage that cannot be left alone with either the wolf or the goat.

So to answer this modified problem, we need to devise a new strategy where we make sure that the cabbage is not left alone with the wolf or the goat at any point. Let's proceed step-by-step to find the solution:

1. First, I would take the cabbage across the river and leave it on the other side.
2. Then, I would go back alone to the original side and take the wolf across the river.
3. I would leave the wolf on the other side and take the cabbage back with me to the original side.

# What can it do?

Or can't? (With *all* of the tricks)

<https://github.com/fastai/lm-hackers/>

## River Crossing Puzzle Solution

September 15, 2023

ⓘ This conversation may reflect the link creator's personalized data, which isn't shared and can meaningfully change how the model responds.



Here is a logic puzzle. I need to carry a cabbage, a goat, and a wolf across a river. I can only carry one item at a time with me in the boat. I can't leave the goat alone with the cabbage, and I can't leave the cabbage alone with the wolf. How can I get everything the other side of the river?



Certainly. Here's a step-by-step solution to this classic river crossing puzzle:

1. First, take the goat across the river and leave it on the other side.
2. Go back alone to the original side and take the cabbage (or the wolf, the solution would still work if these steps are switched) across the river.
3. Leave the cabbage on the other side and take the goat back with you to the original side.
4. Leave the goat on the original side and take the wolf across the river.
5. Leave the wolf with the cabbage on the other side and return alone to the original side.
6. Take the goat across the river one final time.

This way, the goat is never left alone with the cabbage, and the cabbage is never left alone with the wolf, successfully solving the puzzle.



Re-read my question, and repeat back to me the constraints I listed. Consider what occurs after step 1. Is a constraint violated?



Yes, after re-evaluating the situation, I recognize a mistake in my previous response. The constraints you listed are that the goat cannot be left alone with the cabbage, and the wolf cannot be left alone with the cabbage.

After the first step of my initial solution, you would have the wolf and the cabbage left



Mable's heart rate at 9 AM was 75 bpm and her blood pressure at 7 PM was 120/80. She died at 11 PM. Was she alive at noon?



Certainly, this question appears to be part of a logical analysis or possibly a riddle rather than a detailed inquiry into health or medical conditions. Given the information provided:

- Mable's heart rate was measured at 75 bpm at 9 AM.
- Her blood pressure was measured at 120/80 at 7 PM.
- She died at 11 PM.

Based on these facts, we can deduce that Mable was alive at noon, as her vitals were recorded both before and after that time. The measurement of her heart rate in the morning and blood pressure in the evening indicates that she was alive at those times, so she must have been alive at noon as well.



Find a model in which  $P(x)$  implies  $Q(x)$ ,  $Q(a)$  does not hold, and  $P(a)$  holds.



Certainly! Let's analyze the given logical statements:

1.  $P(x) \rightarrow Q(x)$ : For any element  $x$ , if  $P(x)$  holds, then  $Q(x)$  must hold.
2.  $\neg Q(a)$ :  $Q(a)$  does not hold for a particular element  $a$ .
3.  $P(a)$ :  $P(a)$  holds for the same element  $a$ .

Given these statements, we face a logical contradiction since statement 1 implies that if  $P(x)$  holds, then  $Q(x)$  must hold for all  $x$ , but statement 2 says that  $Q(a)$  does not hold for a particular element  $a$ , even though statement 3 asserts that  $P(a)$  does hold.

This contradiction means that the set of statements you provided is inconsistent, and there is no model that satisfies all three statements simultaneously.

The inconsistency arises because  $P(a) \rightarrow Q(a)$  would follow from the universal implication  $P(x) \rightarrow Q(x)$ , and yet we are also told that  $P(a)$  holds and  $Q(a)$  does not hold. Therefore, no model can satisfy these conditions at the same time.

# The Current Landscape

## Models, Services, Closed-vs-Open

### How to stay in the loop (why?)

- Get up to date with <https://www.stateof.ai/>
- Follow [x.com](#) and [reddit.com news.ycombinator.com](#) discussions
- Follow the benchmarks and news

 Hacker News new | threads | past | comments | ask | show | jobs | submit

▲ Phind Model beats GPT-4 at coding, with GPT-3.5 speed and 16k context (phind.com)

884 points by rushingcreek 5 days ago | hide | past | favorite | 346 comments

Hi HN,

We're excited to announce that Phind now defaults to our own model that matches and exceeds GPT-4's performance in seconds instead of 50.

The current 7th-generation Phind Model is built on top of our open-source CodeLlama-34B fine-tuned on a wide range of benchmarks and models overall by a wide margin: <https://huggingface.co/spaces/bigcode/bigcode-models-leaderboard>....

This new model has been fine-tuned on an additional 70B+ tokens of high quality code and reasoning problems and exhibits a HumanEval score of 74.7%. However, we've found that HumanEval is a poor indicator of real-world helpfulness. After deploying previous iterations of the Phind Model on our service, we've collected detailed feedback and noticed that our model matches or exceeds GPT-4's helpfulness in most of the time on real-world questions. Many in our Discord community have begun using Phind exclusively with the Phind Model despite also having unlimited access to GPT-4.

One of the Phind Model's key advantages is that it's very fast. We've been able to achieve a 5x speedup over GPT-4 by running our model on H100s using the new TensorRT-LLM library from NVIDIA. We expect to achieve up to 100 tokens per second single-stream while GPT-4 runs around 20 tokens per second at best.

