



COMP 336 | Natural Language Processing

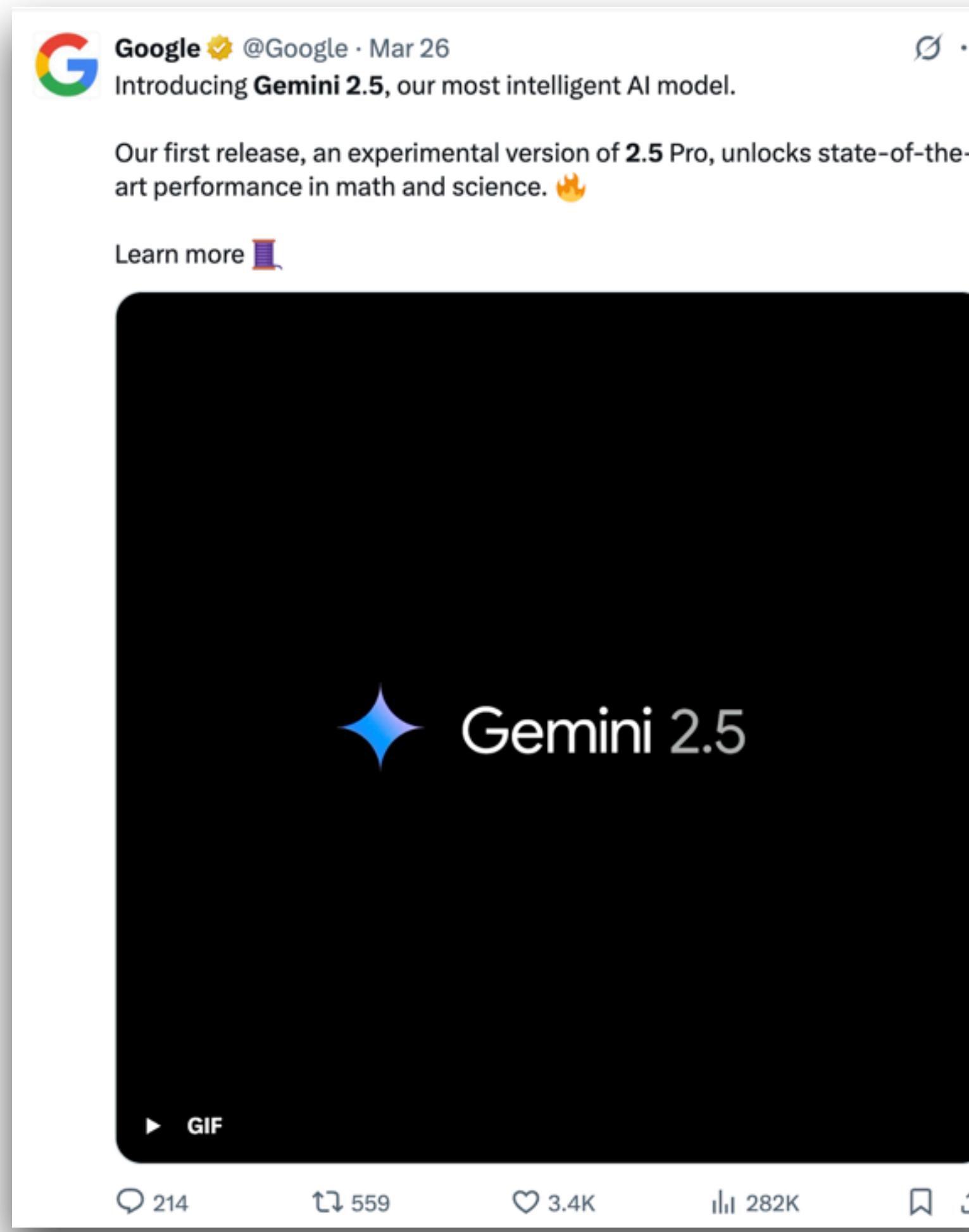
Lecture 15: LLM prompting, in-context learning,
scaling laws

Spring 2025

Announcements

- Final exam is scheduled at 2:30 pm - 4:30 pm on May 20 @Rm 167 Main Bldg.
- **#assignment-2** due next week!
 - Join **#assignment-2** Slack channel for discussion

Latest AI news



GPT4o image generation

Pretraining: training objectives?

- During pretraining, we have a large text corpus (**no task labels**)
 - **Key question: what labels or objectives used to train the vanilla Transformers?**



**Training
labels/objectives?**

Pretraining Transformers

Pretraining objectives



BERT (Encoder-only)

Devlin et al., 2018

The cabs ___ the same rates as those
___ by horse-drawn cabs and were ___
quite popular, ___ the Prince of
Wales (the ___ King Edward VII)
travelled in ___. The cabs quickly
___ known as "hummingbirds" for ___
noise made by their motors and their
distinctive black and ___ livery.
Passengers ___ ___ the interior
fittings were ___ when compared to
___ cabs but there ___ some
complaints ___ the ___ lighting made
them too ___ to those outside ___.

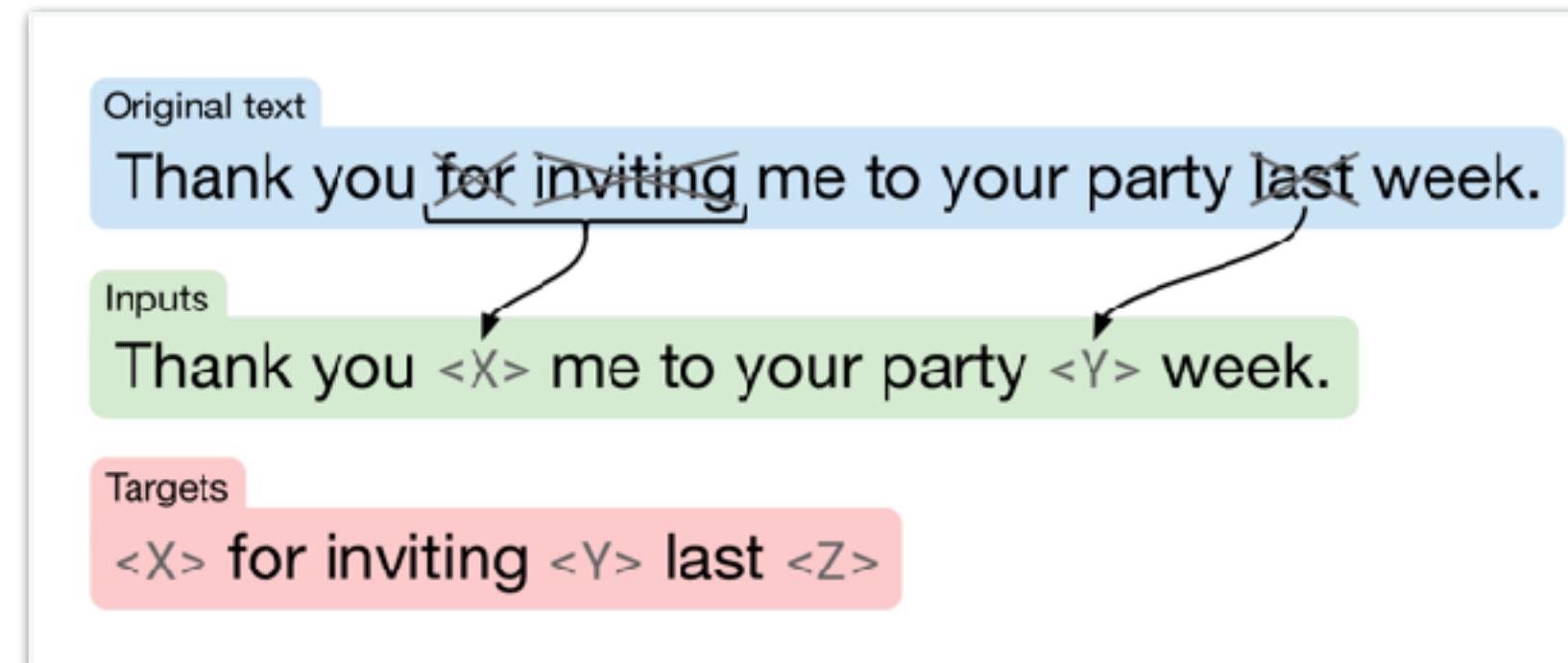
charged, used, initially, even,
future, became, the, yellow,
reported, that, luxurious,
horse-drawn, were that,
internal, conspicuous, cab

