COMP 3361 Natural Language Processing

Lecture 13: LLM prompting, in-context learning, scaling laws, emergent capacities (cont'd)

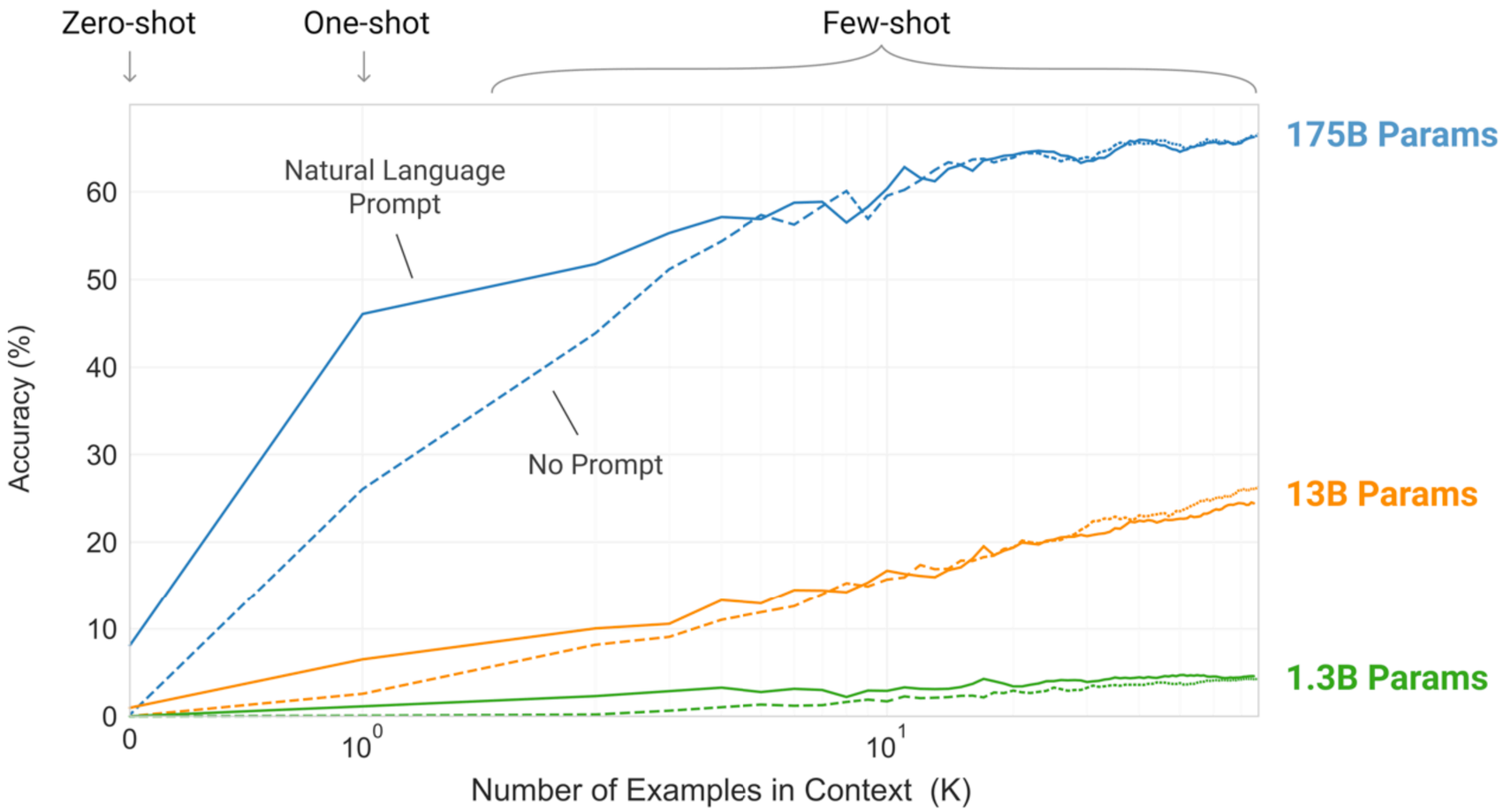
Spring 2024

Many materials from COS484@Princeton and CSE447@UW (Taylor Sorensen) with special thanks!

