Fall 2024 COS597R:
Deep Dive into Large Language Models

Danqi Chen, Sanjeev Arora

PRINCETON UNIVERSITY

Lecture 7: Instruction Tuning

https://princeton-cos597r.github.io/

Required reading

Scaling Instruction-Finetuned Language Models

Hyung Won Chung* Le Hou* Shayne Longpre* Barret Zoph† Yi Tay†
William Fedus† Yunxuan Li Xuezhi Wang Mostafa Dehghani Siddhartha Brahma
Albert Webson Shixiang Shane Gu Zhuyun Dai Mirac Suzgun Xinyun Chen
Aakanksha Chowdhery Alex Castro-Ros Marie Pellat Kevin Robinson
Dasha Valter Sharan Narang Gaurav Mishra Adams Yu Vincent Zhao
Yanping Huang Andrew Dai Hongkun Yu Slav Petrov Ed H. Chi
Jeff Dean Jacob Devlin Adam Roberts Denny Zhou Quoc V. Le
Jason Wei*

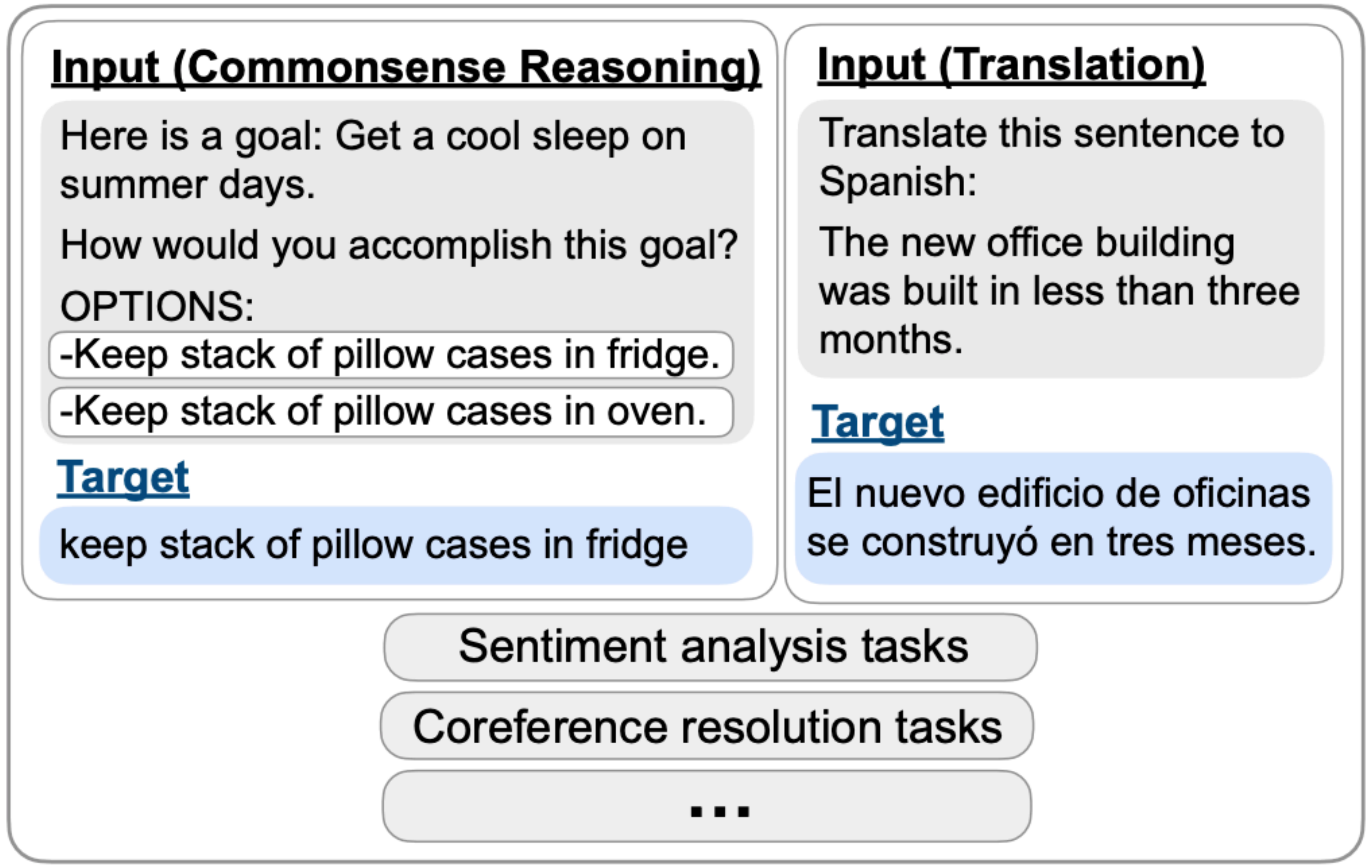
Google

What is instruction tuning?

- **Related keywords**: post-training, instruction fine-tuning, supervised fine-tuning (SFT)
- **Instruction tuning** = supervised fine-tuning on ***instruction data***

Instruction data (prompt, completion): (x, y)

$$-\sum_{i=1}^{|y|} \log P(y_i \mid y_{<i}, x)$$



- FLAN (Wei et al., 2021)

What is instruction tuning?

<|user|>
Explain the fault-tolerance of the reaction control system on the Space Shuttle.

<|assistant|>
The reaction control system (RCS) on the Space Shuttle was designed to be fault-tolerant, meaning it was able to continue functioning even if one or more of its components failed. The RCS consisted of two sets of ... </s>

<|user|>
Did the RCS have any on-orbit failures?

<|assistant|>
There were several instances where the reaction control system (RCS) on the Space Shuttle experienced failures or malfunctions during on-orbit missions. These ... </s>

$$L = -\sum_j \log p_\theta(t_j \mid t_{<j}) \times \begin{cases} 1 & \text{if } t_j \in Y \\ 0 & \text{otherwise} \end{cases}$$

- Tulu (Wang et al., 2023)

$$-\sum_{i=1}^{|y|} \log P(y_i \mid y_{<i}, x)$$

(Optional) calculate loss on **output tokens only**, or the **entire input + output** (same as from pre-training)