Masked token prediction



T5 (Encoder-decoder)

Raffel et al., 2019



Denoising span-mask prediction



GPT - 4

Decoder-only

Text: Second Law of Robotics: A robot must obey the orders given it by human beings

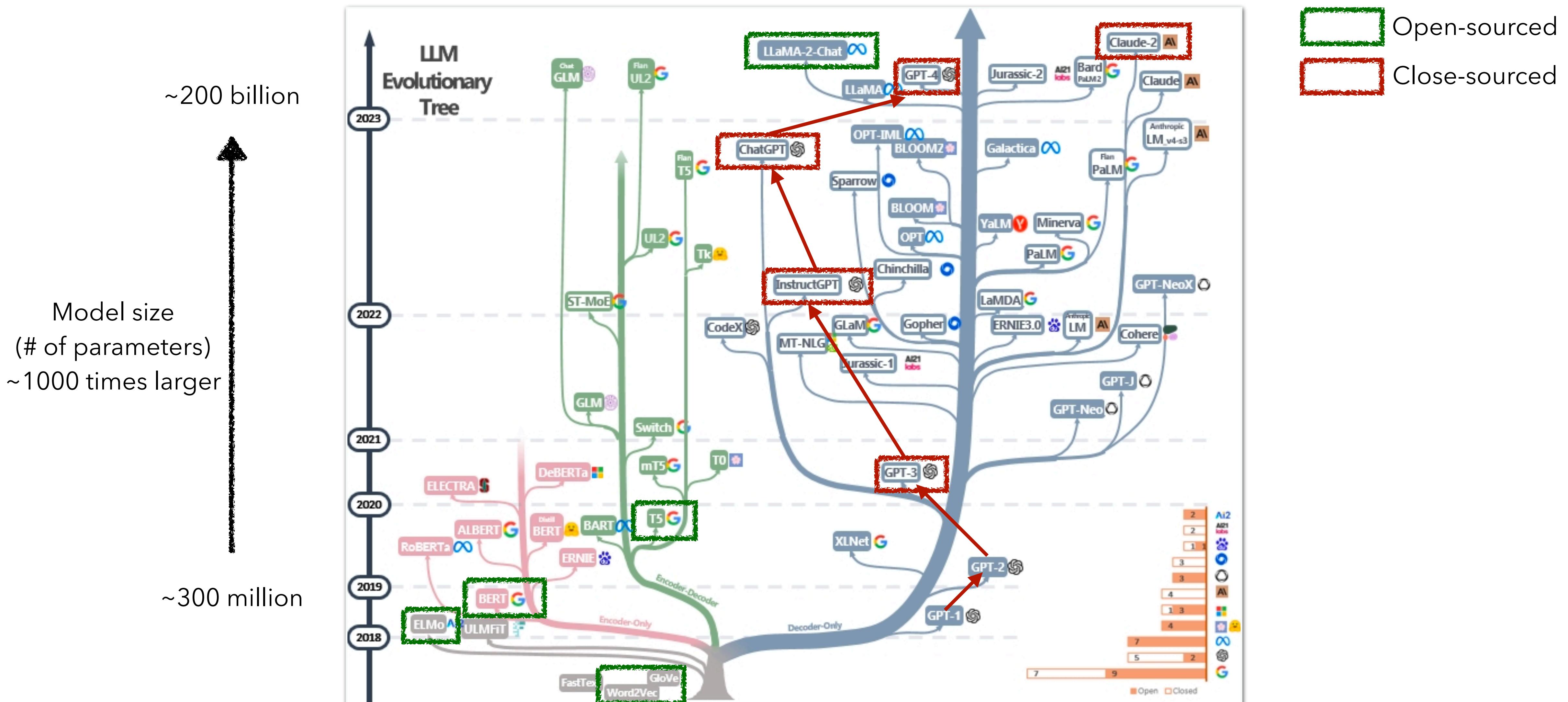
Example #	Input (features)	Correct output (labels)
1	Second law of robotics :	a
2	Second law of robotics : a	robot
3	Second law of robotics : a robot	must
...		



Generated training examples

Next token prediction

Evolution tree of pretrained LMs



From GPT I to GPT-2 to GPT-3

- All **decoder-only Transformer-based language models**
- Model size ↑, training corpora ↑