Announcements

- As mentioned, #assignment-2 due date is extended to Mar 22, next Friday
  - #course-project will be released on Mar 22
- TA will deliver a coding tutorial on Transformers and #assignment-2 next Tuesday (Mar 19).
  - Come to the class if you have problems with transforming what we taught in class into code implementation!
- Final exam plan
- Course project plan

GPT-3's scaling laws in performance



(Brown et al., 2020): Language Models are Few-Shot Learners

Chain-of-thought (CoT) prompting

Standard Prompting

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: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ✗

Chain of Thought Prompting

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 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 - 20 = 3. They bought 6 more apples, so they have 3 + 6 = 9. The answer is 9. ✓

(Wei et al., 2022): Chain-of-Thought Prompting Elicits Reasoning in Large Language Models

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! 😂

Okay, so bigger is better? Can you be more specific?

Scaling Laws

Scaling Laws (Kaplan et al., 2020)

- Kaplan et al., 2020 (OpenAI) explore how performance scales w.r.t. several parameters
- Vary:
  - Scale: N - # Model Params, D - Dataset size (tokens)
  - Other hyperparameters: Hidden layer sizes, context length, batch size
- Goal: Can we reliably predict test loss L based on training scale (parameters and dataset size)?

Scaling Laws (Kaplan et al., 2020)

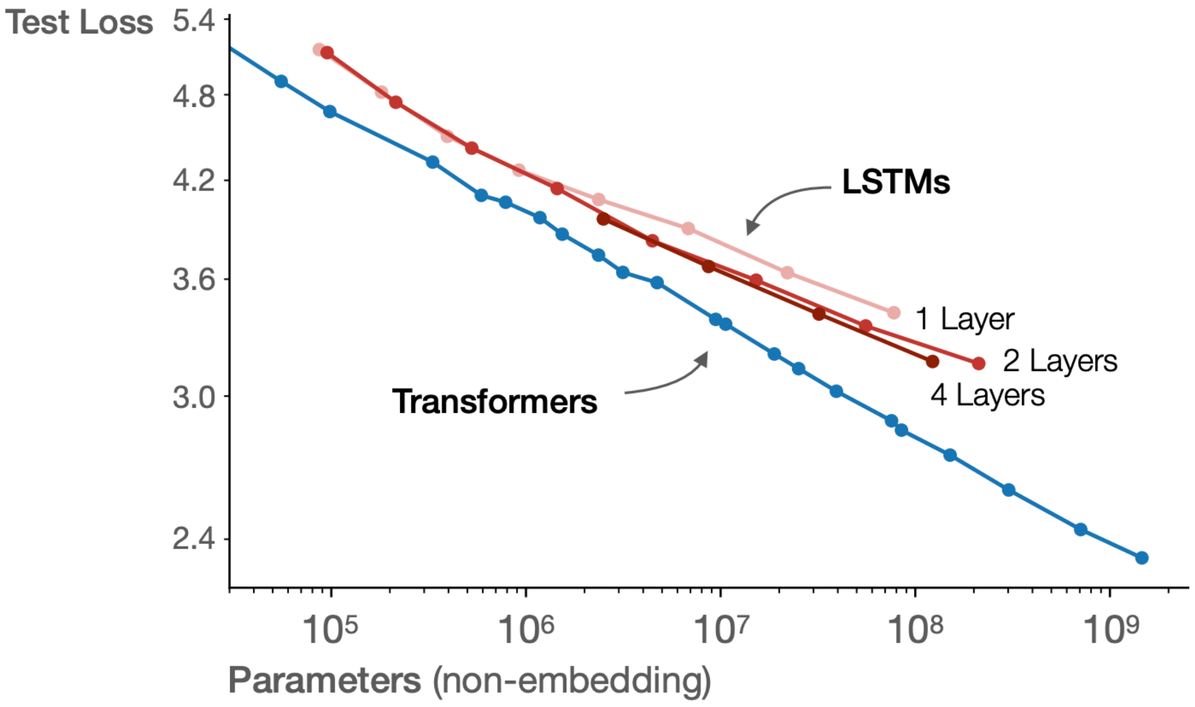
- Result: Test loss L very closely follows a *power law:*
  - Given constant dataset size D,

$$L(N) \approx \left(\frac{N_c}{N}\right)^{\alpha N}$$

  - Given constant model size N,

$$L(D) \approx \left(\frac{D_c}{D}\right)^{\alpha D}$$

To linearly decrease test loss L, you need to exponentially increase dataset size D or model size N