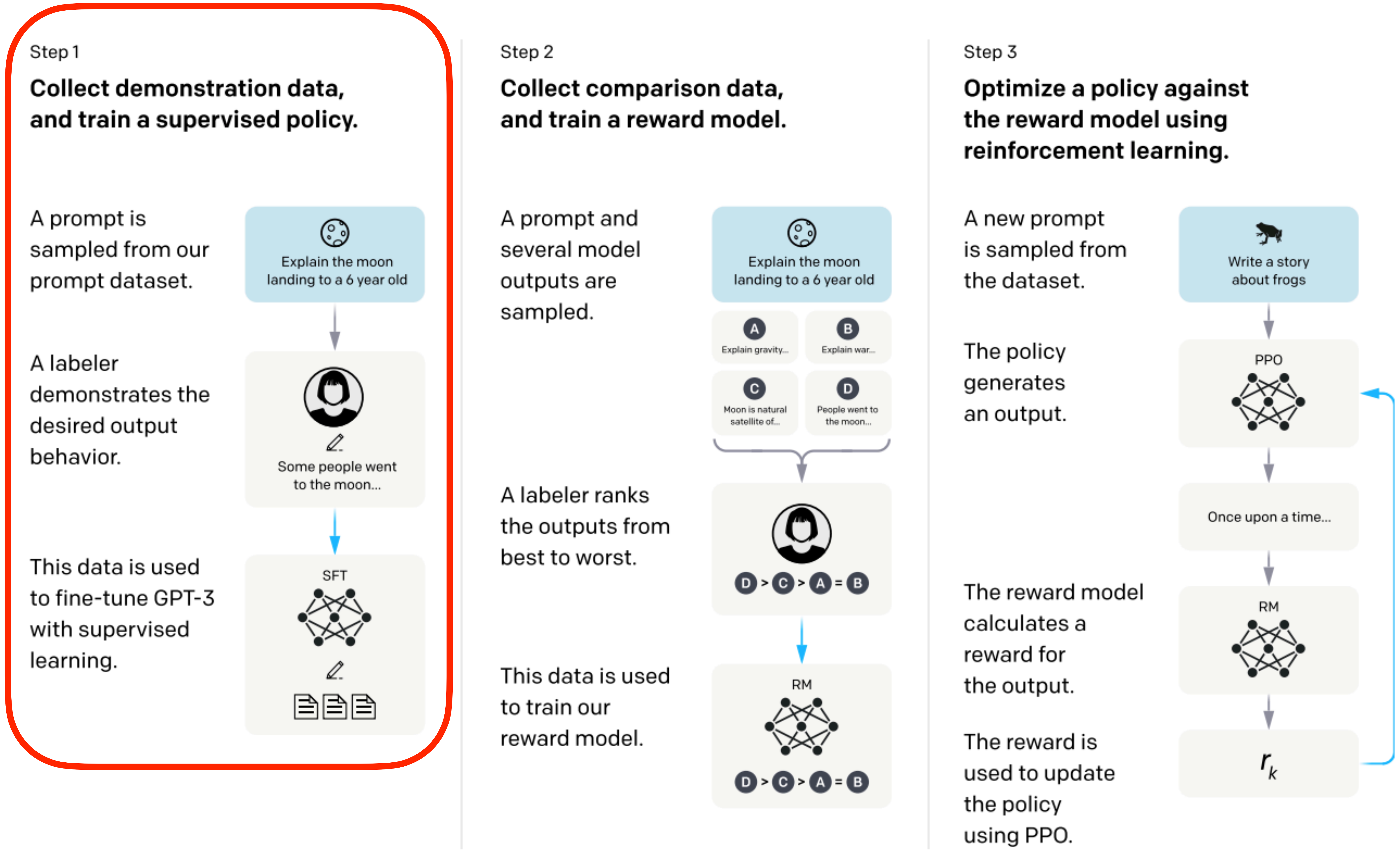
For short instruction data, we concatenate them as 16,384-token sequences. For long instruction data, we add padding tokens on the right so that models can process each long instance individually without truncation. While standard instruction tuning only calculates loss on the output tokens, we find it particularly beneficial to also calculate the language modeling loss on the long input prompts, which gives consistent improvements on downstream tasks (Section 4.3).

- Llama 2 Long (Xiong et al., 2023)

What is instruction tuning?

- **First wave (2021-2022):** instruction tuning on massive (NLP) tasks can generalize to unseen tasks
    - Cross-task generalization
    - Limited to standard tasks - easier to evaluate!
- **Second wave (2022-??):** "open-ended" instruction tuning, popularized by InstructGPT/ChatGPT
    - Anything can be a task - infinite possibilities!
    - Evaluation is hard: human evaluation, LLM as judge..

What is instruction tuning?



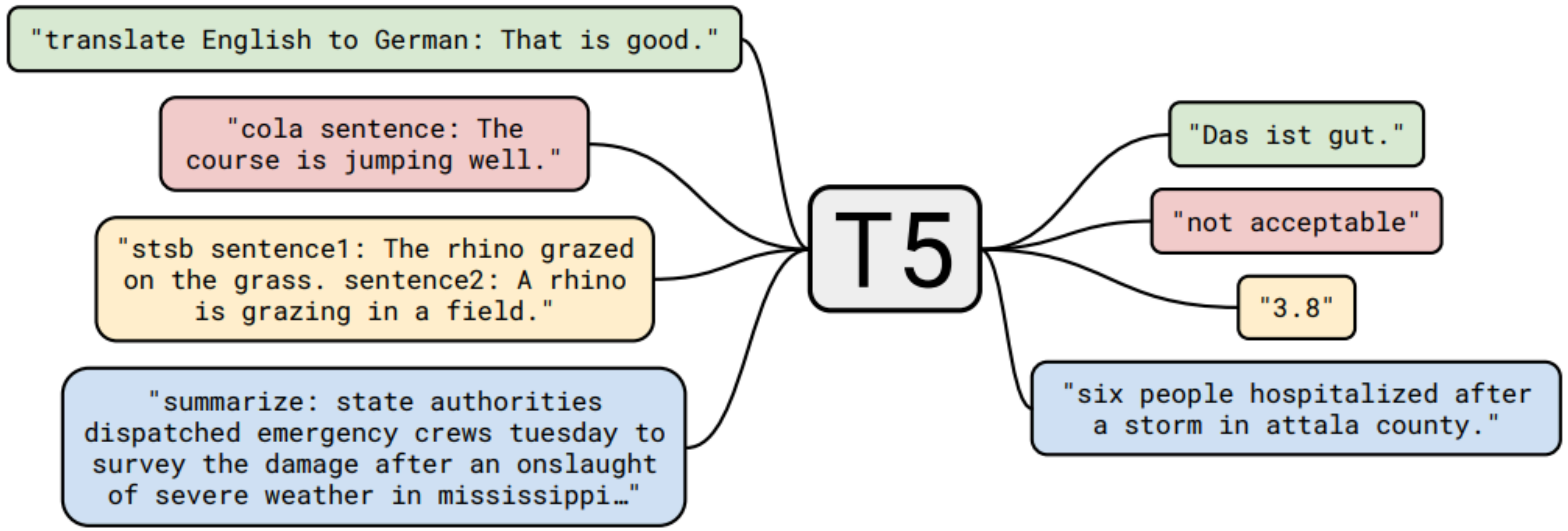
InstructGPT (Ouyang et al., 2022)

Since ChatGPT, instruction tuning is also viewed as the first stage of post-training…

Instruction tuning generalizes to unseen tasks

Comparisons of different paradigms

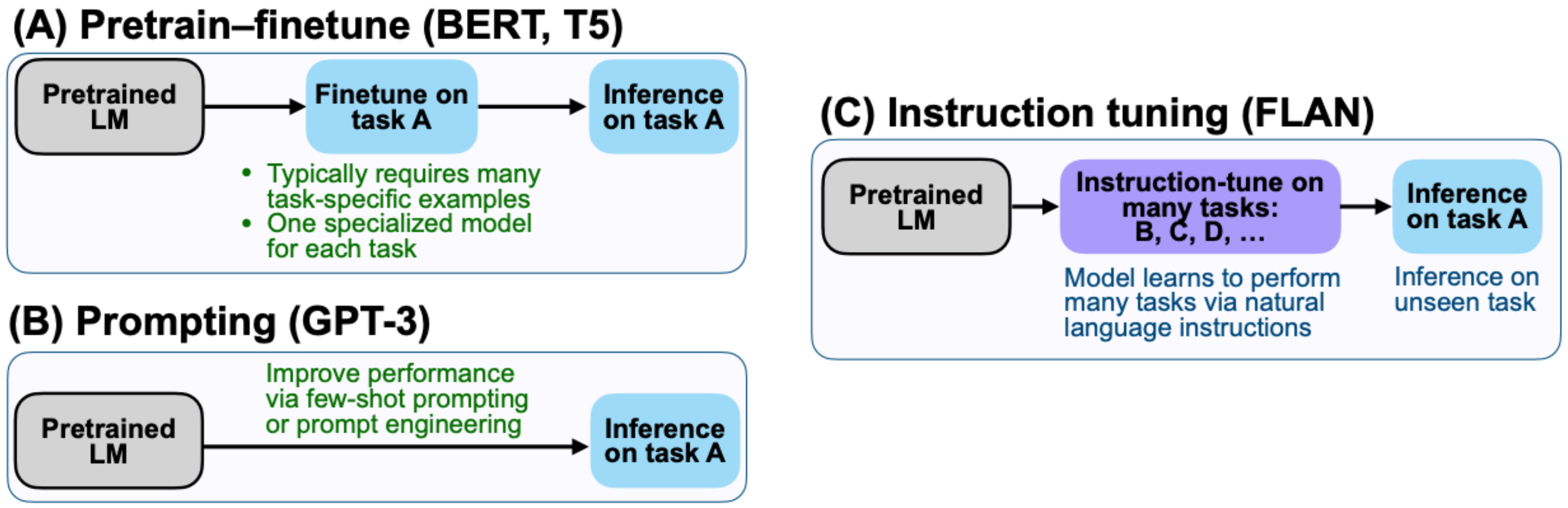
- Pretraining / multi-task training → fine-tuning on task A, evaluating on task A
  - **Examples**: BERT / T5