GPT-2

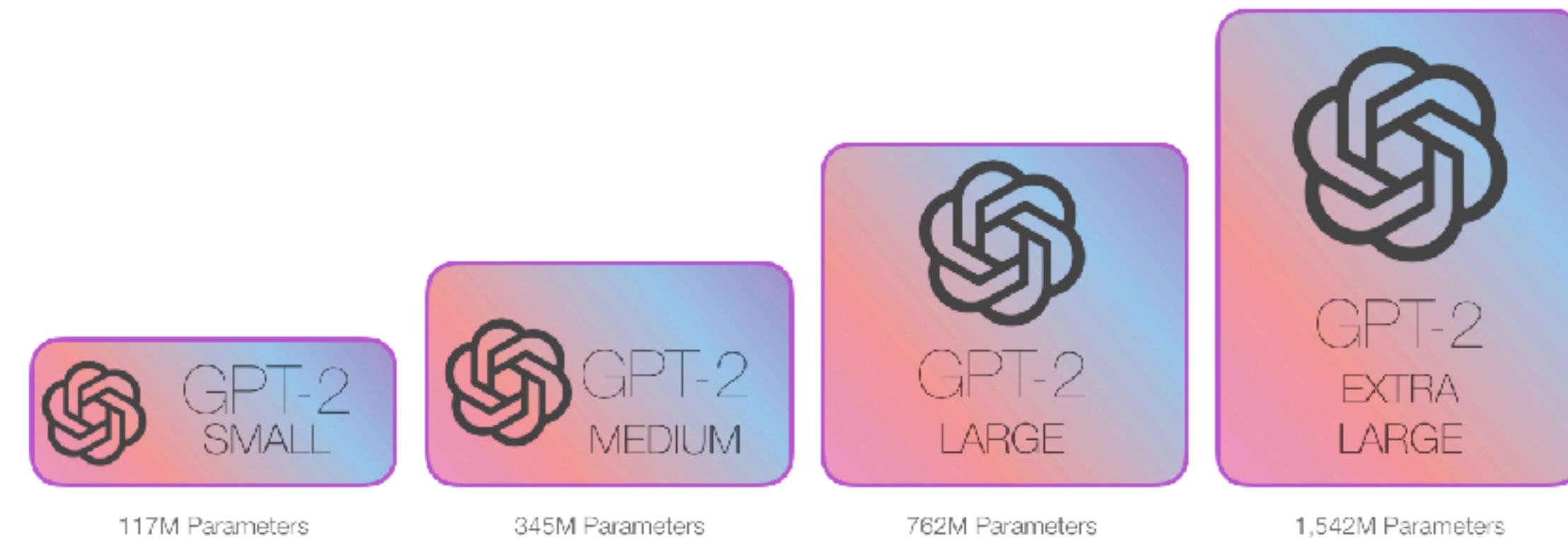
Better language models and their implications

February 14, 2019

[Read paper ↗](#)

[View code ↗](#)

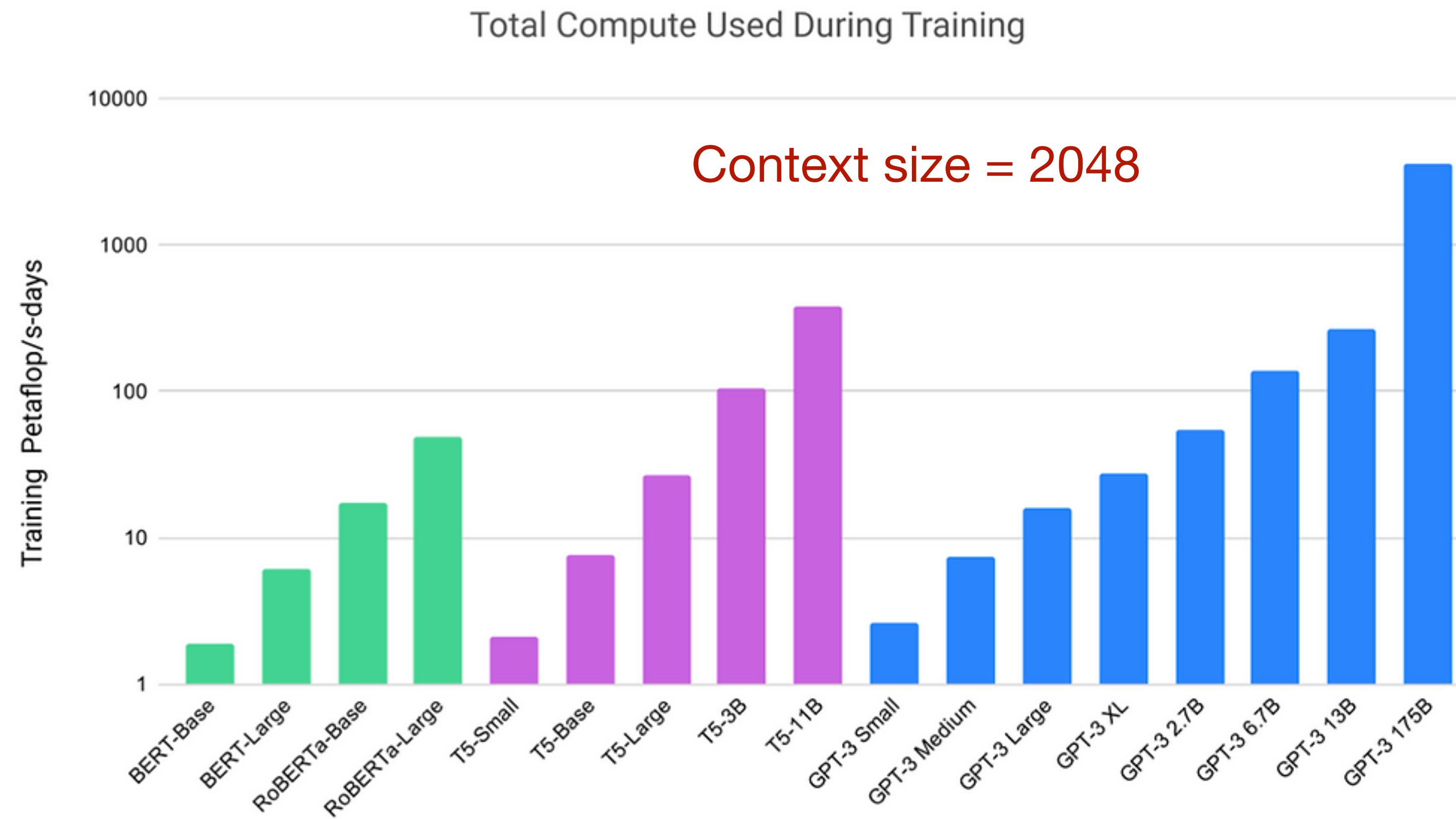
Context size = 1024



.. trained on **40Gb** of Internet text ..

GPT-3: language models are few-shot learners

- GPT-2 → GPT-3: 1.5B → **175B** (# of parameters), ~14B → **300B** (# of tokens)



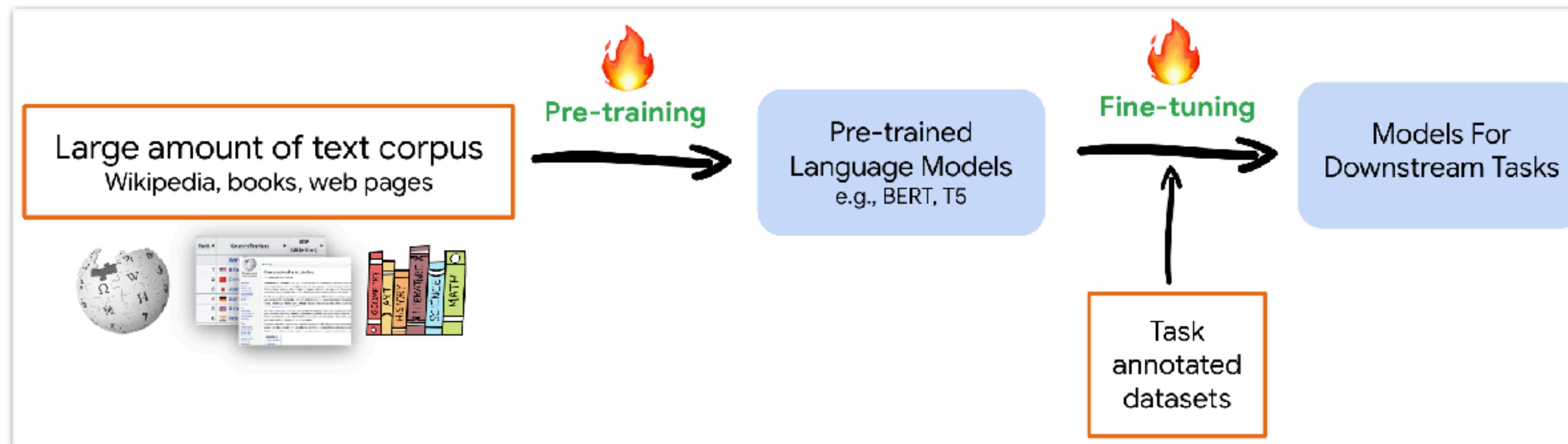
Training computation is measured using floating-point operations or “FLOP”.