Scaling Laws (Kaplan et al., 2020)

- Result: Test loss L very closely follows a *power law:*
  - Given constant dataset size D,

$$L(N) \approx \left(\frac{N_c}{N}\right)^{\alpha N}$$

  - Given constant model size N,

$$L(D) \approx \left(\frac{D_c}{D}\right)^{\alpha D}$$

- Bringing it together:

$$L(N, D) \approx \left[\left(\frac{N_c}{N}\right)^{\frac{\alpha N}{\alpha D}} + \frac{D_c}{D}\right]^{\alpha D}$$

Parameter	α_N	α_D	N_c	D_c
Value	0.076	0.103	6.4×10^{13}	1.8×10^{13}

← Empirical estimates of parameters from experiments

Table 2 Fits to $L(N, D)$

LLaMA (Touvron et al., 2023)

- OpenAI/Deepmind only looked at the optimal size given a fixed *training* compute budget

$$\underset{N,D \text{ s.t. } \mathrm{FLOPs}(N,D)=C}{\operatorname{argmin}} L(N, D)$$

- What if you care more about *inference* time compute cost?
- Smaller model => Smaller inference cost
- To get best small model, should just train a small model on as much data as possible (beyond “Chinchilla-optimal”)
- “Overtrained” LLaMA-13B outperformed GPT-3 on many benchmarks

Recently

- A lot of recent progress has been made from training bigger models on more data: LLaMA 2, GPT-4, Gemini, Mistral, etc.
  - Note: quality matters too! Need more *high-quality data,* low-quality data does not improve performance
- Limits of scale:
  - Limits on data: Modern LLMs are trained on basically the *entire internet* - we can't find 10 new internets out of nowhere
  - Limits on compute: Big tech companies can't continue to 10x their model sizes for much longer

TECHNOLOGY | ARTIFICIAL INTELLIGENCE

Sam Altman Seeks Trillions of Dollars to Reshape Business of Chips and AI

OpenAI chief pursues investors including the U.A.E. for a project possibly requiring up to $7 trillion