- Pre-training → prompting with instructions and/or demonstrations on task A
  - **Example**: GPT-3

Comparisons of different paradigms

- Fine-tuning on many tasks with **instructions** → evaluate on unseen task A with **instruction**
    - **Examples**: FLAN, Natural Instructions



(Wei et al., 2021)

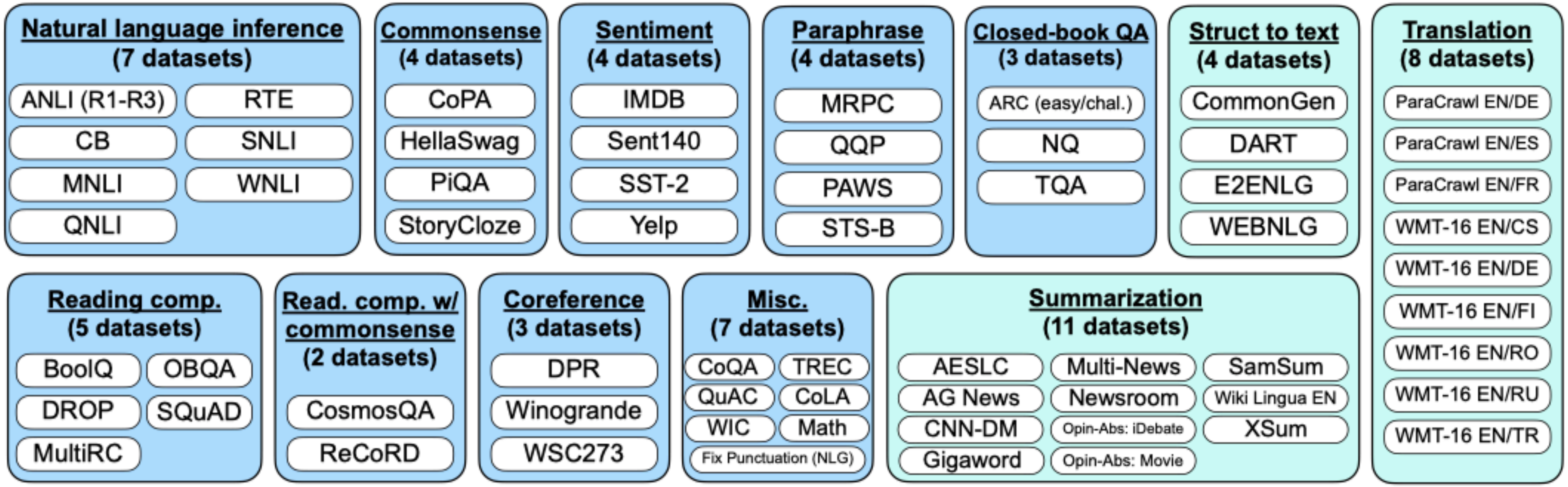
Task	Instance-Level Generalization	Task-Level Generalization
Training data	$X^{\text{train}}, Y^{\text{train}}$	$(I_t, X_t^{\text{train}}, Y_t^{\text{train}})$ $t \in \mathcal{T}_{\text{seen}}$
Evaluation	$x \to y$ where: $(x, y) \in (X^{\text{test}}, Y^{\text{test}})$	$(x, I_t) \to y$ where: $(x, y) \in (X_t^{\text{test}}, Y_t^{\text{test}})$ $t \in \mathcal{T}_{\text{unseen}}$

(Mishra et al., 2021)

“Fine-tunes 140M BART models”

The FLAN paper

- 62 datasets in 12 clusters:

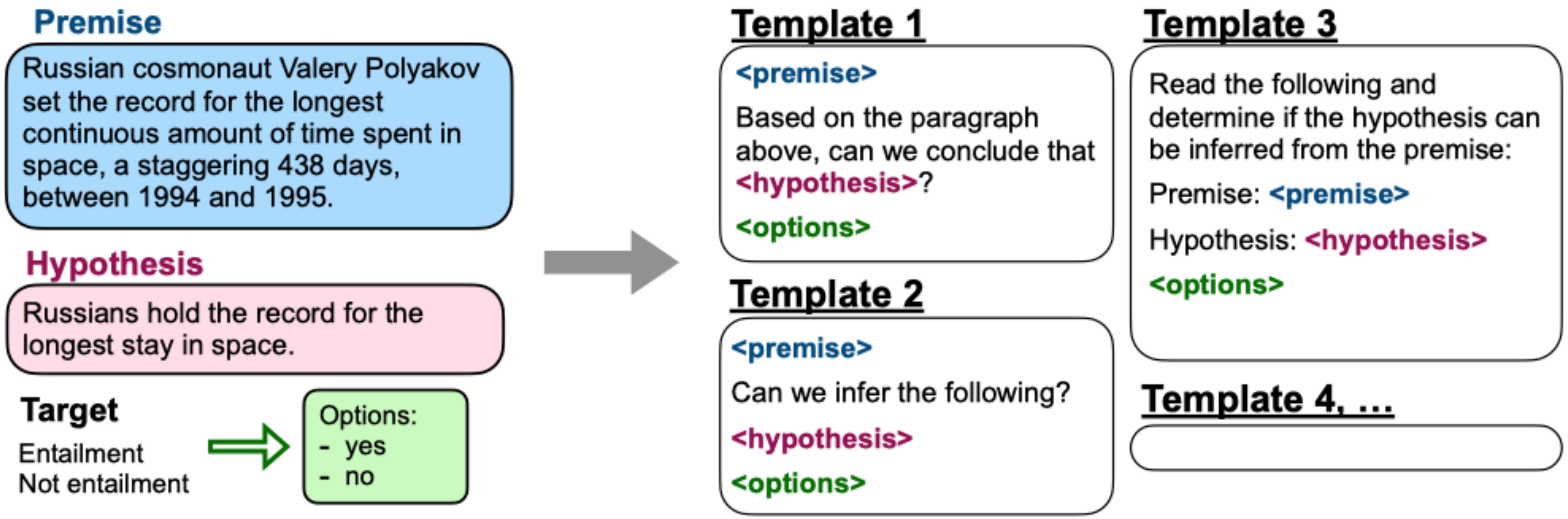


Unseen tasks: any tasks in the same cluster could only appear in training or testing together

Finetuned Language Models Are Zero-Shot Learners

The FLAN paper

- Different instructions (templates) written for the same task:

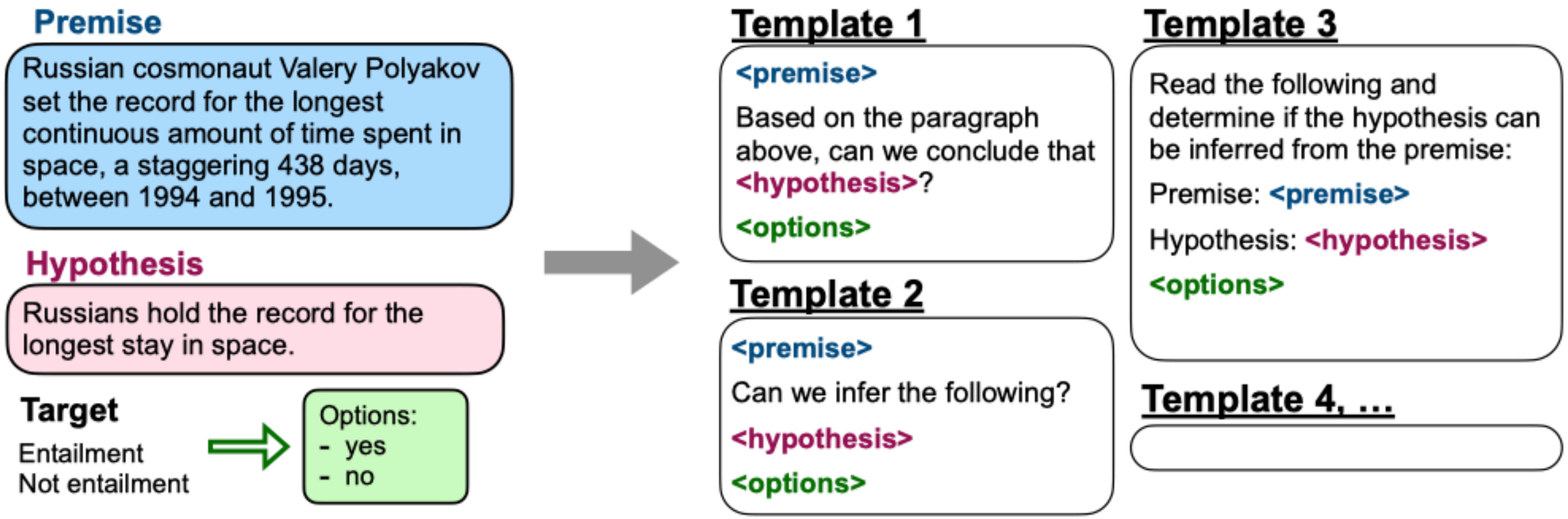


(Some discussions of how to handle classification tasks)

Finetuned Language Models Are Zero-Shot Learners

The FLAN paper

- Different instructions (templates) written for the same task:

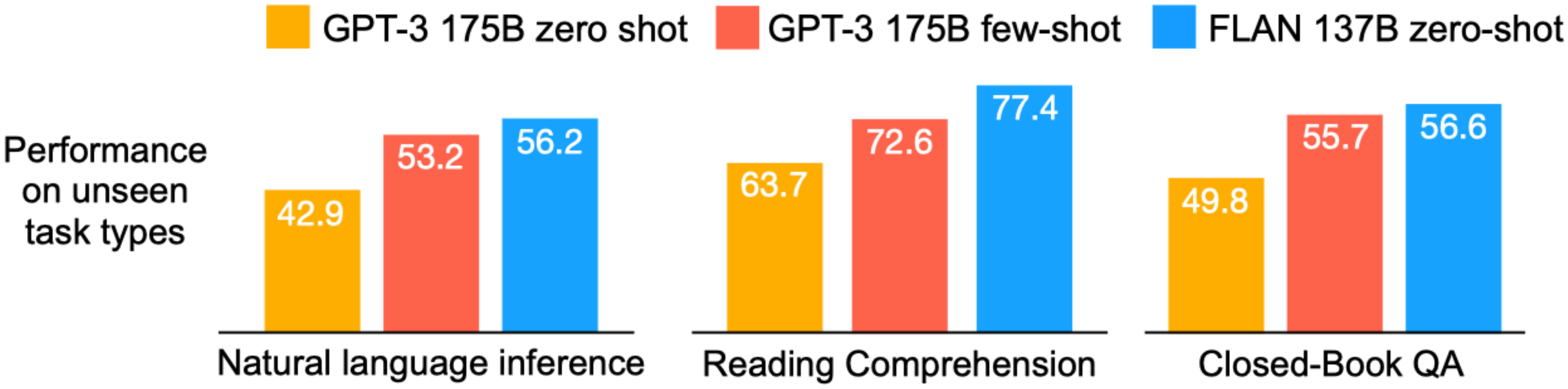


(Some discussions of how to handle classification tasks)