One FLOP represents a single arithmetic operation involving floating-point numbers, such as addition, subtraction, multiplication, or division.

Before GPT3: Modern learning paradigm

- **Pre-training + supervised training/fine-tuning**

- First train Transformer using a lot of general text using unsupervised learning. This is called **pretraining**.
- Then train the pretrained Transformer for a specific task using supervised learning. This is called **finetuning**.



Paradigm shift since GPT-3

- Before GPT-3, **Pre-training + supervised training/fine-tuning** is the default way of doing learning in models like BERT/T5/GPT-2
 - SST-2 has 67k examples, SQuAD has 88k (passage, answer, question) triples
- Fine-tuning requires computing the gradient and applying a parameter update on every example (or every K examples in a mini-batch)
- However, this is very expensive for the 175B GPT-3 model

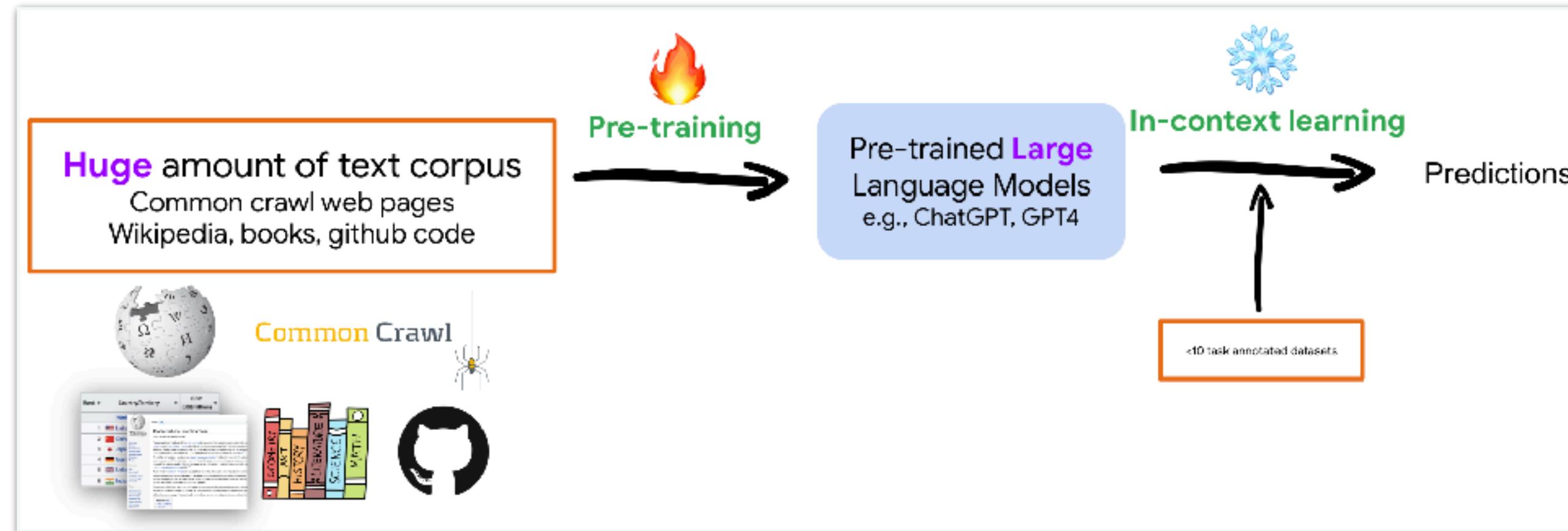
Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.



Latest learning paradigm shift since GPT-3

- Pre-training + prompting/in-context learning (no training this step)
 - First train a **large (>7~175B)** Transformer using a lot of general text using unsupervised learning. This is called **large** language model **pretraining**.
 - Then **directly use** the pretrained large Transformer (**no further finetuning/training**) for any different task given only a natural language description of the task or a few task (x, y) examples. This is called **prompting/in-context learning**.