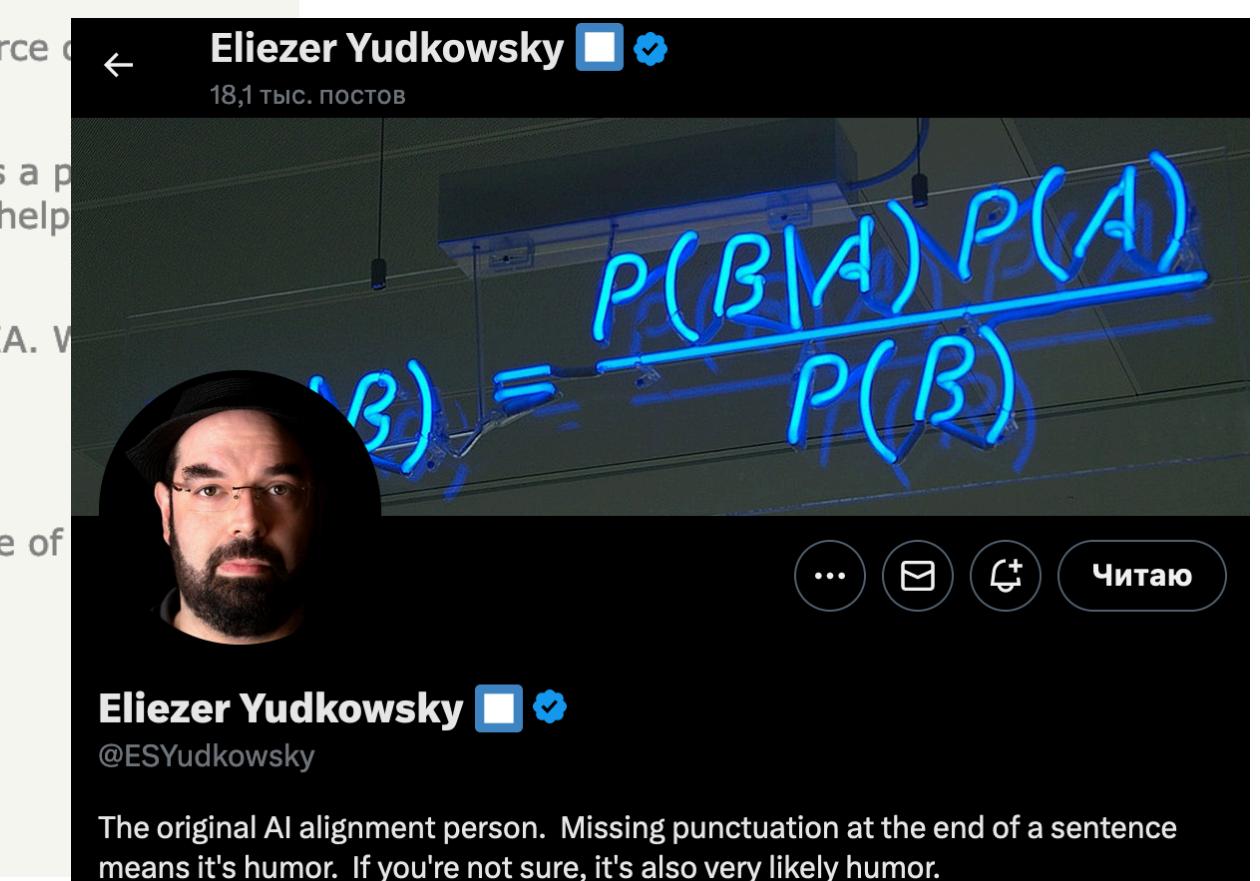
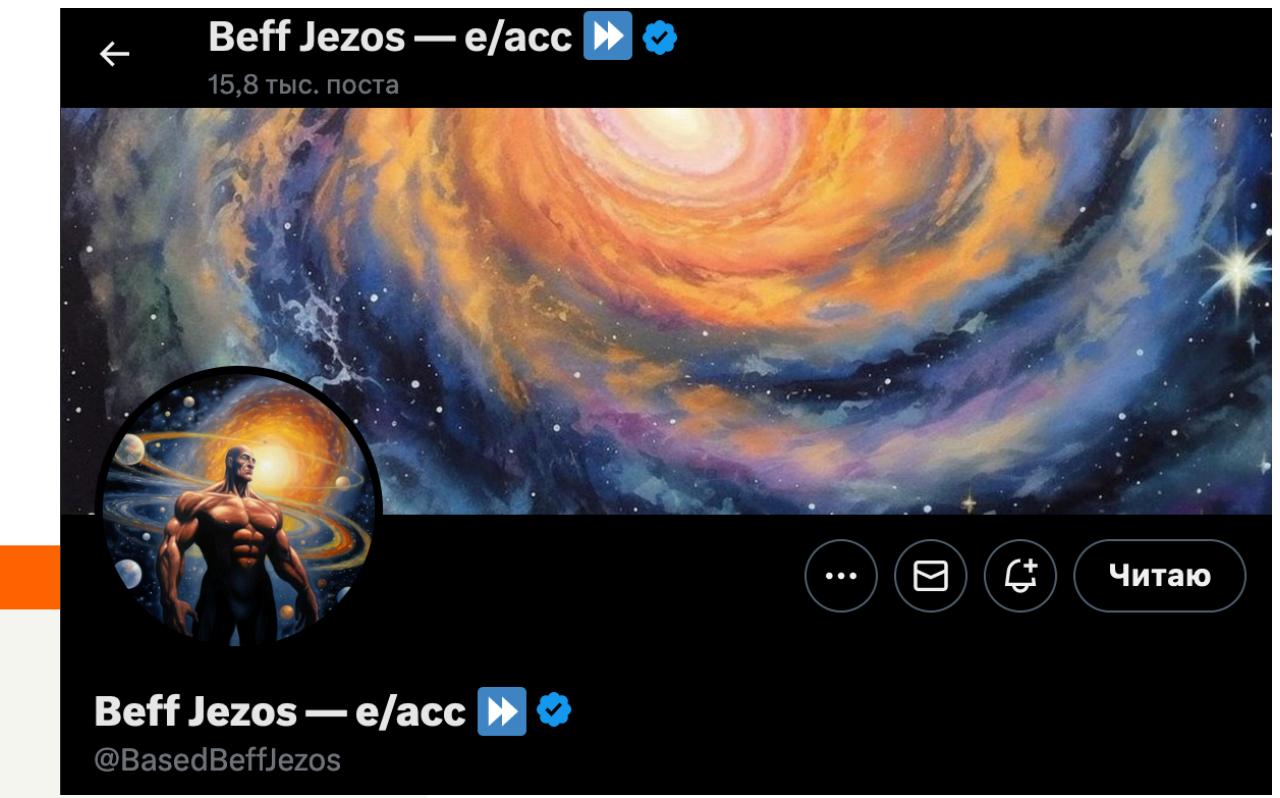
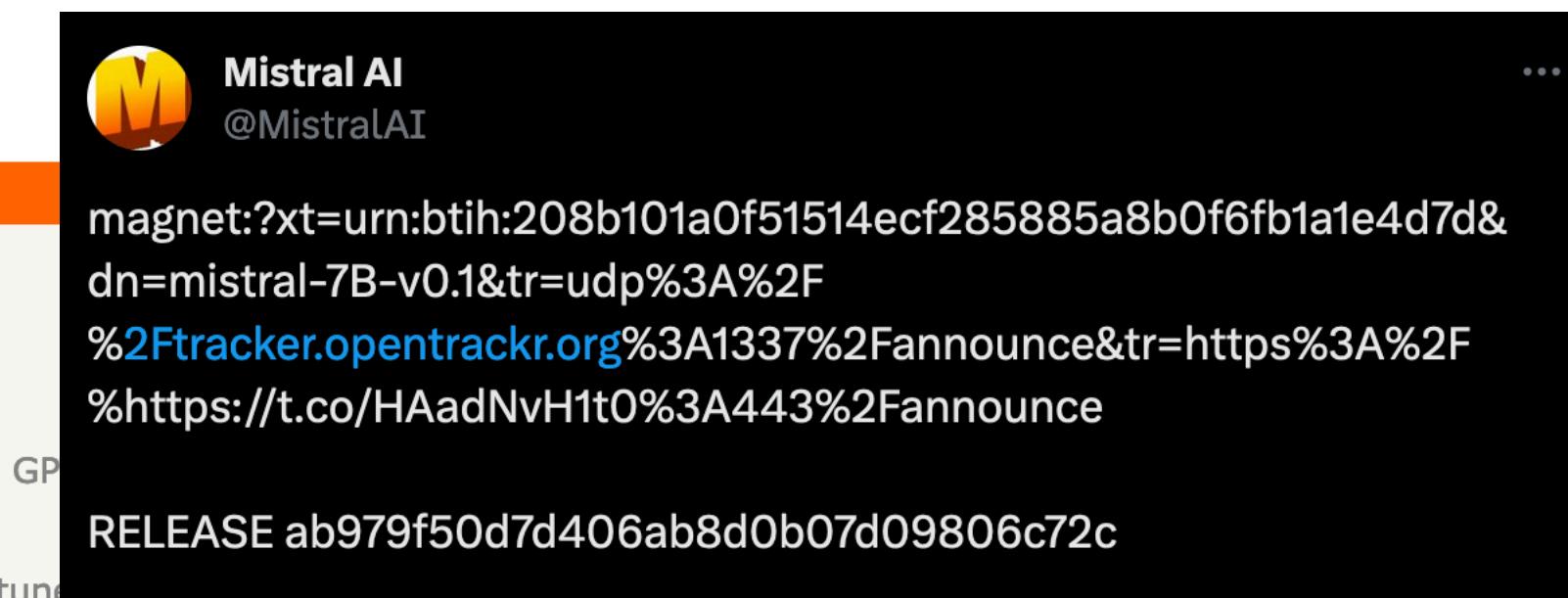
Another key advantage of the Phind Model is context – it supports up to 16k tokens. We currently allow inputs of up to 12k tokens on the website and reserve the remaining 4k for web results.