By Keach Hagey Follow and Asa Fitch Follow

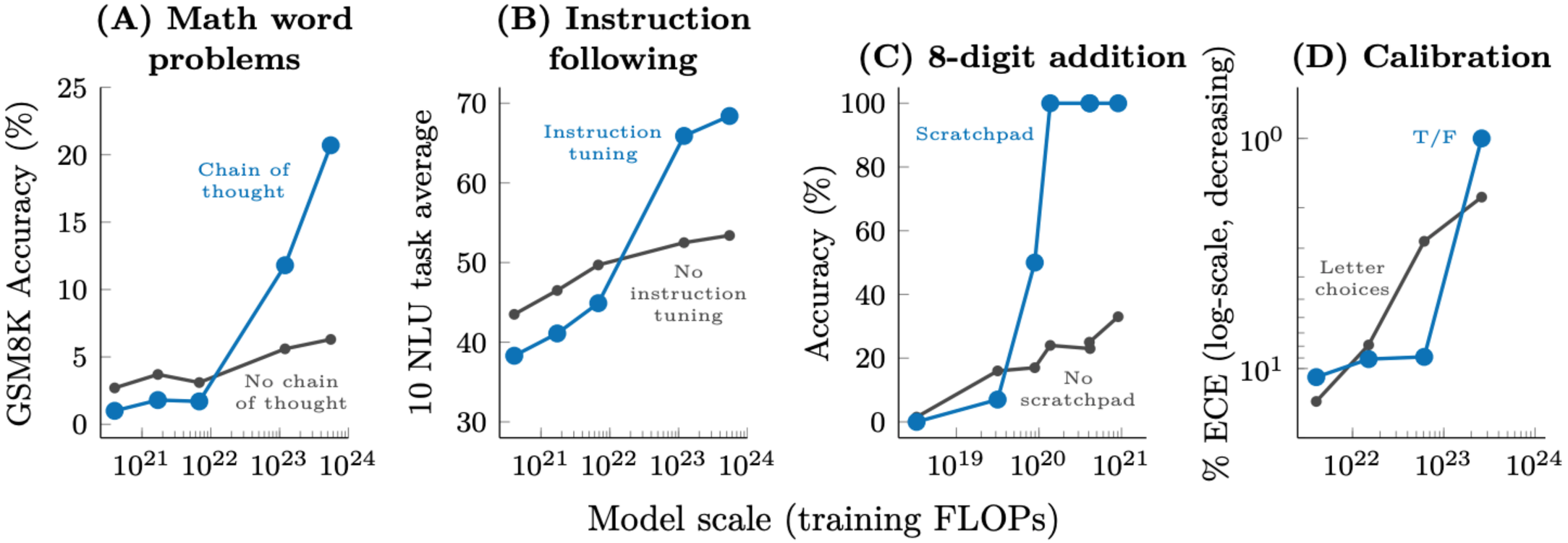
Feb. 8, 2024 9:00 pm ET

(For context: $7T is more than GDP of all countries except US and China! Japan: $4.2T, Germany: $4T, …)

But that won't stop Sam Altman from trying!

Emergent capabilities of LLMs?

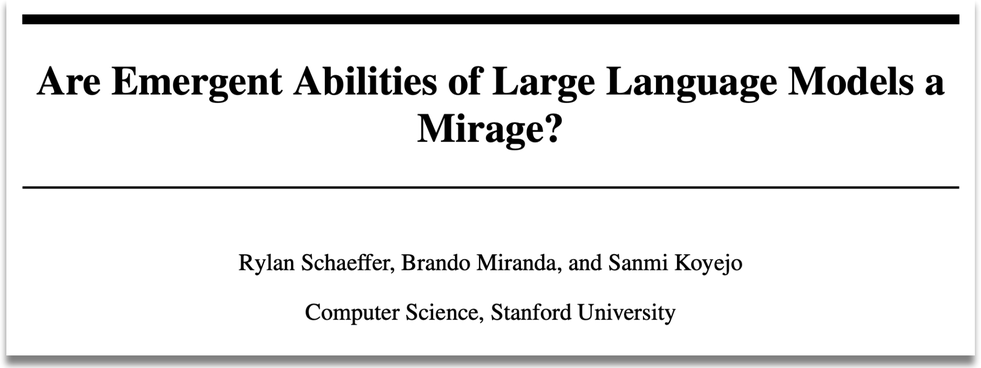
Emergent properties of LLMs



(Wei et al., 2022) Emergent Abilities of Large Language Models

Emergent capabilities a mirage?

- (Schaeffer et al., 2023) take issue with the characterization of "emergent capabilities"
- Most metrics used in (Wei et al., 2022) were "hard" metrics which don't give partial credit like accuracy



Are Emergent Abilities of Large Language Models a Mirage?

Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo

Computer Science, Stanford University

Hard Accuracy:

A) 123 + 456 = 579 ✅

B) 123 + 456 = 578 ❌

C) 123 + 456 = 42 ❌

In (Wei et al., 2022), B and C are both wrong, even though B is much closer to correct than C

Emergent capabilities a mirage?

- (Schaeffer et al., 2023) measure soft metrics (e.g., how many digits are correct, probability of the right answer) for “emergent abilities”
- Find much more predictable scaling
- Different metric choices lead to different appearances of “emergent” or not emergent
- “Emergent abilities” are a mirage(?)

Hard Accuracy:

A) 123 + 456 = 579 ✅

B) 123 + 456 = 578 ❌

C) 123 + 456 = 42 ❌

Soft Accuracy (# correct digits):