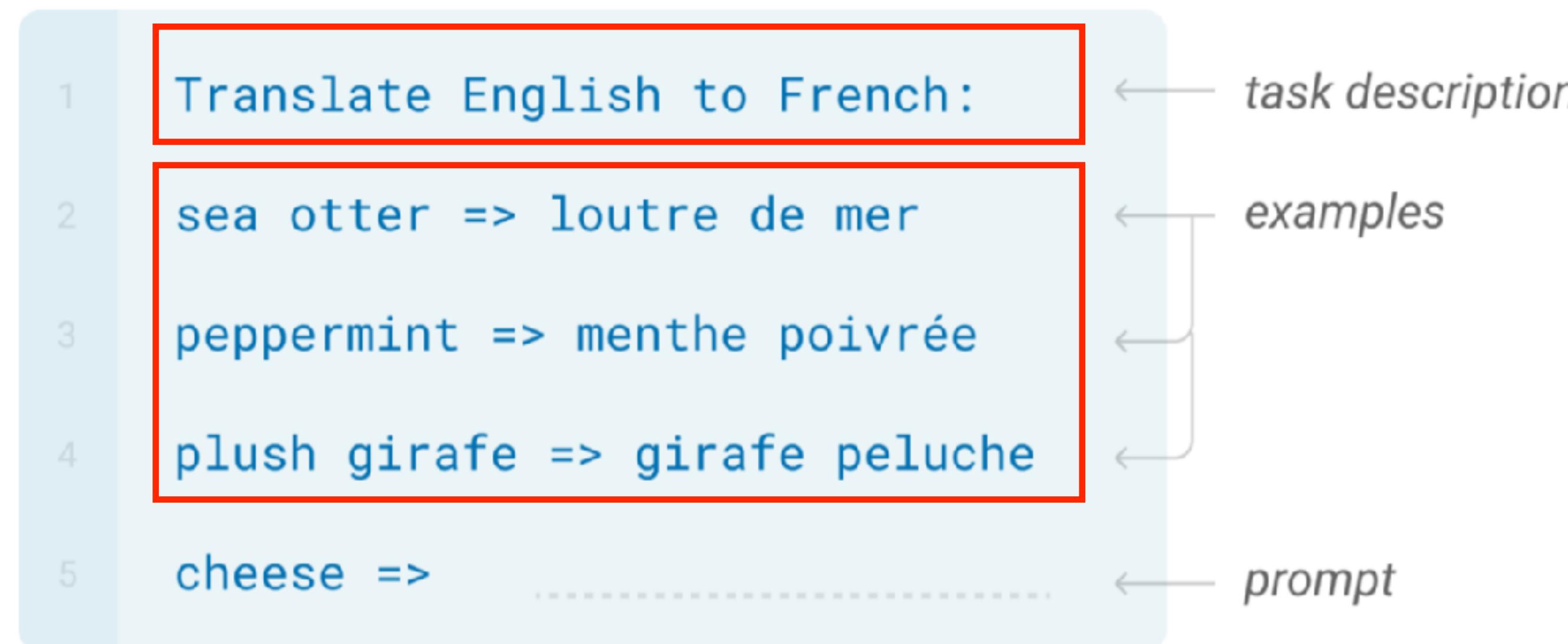
GPT-3: few-shot in-context learning

- GPT-3 proposes an alternative: **in-context learning**

Few-shot

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

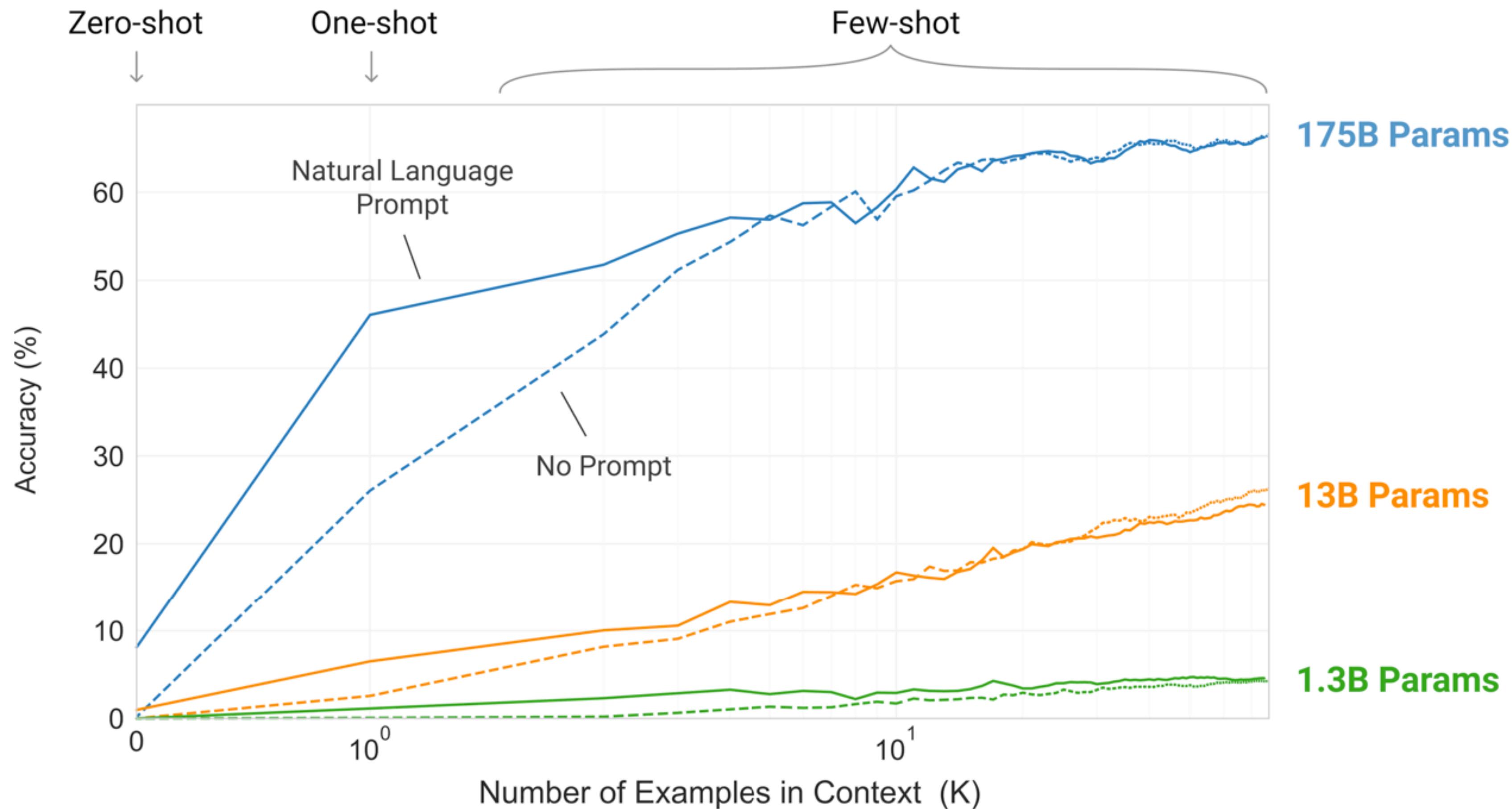
- This is just a forward pass, **no gradient update at all!**