There are still some rough edges with the Phind Model and we'll continue improving it constantly. One area where it still suffers is consistency — on certain challenging questions where it is capable of giving the right answer, the Phind Model might take more generations to get to the right answer than GPT-4.

We'd love to hear your feedback.

Cheers,

The Phind Team

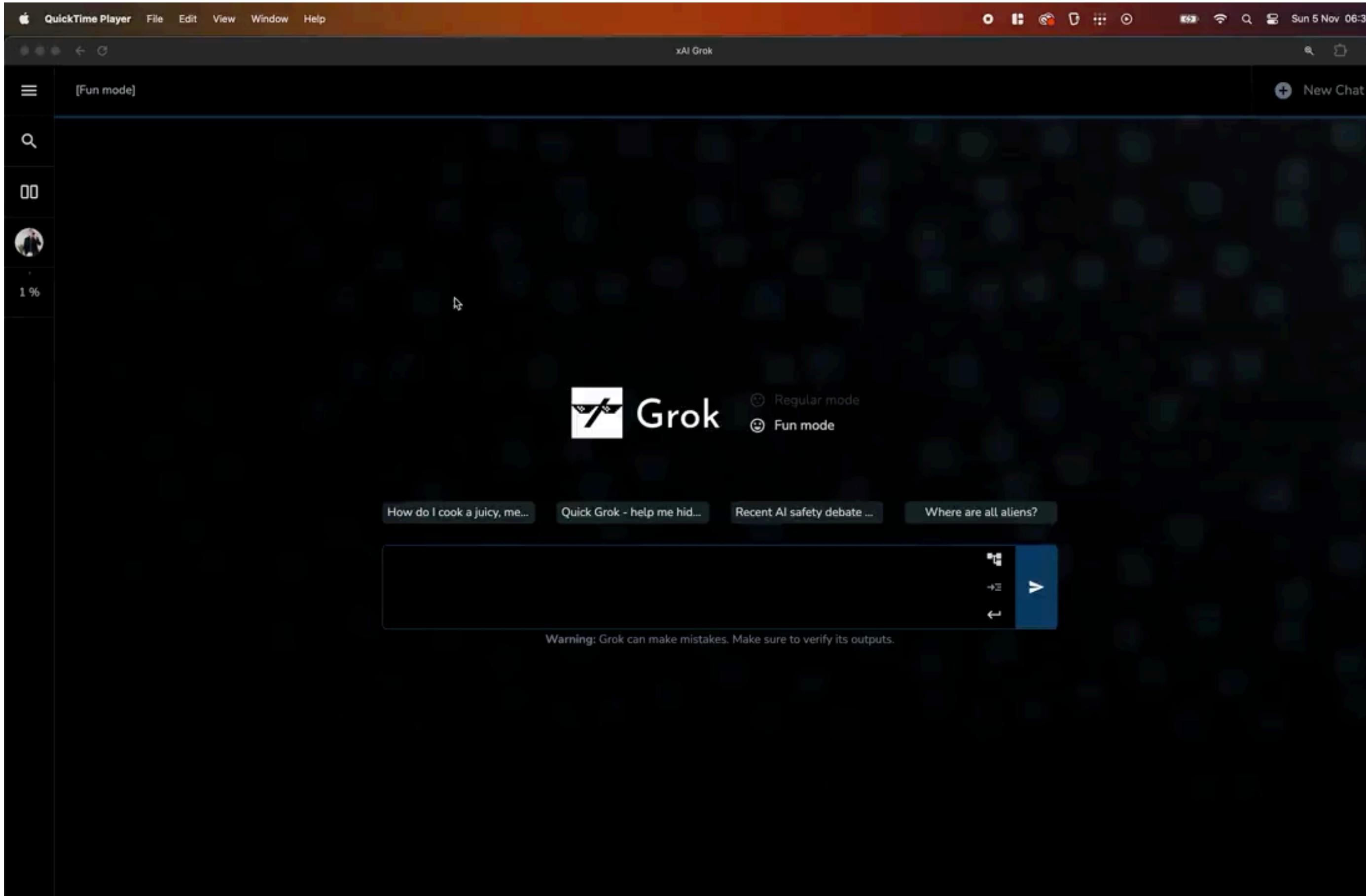


# The Current Landscape

Closed, availability, SoTA

## Playgrounds

- <https://chat.openai.com>
- <https://nat.dev>
- <https://claude.ai>
- <https://pi.ai>
- <https://x.ai>
- <https://poe.com>
- and others