A) 123 + 456 = 579 3/3 ✅

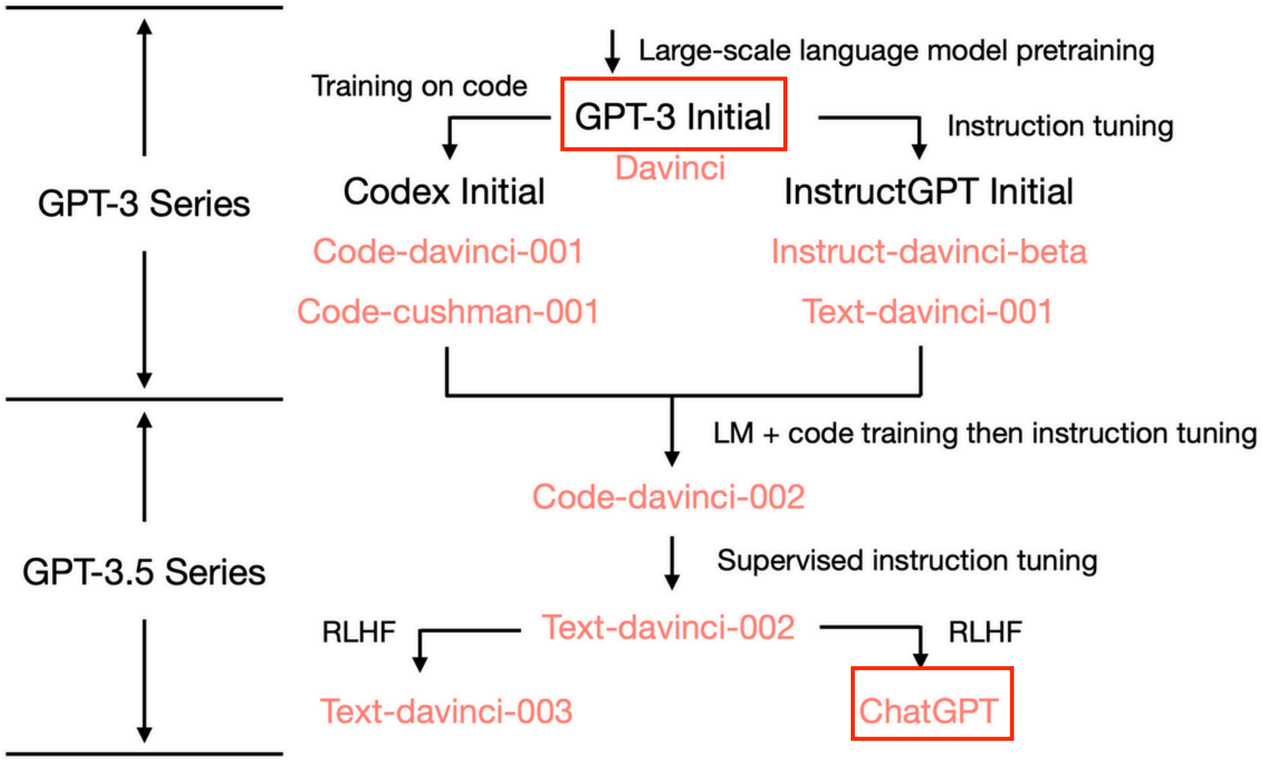
B) 123 + 456 = 578 2/3 🤷‍♀️

C) 123 + 456 = 42 0/3 ❌

What happened after GPT-3?

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

How was ChatGPT developed?