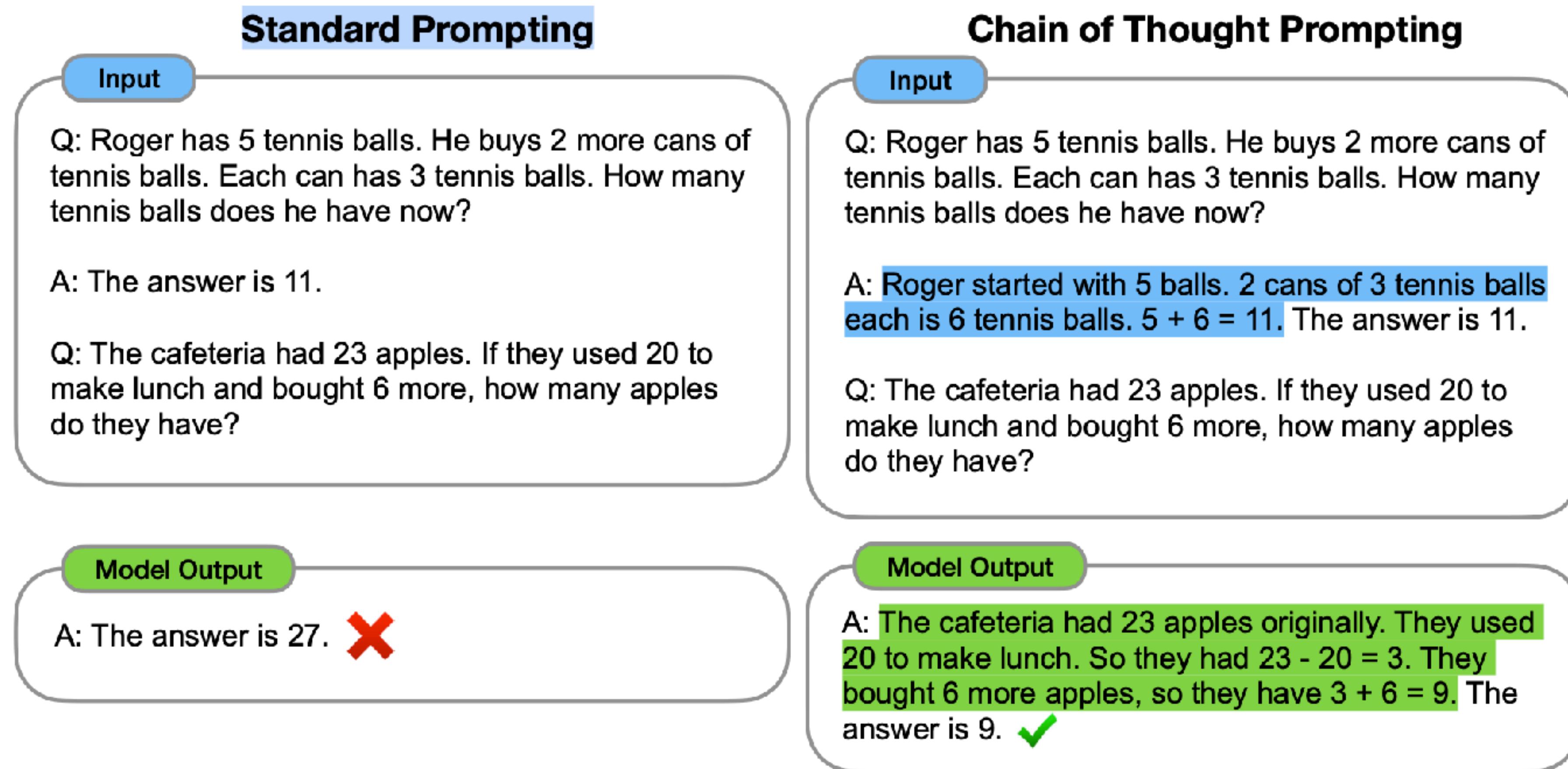
- You only need to feed a small number of examples (e.g., 32)

(On the other hand, you can't feed many examples at once too as it is bounded by context size)

GPT-3's scaling laws in performance



Chain-of-thought (CoT) prompting



Why in-context learning with LLMs?

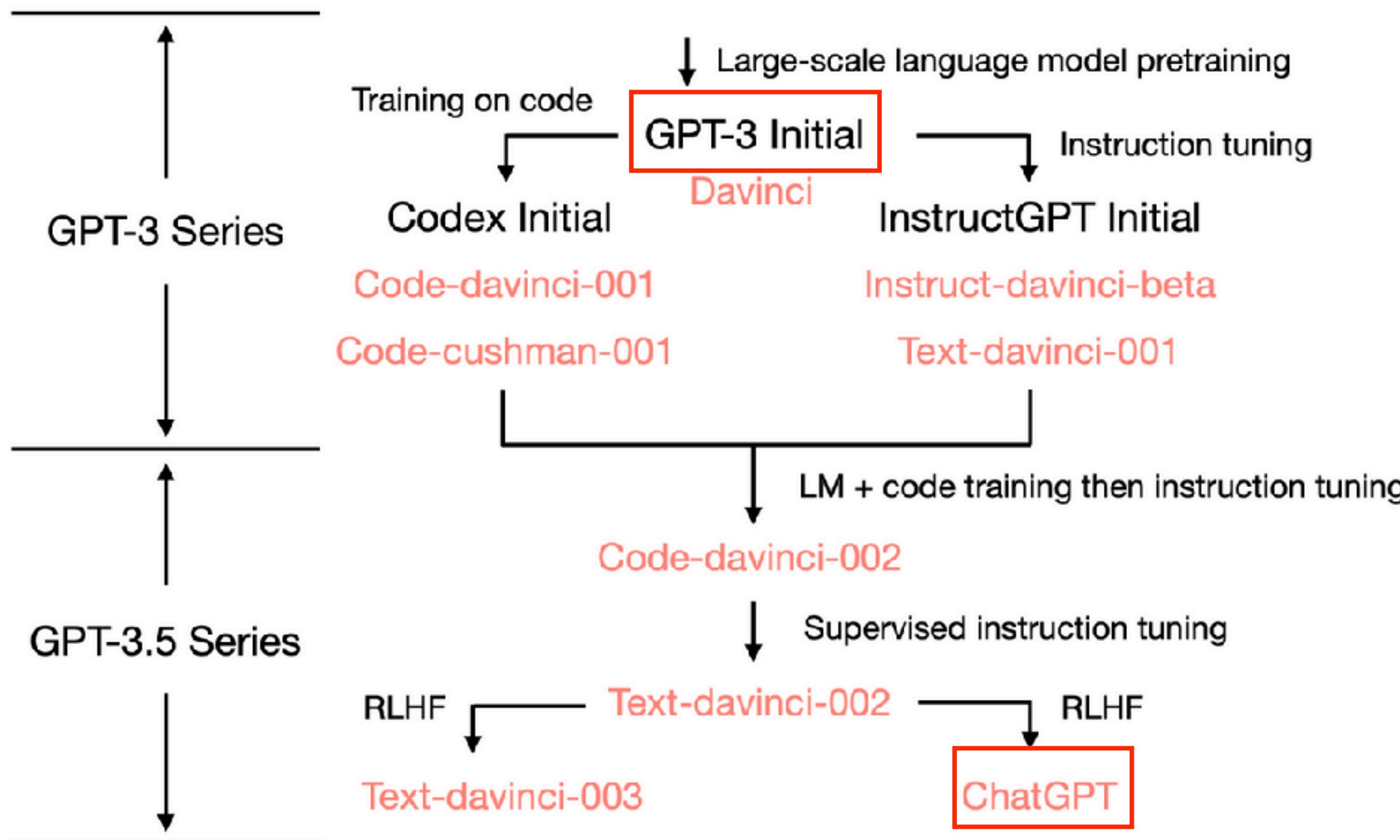
- Amazing zero/few-shot performance
 - Save a lot of annotation! 🎉
- Easy to use without training
 - Just talk to them! 👍
- One model for many NLP applications 😊
 - No need to annotate and fine-tune for different tasks

But, again, they are sensitive to prompts! Need to design a good prompt or train a good example retriever! 😂

What happened after GPT-3?

(Is model size ↑, training corpora ↑ the only way to go?)