Finetuned Language Models Are Zero-Shot Learners

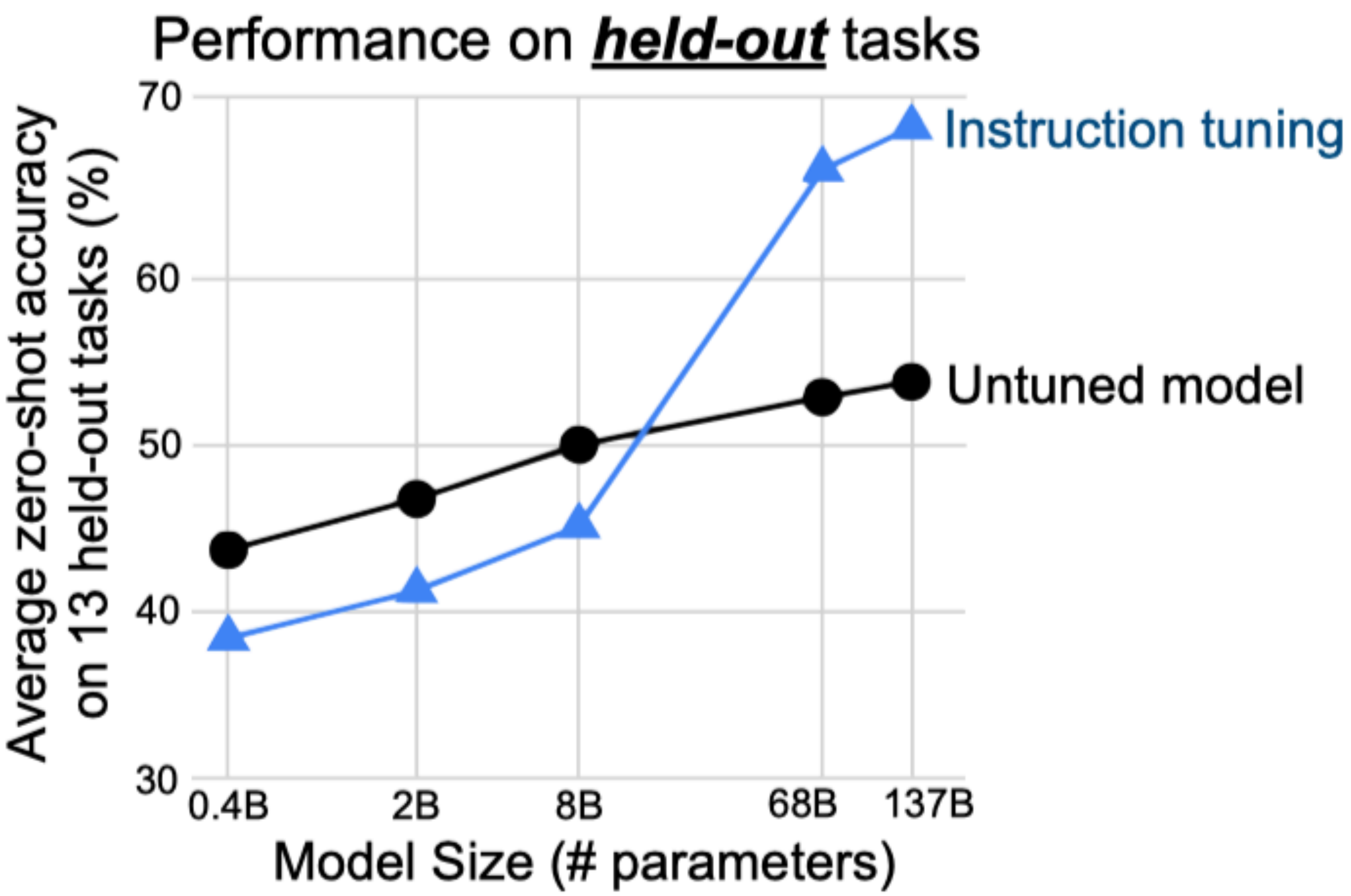
The FLAN paper

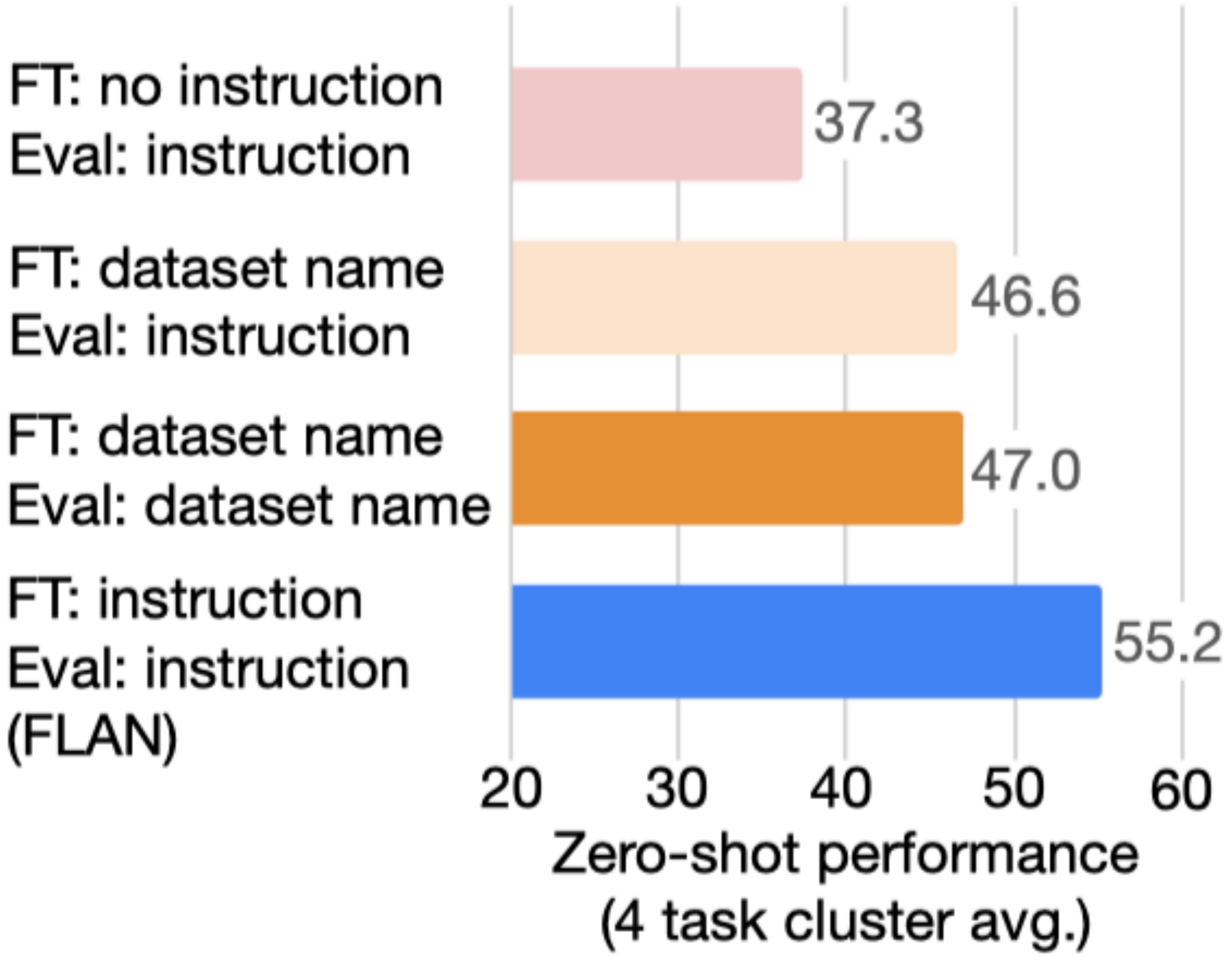
- Fine-tuning on LaMDA-PT (137B parameters)



Finetuned Language Models Are Zero-Shot Learners

The FLAN paper



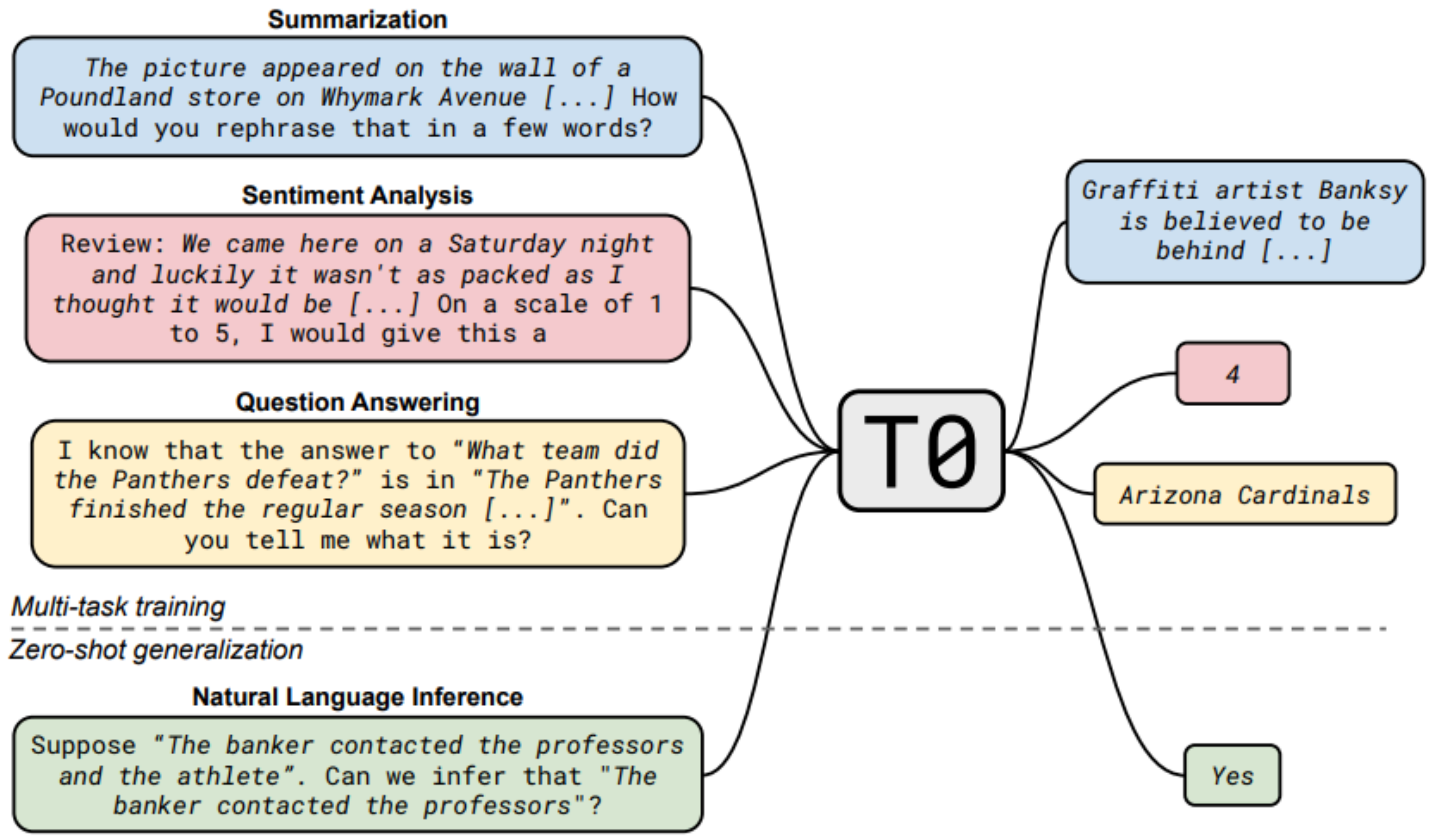


Finetuned Language Models Are Zero-Shot Learners

What factors to consider?