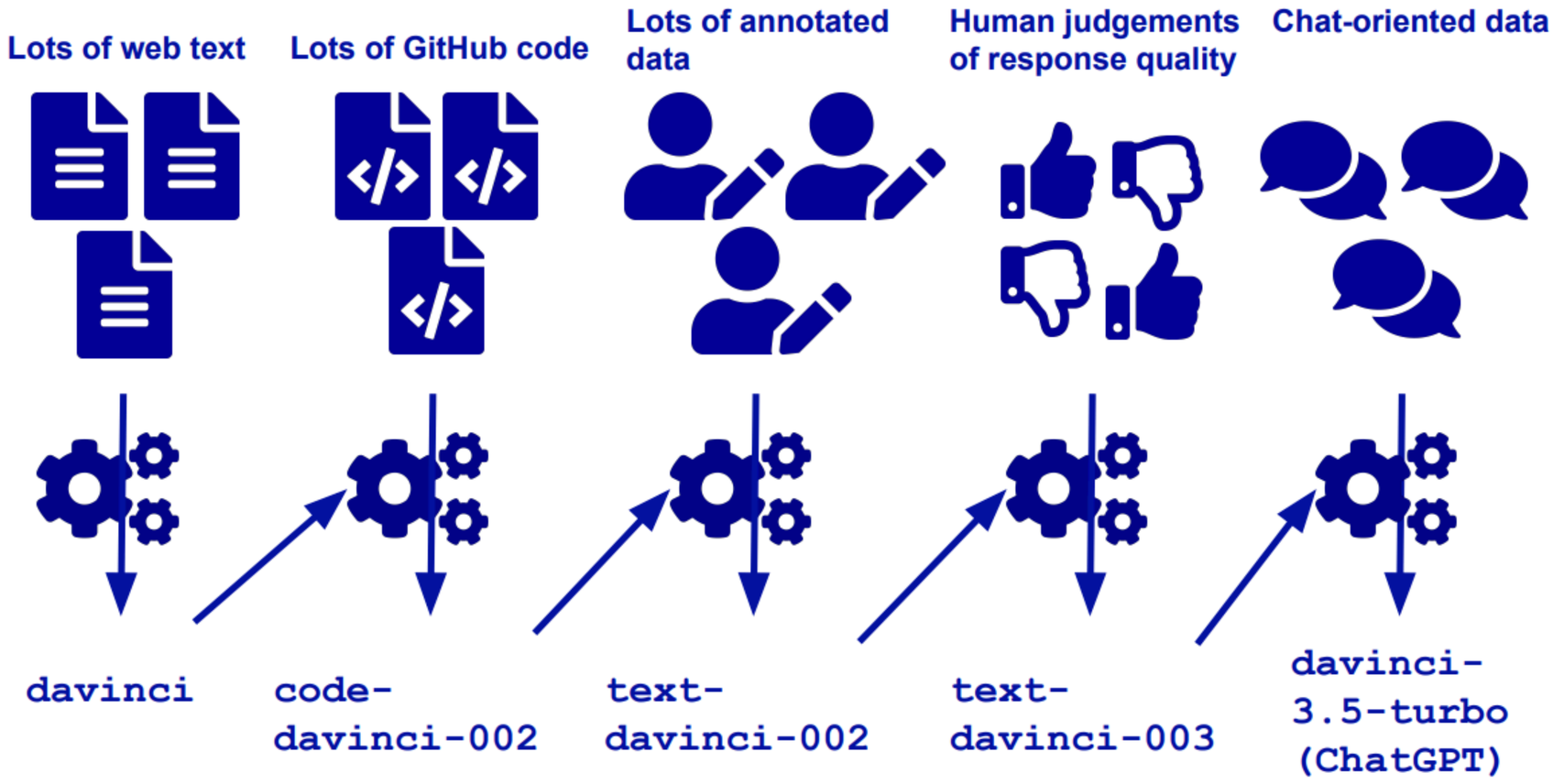


What's new?

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

https://yaofu.notion.site/How-does-GPT-Obtain-its-Ability-Tracing-Emergent-Abilities-of-Language-Models-to-their-Sources-b9a57ac0fcf74f30a1ab9e3e36fa1dc1

How was ChatGPT developed?



(Slide credit: Graham Neubig)

InstructGPT: Supervised instruction tuning + RLHF

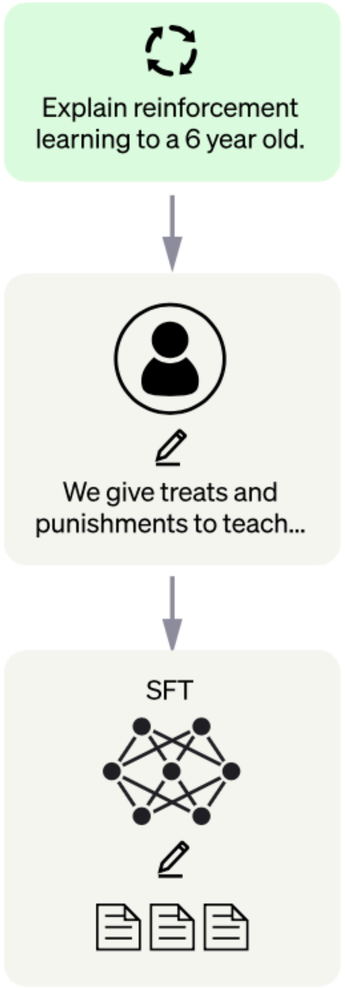
Step 1

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.