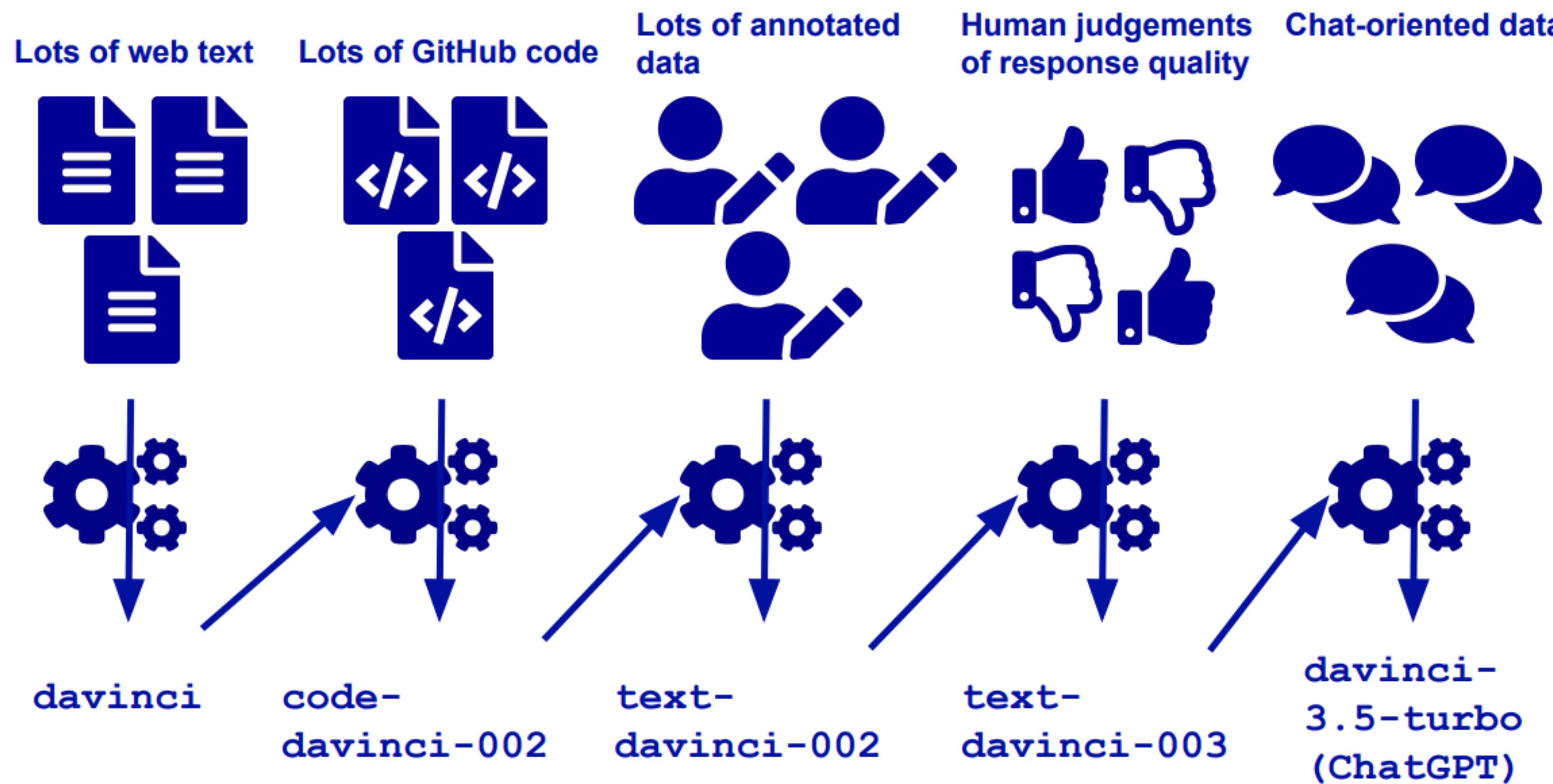
How was ChatGPT developed?



What's new?

- Training on code
- **Supervised instruction tuning**
- **RLHF = Reinforcement learning from human feedback**

How was ChatGPT developed?



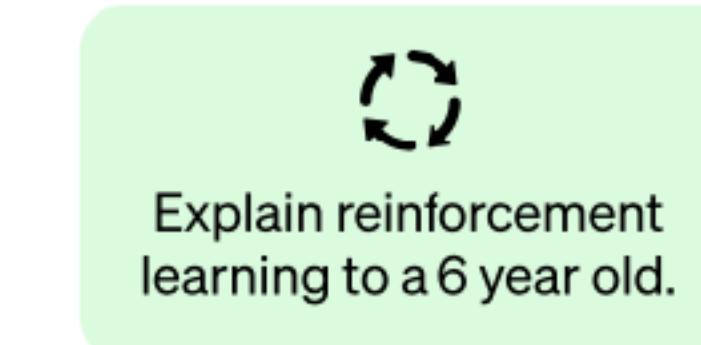
(Slide credit: Graham Neubig)

InstructGPT: Supervised instruction tuning + RLHF

Step 1

**Collect demonstration data
and train a supervised policy.**

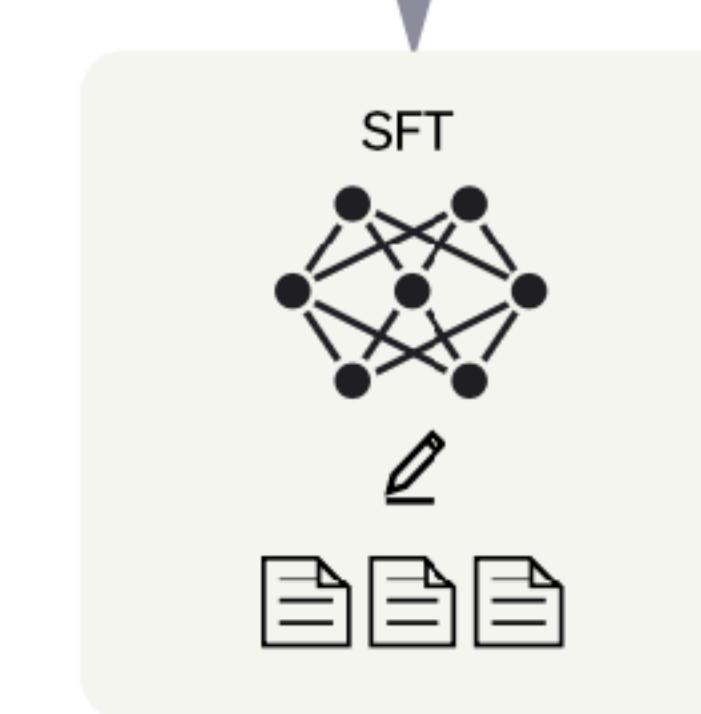
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



This data is used to fine-tune GPT-3.5 with supervised learning.



Supervised instruction tuning

Use-case	Prompt
Brainstorming	List five ideas for how to regain enthusiasm for my career
Generation	Write a short story where a bear goes to the beach, makes friends with a seal, and then returns home.
Rewrite	This is the summary of a Broadway play: """ {summary} """ This is the outline of the commercial for that play: """

Use-case	(%)
Generation	45.6%
Open QA	12.4%
Brainstorming	11.2%
Chat	8.4%
Rewrite	6.6%
Summarization	4.2%
Classification	3.5%
Other	3.5%
Closed QA	2.6%
Extract	1.9%

Number of Prompts		
SFT Data		
split	source	size
train	labeler	11,295
train	customer	1,430
valid	labeler	1,550
valid	customer	103

SFT data: only ~13k (not public)

InstructGPT: Supervised instruction tuning + RLHF

Step 2

Collect comparison data and train a reward model.