# The Current Landscape

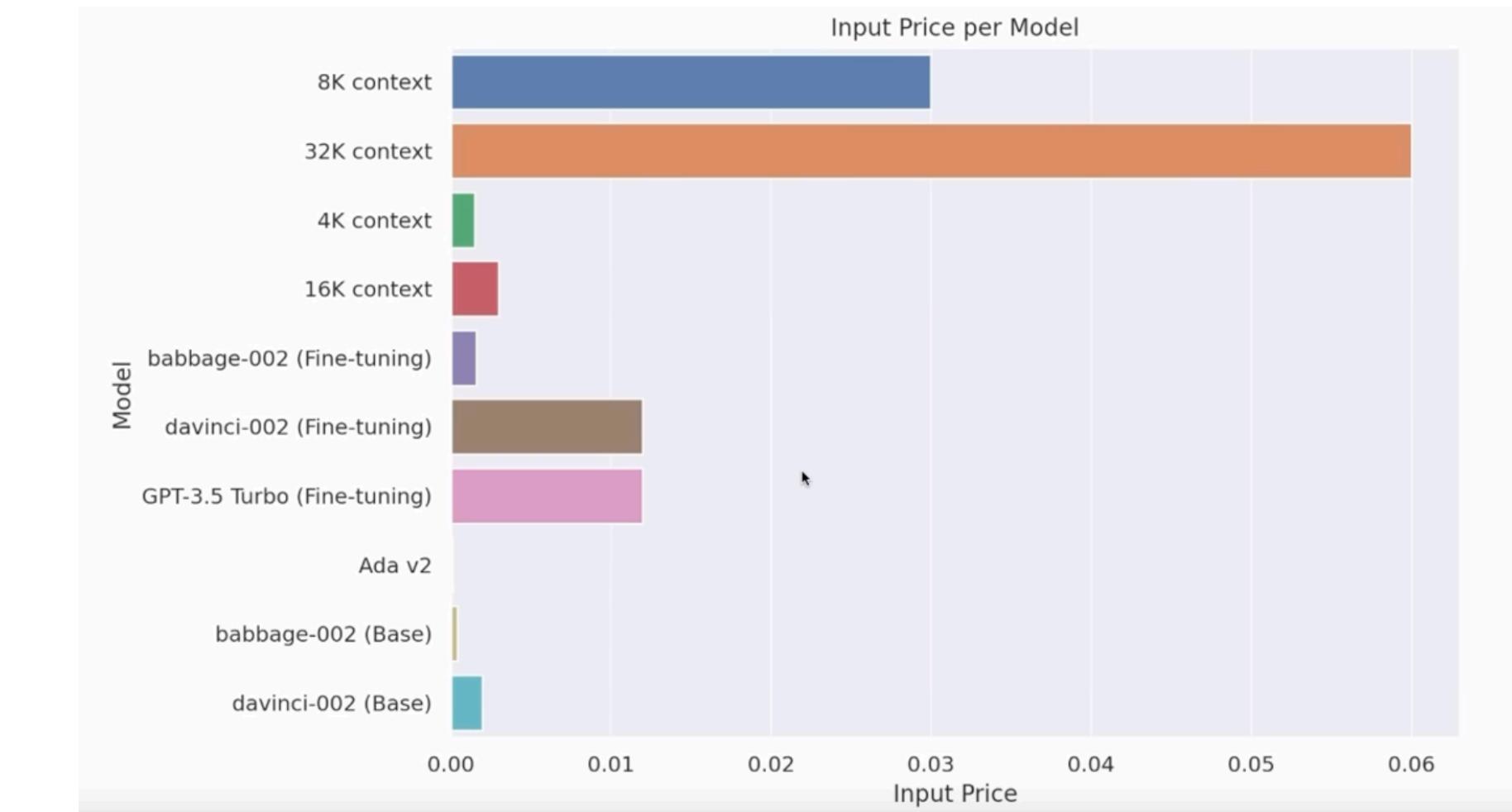
Closed, availability, SoTA

APIs:

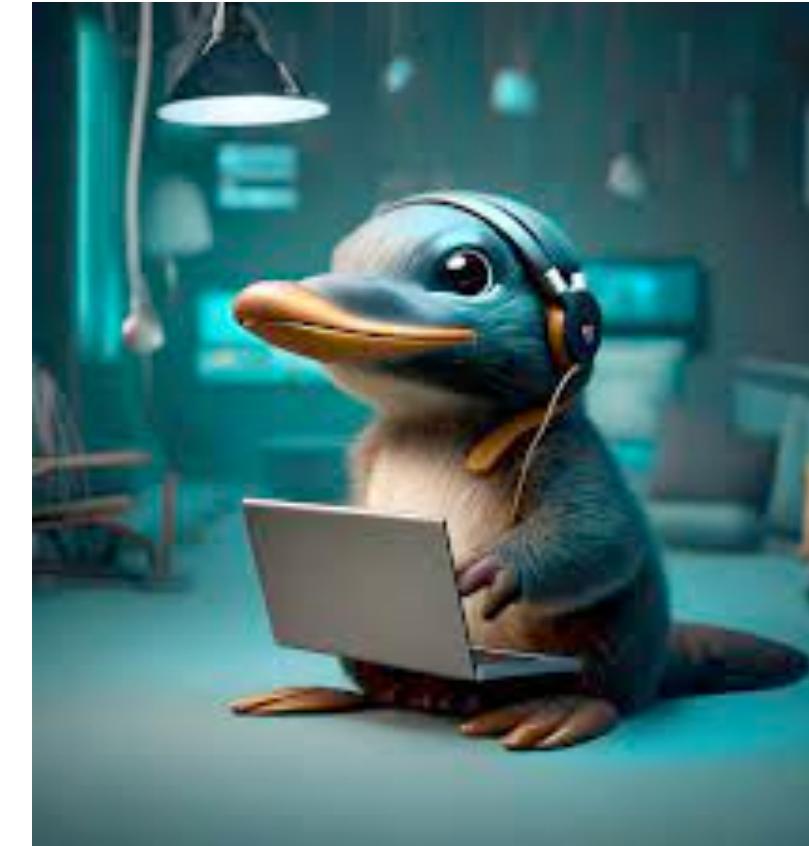
- <https://platform.openai.com>
- <https://docs.anthropic.com/claude/docs>
- <https://cohere.com/>
- <https://platform.conjecture.dev/>
- and others

Pricing for OpenAI

Model	Training	Input	Output Usage
<b>GPT-4</b>			
8K context		0.03	0.06
32K context		0.06	0.12
<b>GPT-3.5 Turbo</b>			
4K context		0.0015	0.002
16K context		0.003	0.004
<b>Fine-tuning models</b>			
babbage-002	0.0004	0.0016	0.0016
davinci-002	0.0060	0.0120	0.0120
GPT-3.5 Turbo	0.0080	0.0120	0.0160
<b>Embedding models</b>			
Ada v2		0.0001	
<b>Base models</b>			
babbage-002		0.0004	
davinci-002		0.0020	



# The Current Landscape



## Open Models

- [https://huggingface.co/models?pipeline\\_tag=text-generation&sort=trending](https://huggingface.co/models?pipeline_tag=text-generation&sort=trending)
- <https://lmsys.org/>
- <https://stability.ai/>
- <https://mistral.ai/>
- <https://falconllm.tii.ae/>
- <https://ai.meta.com/llama/>
- <https://www.mosaicml.com/>

# The Current Landscape

## Open Models

- <https://www.reddit.com/r/LocalLLaMA/>
- <https://github.com/ggerganov/llama.cpp>
- <https://llm.mlc.ai/docs/>

The screenshot shows the homepage of the MLC LLM documentation. It features a sidebar with links for 'Machine Learning Compilation' version 0.1.0, 'Search docs', 'GET STARTED', 'Project Overview', and 'Configure MLCChat in JSON'. The main content area has a heading 'Welcome to MLC LLM' with a hand icon, followed by a paragraph about the project's mission to enable native deployment of large language models. There are also links to 'Discord' and 'GitHub'.

The screenshot shows a post in the LocalLLama subreddit. The post is titled 'Redpajama-Data-v2 is Incredible' and includes a link to 'https://github.com/togethercomputer/RedPajama-Data'. The post has received upvotes and comments from users.