- **Scaling the number of tasks**
- **Format of instructions**: zero-shot, few-shot, chain-of-thought
- **Model architectures** (PaLM, T5, U-PaLM; skipped today)

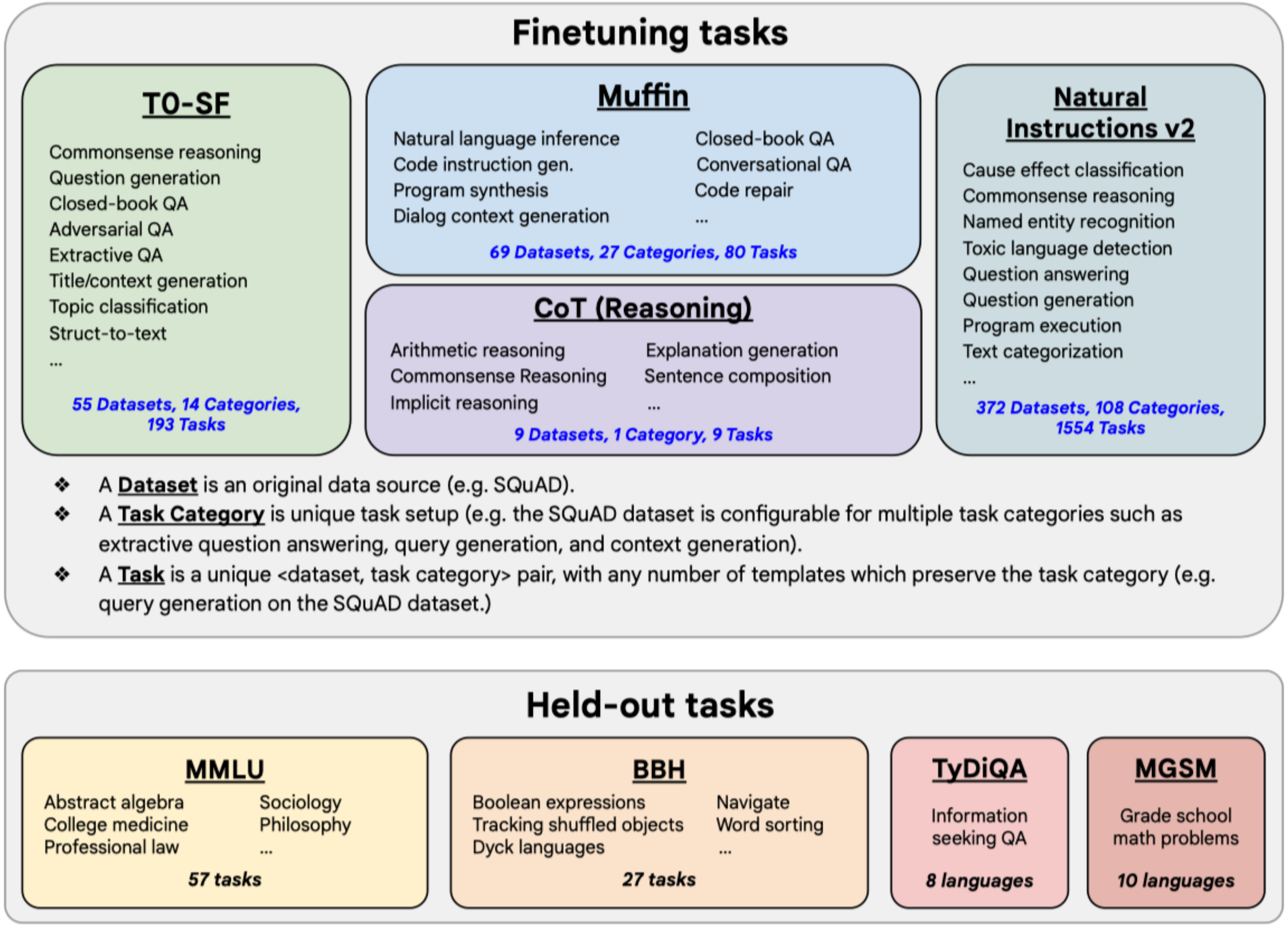
Scaling the number of tasks



T0 (Sanh et al., 2021)

Super-NaturalInstrutions (Sanh et al., 2021)