A prompt and several model outputs are sampled.

Explain reinforcement learning to a 6 year old.

A
In reinforcement learning, the agent is...

B
Explain rewards...

C
In machine learning...

D
We give treats and punishments to teach...

A labeler ranks the outputs from best to worst.

D > C > A > B

This data is used to train our reward model.

RM
D > C > A > B

InstructGPT: Supervised instruction tuning + RLHF

Step 3

Optimize a policy against the reward model using the PPO reinforcement learning algorithm.

A new prompt is sampled from the dataset.



The PPO model is initialized from the supervised policy.

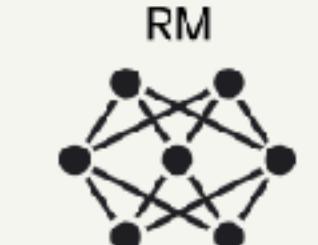


The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

Once upon a time...



r_k

ChatGPT = InstructGPT + dialogue data

Introducing ChatGPT

We've trained a model called ChatGPT which interacts in a conversational way. The dialogue format makes it possible for ChatGPT to answer followup questions, admit its mistakes, challenge incorrect premises, and reject inappropriate requests.

"We trained this model using Reinforcement Learning from Human Feedback (RLHF), **using the same methods as InstructGPT**, but with slight differences in the data collection setup. We trained an initial model using supervised fine-tuning: human AI trainers provided conversations in which they played both sides—the user and an AI assistant. We gave the trainers access to model-written suggestions to help them compose their responses. **We mixed this new dialogue dataset with the InstructGPT dataset**, which we transformed into a dialogue format."

Human feedback data is the key!

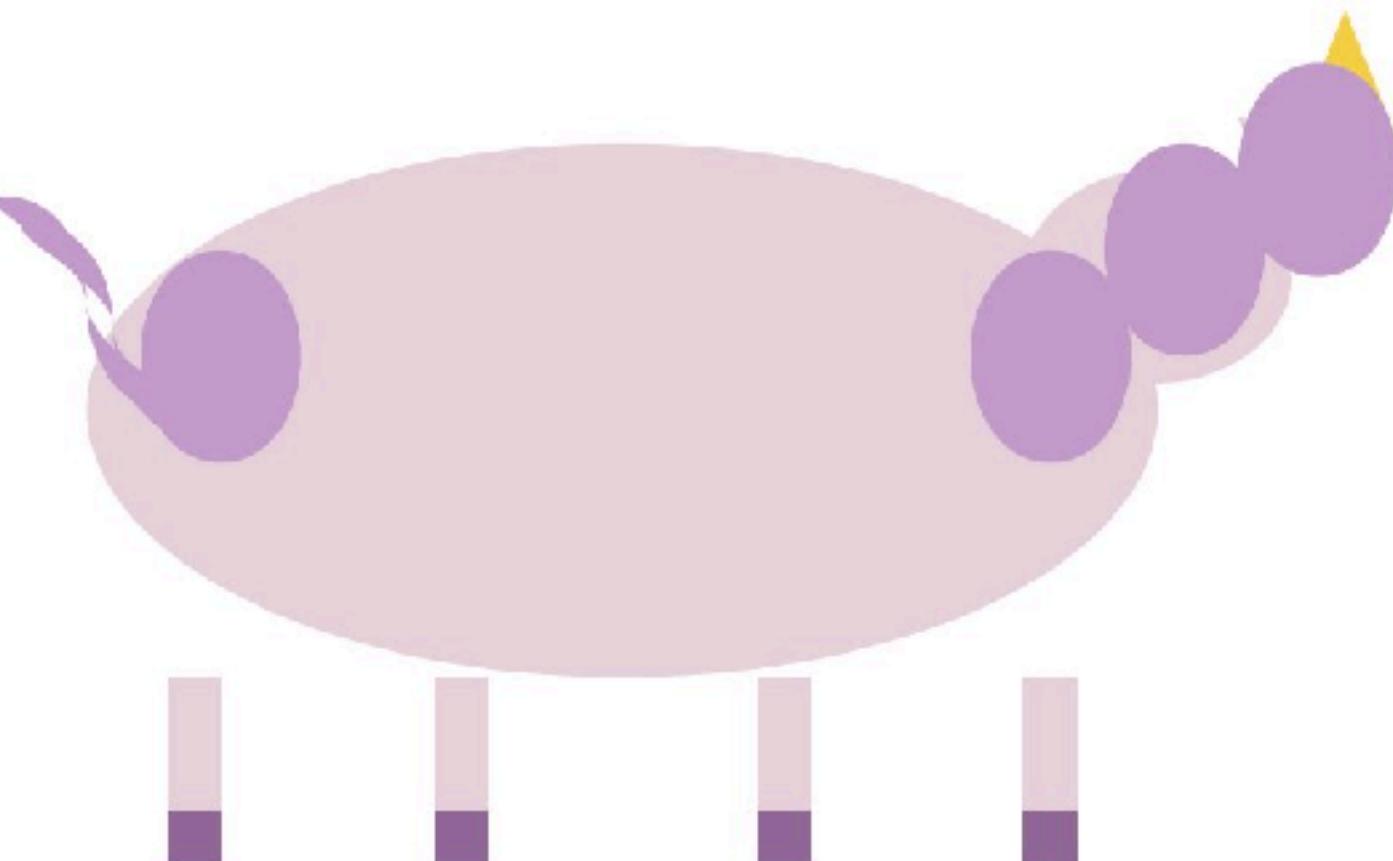
<https://openai.com/blog/chatgpt>

GPT-4

- GPT-4: a **multi-modal** model capable of processing image and text inputs and producing text outputs.
- Model size and training details unknown
- Can process up to **32k** context size

Prompt: Draw a unicorn in TiKZ.

GPT-4: [Produces L^AT_EX compiling to following picture.]



GPT-4

GPT-4 visual input example, Moar Layers:

User Can you explain why this is funny. Think about it step-by-step.



GPT-4

The comic is satirizing the difference in approaches to improving model performance between statistical learning and neural networks.

In statistical learning, the character is shown to be concerned with overfitting and suggests a series of complex and technical solutions, such as minimizing structural risk, reworking the loss function, and using a soft margin.

In contrast, the neural networks character simply suggests adding more layers to the model. This is often seen as a common solution to improving

GPT-4 visual input example, Extreme Ironing:

User What is unusual about this image?



Source: <https://www.barnorama.com/wp-content/uploads/2016/12/03-Confusing-Pictures.jpg>

GPT-4

The unusual thing about this image is that a man is ironing clothes on an ironing board attached to the roof of a moving taxi.