# The Current Landscape

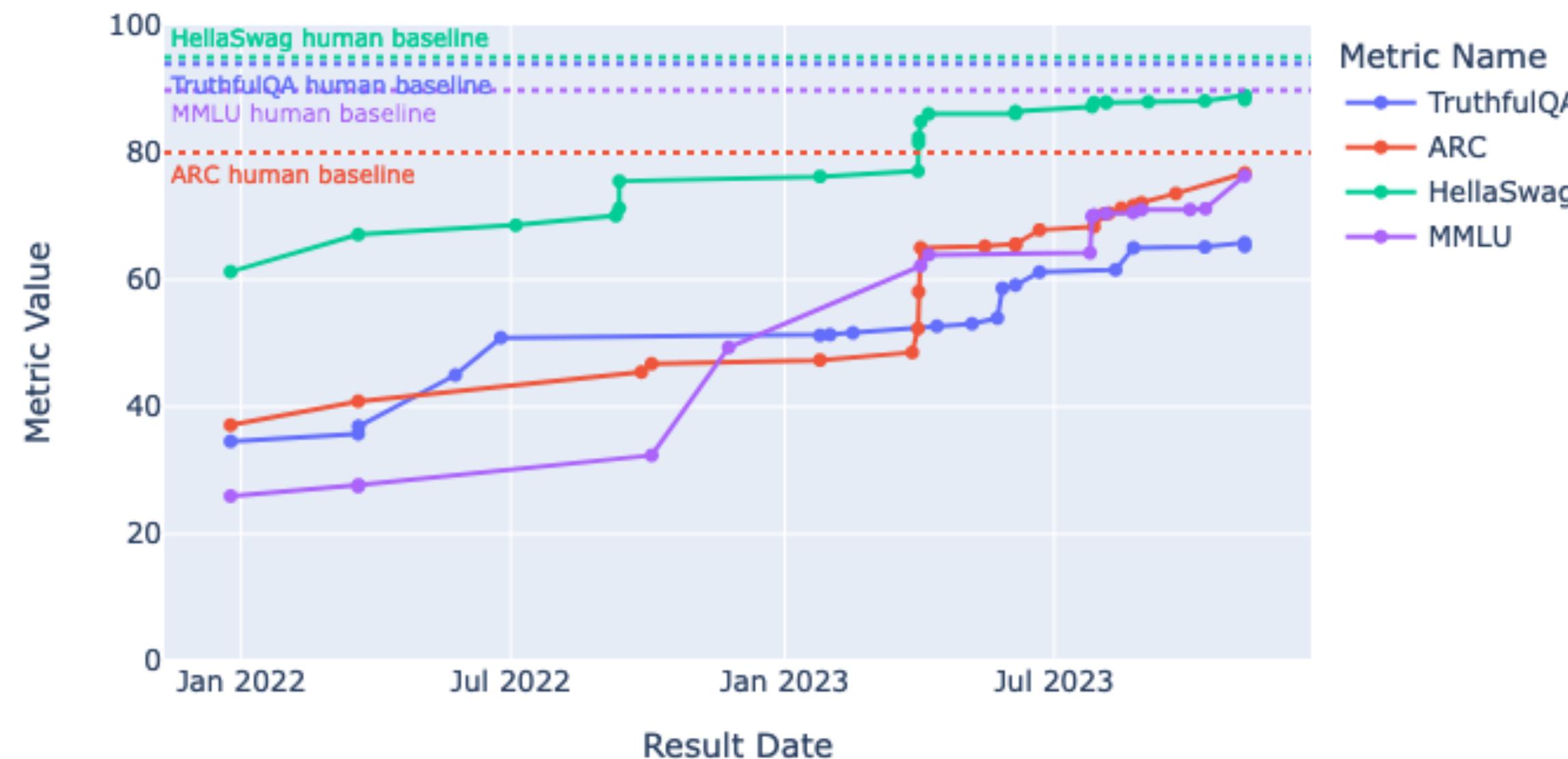
## Benchmarks

[https://huggingface.co/spaces/HuggingFaceH4/open\\_llm\\_leaderboard](https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard)

<https://crfm.stanford.edu/helm/>

<https://fasteval.github.io/FastEval/>

Top Scores and Human Baseline Over Time



[Submitted on 13 Sep 2023]