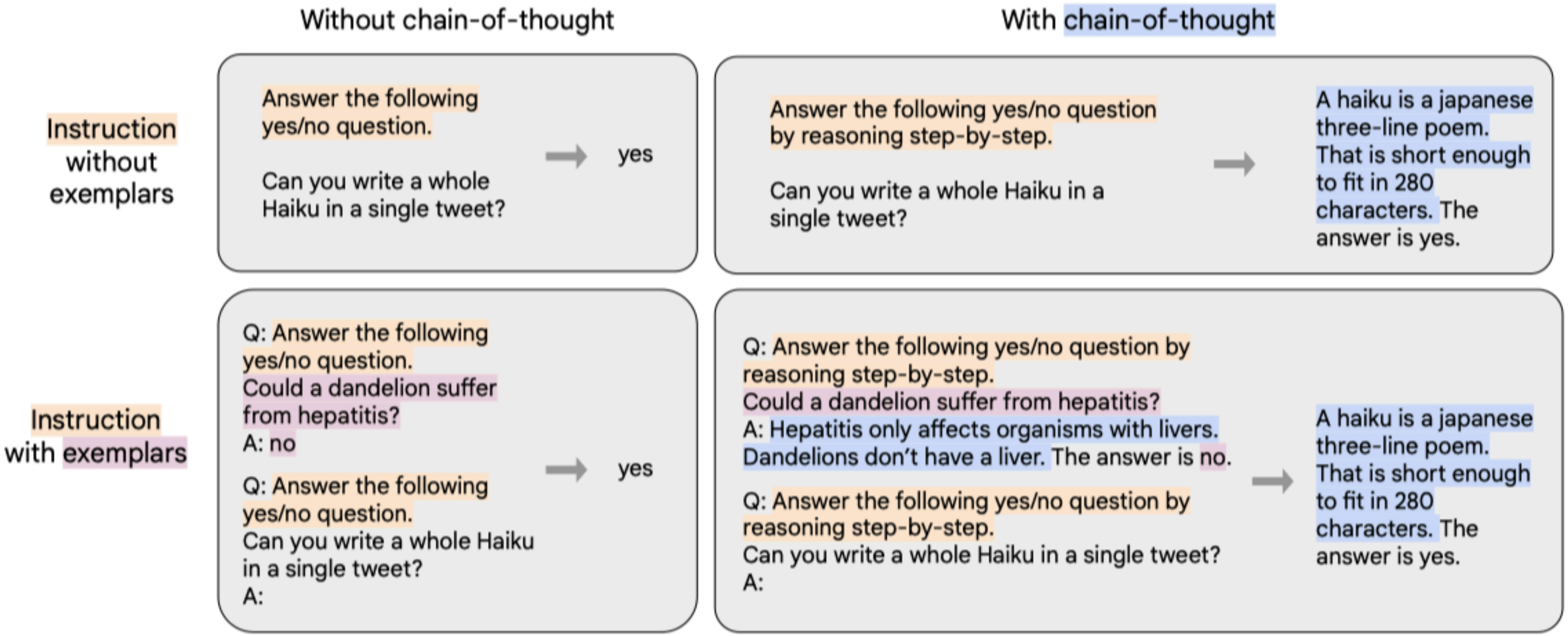
Scaling the number of tasks



- 473 datasets
- 146 task categories
- 1836 tasks

Scaling Instruction-Finetuned Language Models

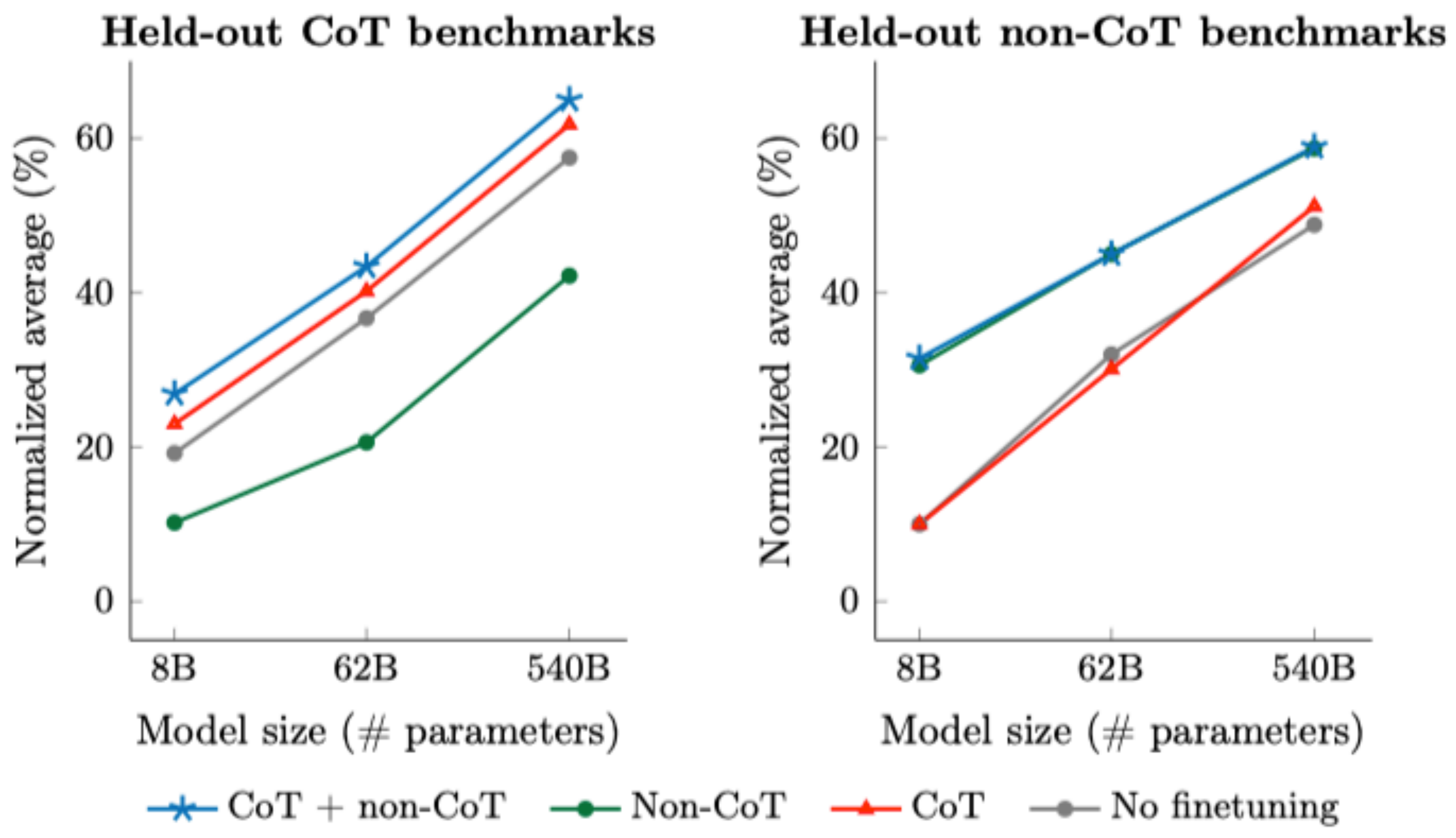
Instruction tuning with exemplars and CoT

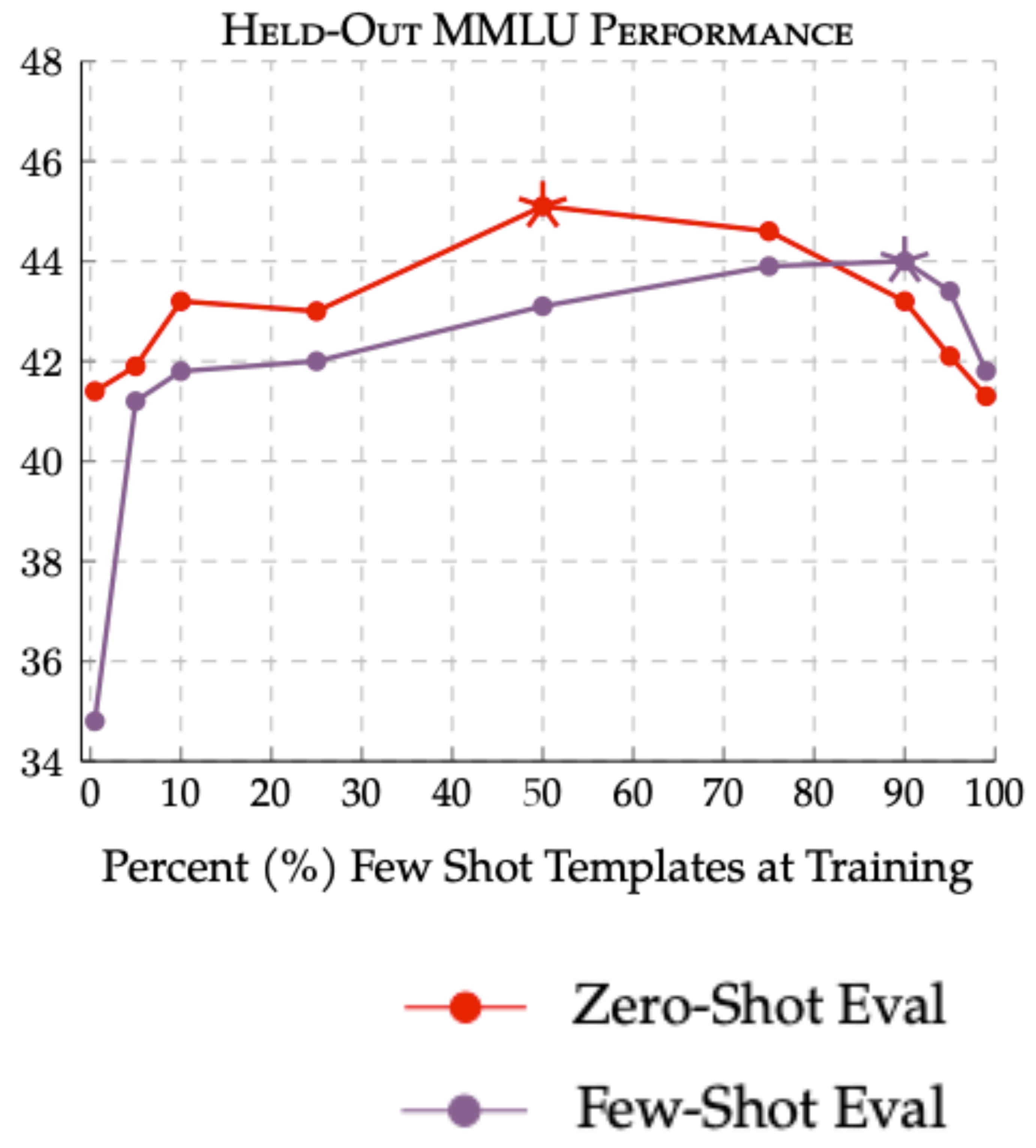


Scaling Instruction-Finetuned Language Models

Interesting results

- Fine-tuning on non-CoT and CoT improves both evaluations
- Fine-tuning on both zero-shot and few-shot improves both evaluations