(Ouyang et al., 2022): Training language models to follow instructions with human feedback

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

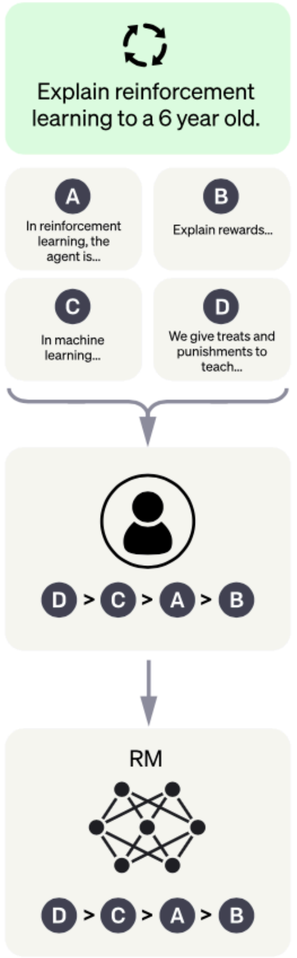
SFT data: only ~13k (not public)

InstructGPT: Supervised instruction tuning + RLHF

Step 2

Collect comparison data and train a reward model.

A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.

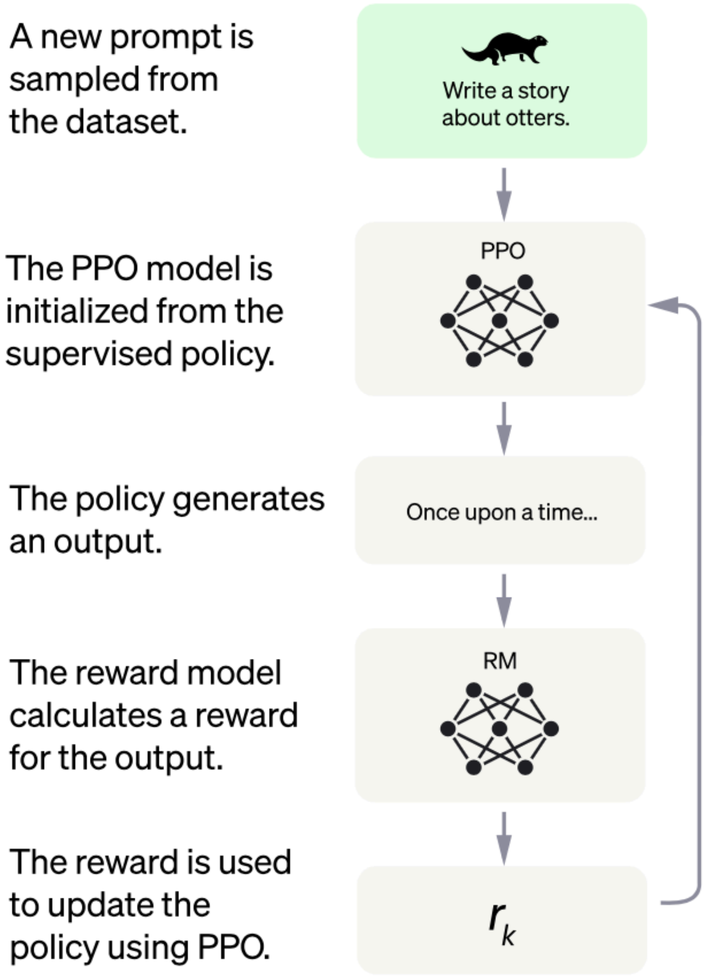
This data is used to train our reward model.

(Ouyang et al., 2022): Training language models to follow instructions with human feedback