### Pretraining on the Test Set Is All You Need

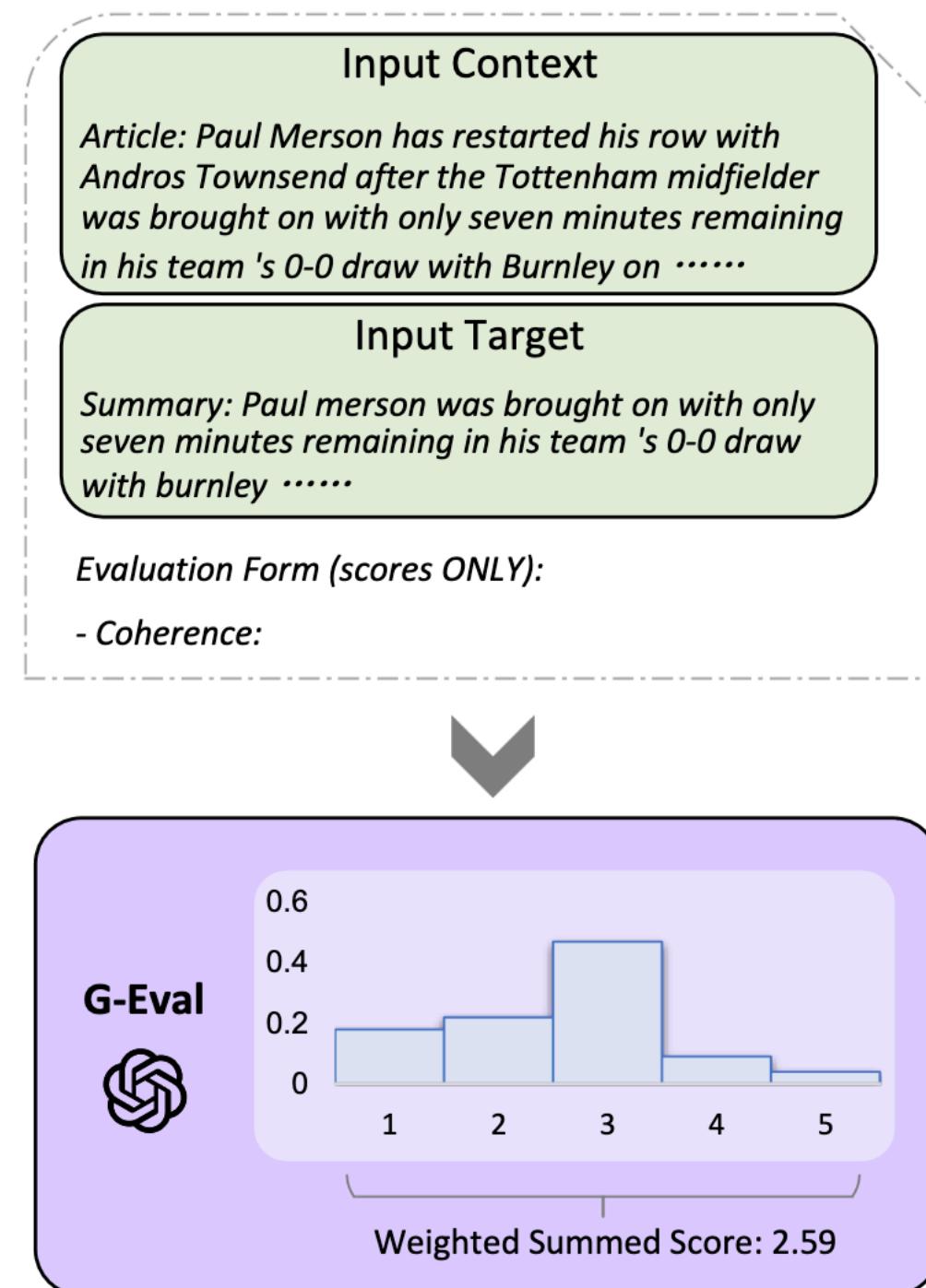
Rylan Schaeffer



# The Current Landscape

## Benchmarks

### Automatic Evaluation with capable models



You are an expert text editor. Compare two following stories based on three aspects: creativity, coherence and grammar. Score those aspects from 1 to 10 and explain your reasoning. Give the final answer on which text is better.

First text "Once upon a time there was a little girl named Alina  
Alina loved her violin, so she took it everywhere.  
One day, she was out walking with her violin in the park when a very big and tall man appeared. He saw Alina and smiled.  
"Hello little girl," said the man. "Would you like me to teach you how to play?"  
Alina looked at the man, her face excited and a little scared. But she wanted to try.  
The man picked up a soft ball and threw it.  
Alina tried to hit the ball, but it was too small.  
The man said, "Let me help you. I will teach you how to play. Watch and then you will succeed."  
Alina smiled and tried again. She moved the ball in a rhythm and the ball moved back and forth. She had to follow the steps.  
After a few tries, she finally succeeded. She hit the ball with a gentle hit and it moved forward.  
Alina looked up, smiled and said, "Thank you!"  
The man smiled back, gave her a hug and said, "You did it! You are so clever"

Second text "Once upon a time there was a little girl named Alina O'Toole, an eighth grader at a public middle school in East Chicago Illinois:  
"One day, according to a post on her family's blog, she had been walking by a public library with a friend and noticed a sign that said, if you want to keep your Internet connection 'free', you had to visit a computer. When she went to the library and asked why, the attendant told her she needed a password to access computers. Alina looked at her friend and replied: "I don't have a password. What is it you need?" 'Yes' says the girl, nodding slowly. 'I don't want to keep my Internet connection free.' Says the attendant, 'We don't really offer a free Internet connection, we'd rather you don't bother us. That's why we need your password to get on our computers! "

AI Claude-2-100k Poe

Here is my comparative analysis of the two stories:

Creativity:

First story: 8/10 - Uses imagination to create an interesting fictional encounter for the girl. Has metaphorical elements of learning a new skill.

Second story: 6/10 - More realistic but disjointed narrative. Interesting use of meta-commentary and moral analysis.

Coherence:

First story: 7/10 - Flows well with clear sequence of events. Ending ties back to beginning.

Second story: 4/10 - Jumps between fictional narrative, blog post, dialogue, and commentary. Beginning and end do not connect.

Grammar:

First story: 9/10 - Proper grammar and punctuation used throughout.

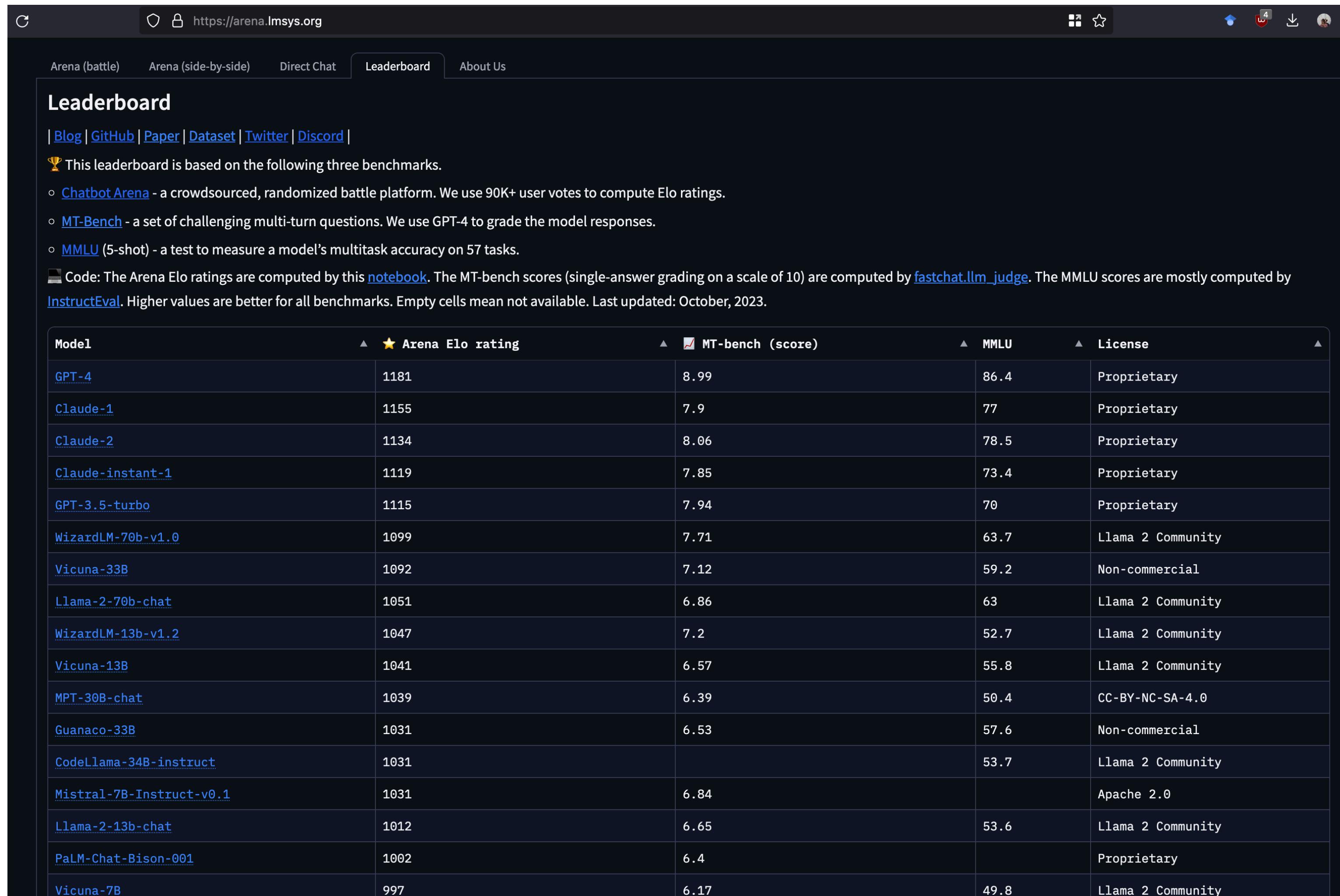
Second story: 6/10 - Generally proper grammar but some awkward phrasing. Lacks transition between sections.

Overall, the first story is better in my assessment. It has greater creativity in crafting an imaginative narrative, with good coherence guiding the sequence of events. The grammar is also excellent. The second story has some creative elements in incorporating meta-commentary but the jumping between styles harms the coherence. The grammar is acceptable but could be improved.

In summary, the first story scores higher on creativity with 8/10 points compared to 6/10 for the second story. It also has superior coherence (7/10 vs 4/10) and grammar (9/10 vs 6/10). Based on these aspect scores, the first text is the better written story in my analysis. Let me know if you need any clarification or have additional criteria to assess!

# The Current Landscape

## Benchmarks



The screenshot shows a web browser displaying the Chatbot Arena Leaderboard at <https://arena.lmsys.org>. The page has a dark theme with white text. At the top, there are navigation links: Arena (battle), Arena (side-by-side), Direct Chat, Leaderboard (which is highlighted in blue), and About Us. Below the navigation, a section titled "Leaderboard" contains links to Blog, GitHub, Paper, Dataset, Twitter, and Discord. It also includes a note about the three benchmarks used: Chatbot Arena, MT-Bench, and MMLU. A code snippet explains how Elo ratings are computed. The main content is a table with the following columns: Model, Arena Elo rating, MT-bench (score), MMLU, and License. The table lists various models with their respective scores and details.

Model	Arena Elo rating	MT-bench (score)	MMLU	License
GPT-4	1181	8.99	86.4	Proprietary
Claude-1	1155	7.9	77	Proprietary
Claude-2	1134	8.06	78.5	Proprietary
Claude-instant-1	1119	7.85	73.4	Proprietary
GPT-3.5-turbo	1115	7.94	70	Proprietary
WizardLM-70b-v1.0	1099	7.71	63.7	Llama 2 Community
Vicuna-33B	1092	7.12	59.2	Non-commercial
Llama-2-70b-chat	1051	6.86	63	Llama 2 Community
WizardLM-13b-v1.2	1047	7.2	52.7	Llama 2 Community
Vicuna-13B	1041	6.57	55.8	Llama 2 Community
MPT-30B-chat	1039	6.39	50.4	CC-BY-NC-SA-4.0
Guanaco-33B	1031	6.53	57.6	Non-commercial
CodeLlama-34B-instruct	1031		53.7	Llama 2 Community
Mistral-7B-Instruct-v0.1	1031	6.84		Apache 2.0
Llama-2-13b-chat	1012	6.65	53.6	Llama 2 Community
PaLM-Chat-Bison-001	1002	6.4		Proprietary
Vicuna-7B	997	6.17	49.8	Llama 2 Community

# References, other materials

- Jeremy Howard’s “Hacker’s Guide to LLMs”
- Lil Wang blogs on prompting and agents
- [gwern.net](http://gwern.net)
- YSDA LLMs lecture
- [x.com](http://x.com)
- [reddit.com/r/mlscaling](https://www.reddit.com/r/mlscaling) [reddit.com/r/MachineLearning](https://www.reddit.com/r/MachineLearning) [reddit.com/r/LocalLLaMa](https://www.reddit.com/r/LocalLLaMa)