InstructGPT: Supervised instruction tuning + RLHF

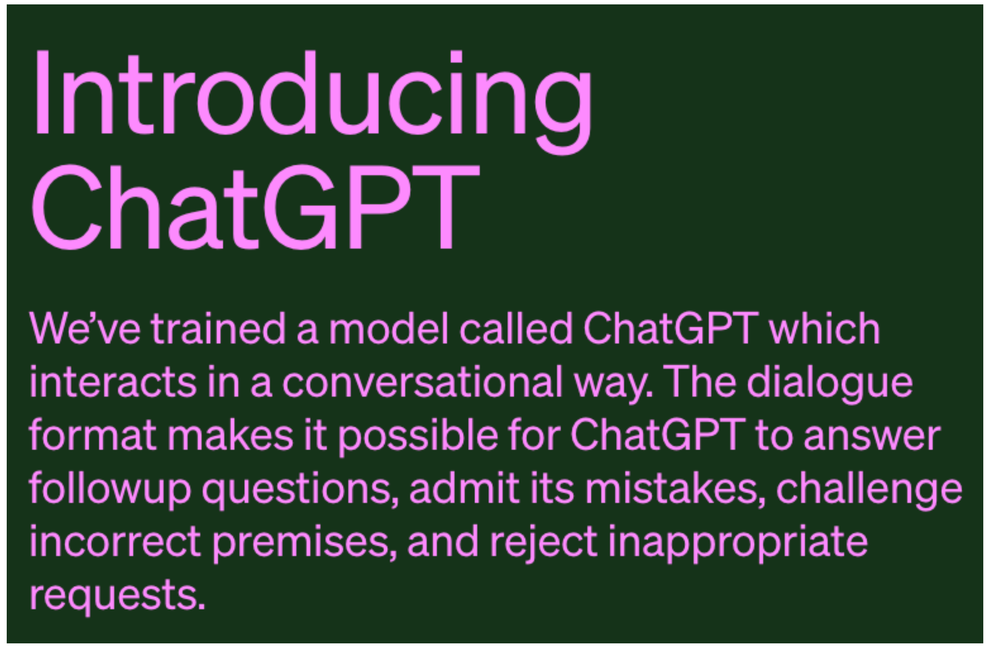
Step 3

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



(Ouyang et al., 2022): Training language models to follow instructions with human feedback

ChatGPT = InstructGPT + dialogue data



"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

Recent models are getting smaller?

RESEARCH

Introducing LLaMA: A foundational, 65-billion-parameter large language model

February 24, 2023

- **Smaller models** trained on **1.4T**, high-quality & publicly available data
- The models (LLaMA 2) are public
- “LLaMA-13B outperforms GPT-3 (175B) on most benchmarks, and LLaMA-65B is competitive with the best models, Chinchilla-70B and PaLM-540B”

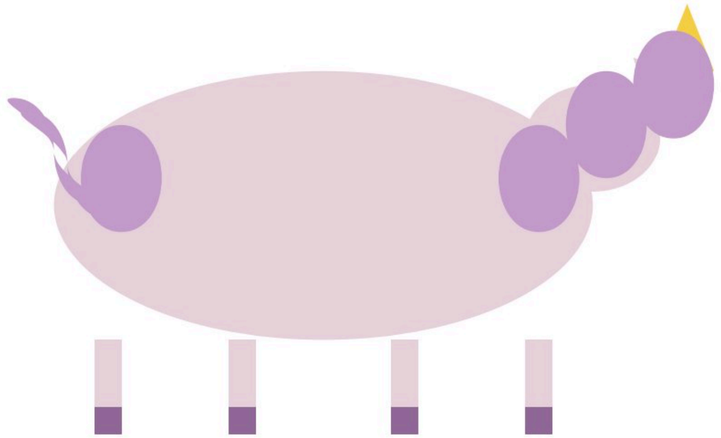
(Touvron et al., 2023): LLaMA: Open and Efficient Foundation Language Models

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.

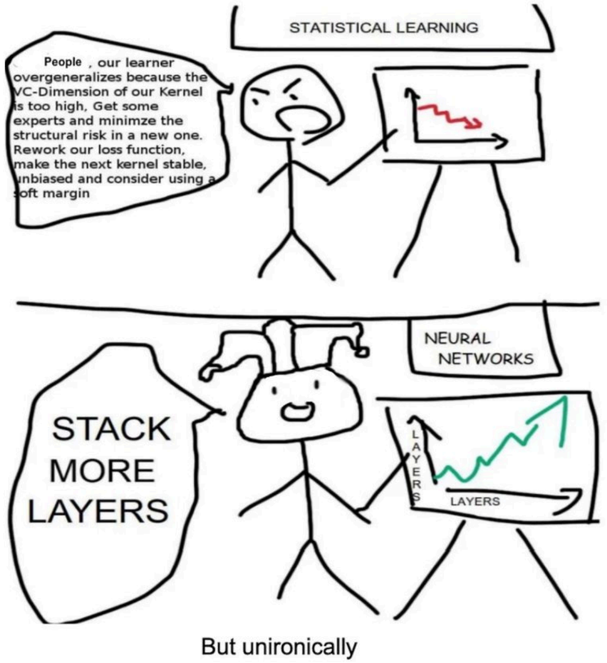
GPT-4: [Produces LaTeX compiling to following picture.]



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.