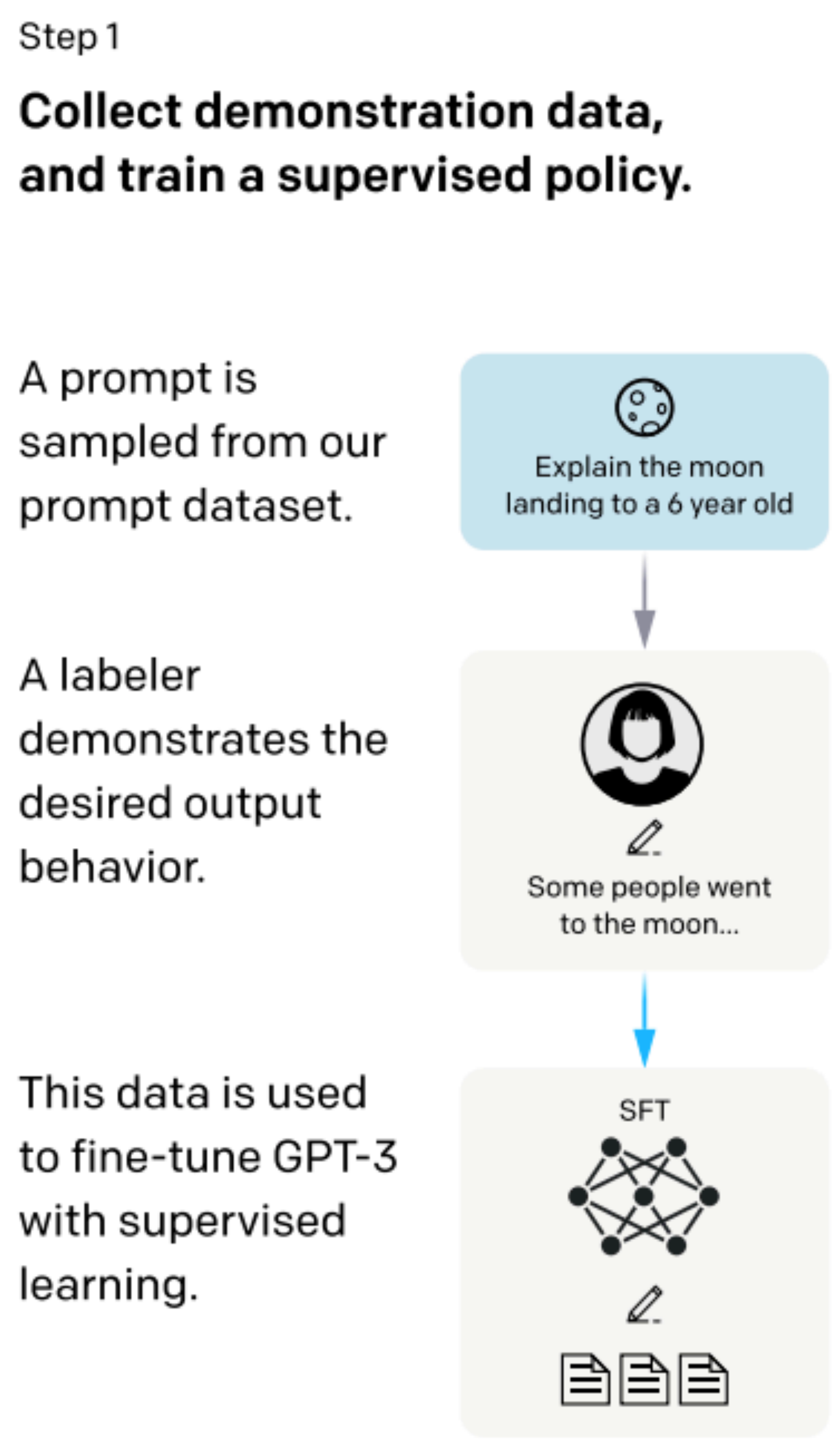


Scaling Instruction-Finetuned Language Models

The Flan Collection: Designing Data and Methods for Effective Instruction Tuning

“Open-ended” instruction tuning

InstructGPT



- 13k data examples

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%

InstructGPT (Ouyang et al., 2022)

InstructGPT

Use Case	Example
brainstorming	List five ideas for how to regain enthusiasm for my career
brainstorming	What are some key points I should know when studying Ancient Greece?
brainstorming	What are 4 questions a user might have after reading the instruction manual for a trash compactor? {user manual}
rewrite	This is the summary of a Broadway play: """ {summary} """ This is the outline of the commercial for that play: """
rewrite	Translate this sentence to Spanish: <English sentence>
rewrite	Create turn-by-turn navigation given this text: Go west on {road1} unto you hit {road2}. then take it east to {road3}. Desination will be a red barn on the right 1.

generation	Write a creative ad for the following product to run on Facebook aimed at parents: Product: {product description}
generation	Write a short story where a brown bear to the beach, makes friends with a seal, and then return home.
classification	{java code} What language is the code above written in?
classification	You are a very serious professor, and you check papers to see if they contain missing citations. Given the text, say whether it is missing an important citation (YES/NO) and which sentence(s) require citing. {text of paper}

InstructGPT (Ouyang et al., 2022)

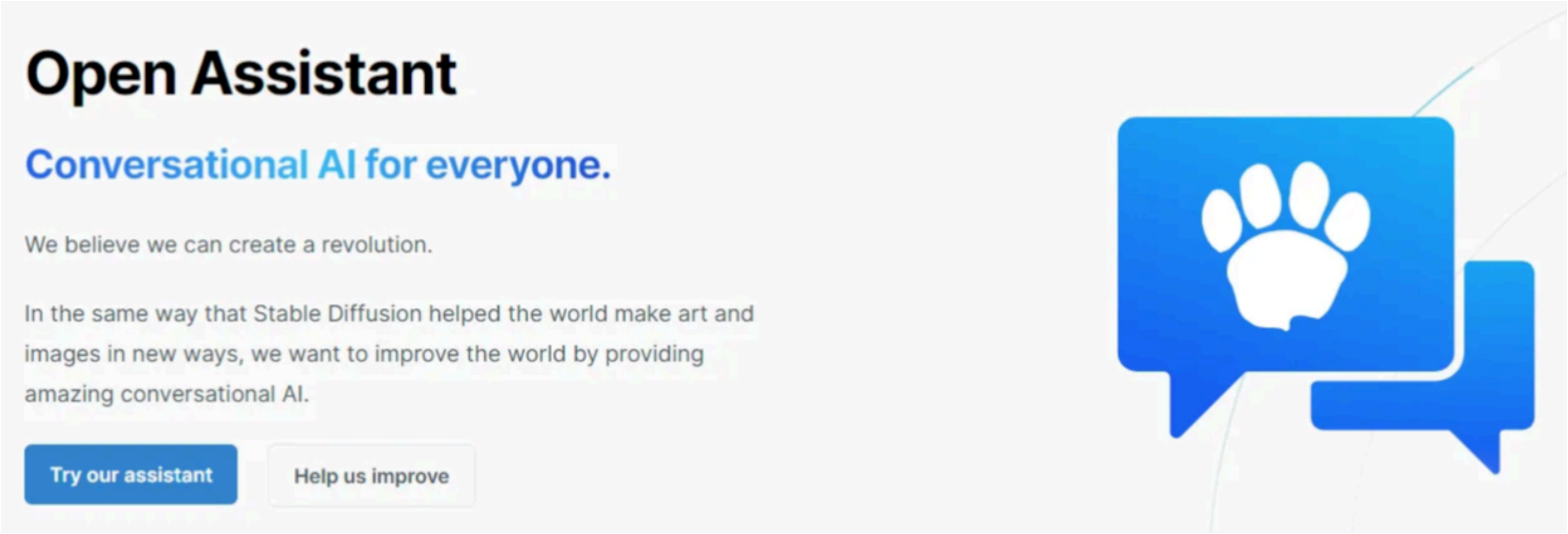
An explosion of instruction datasets

- How can get prompts?
- How can get completions?

- **Option #1:** human-written from scratch

Dolly

15k examples

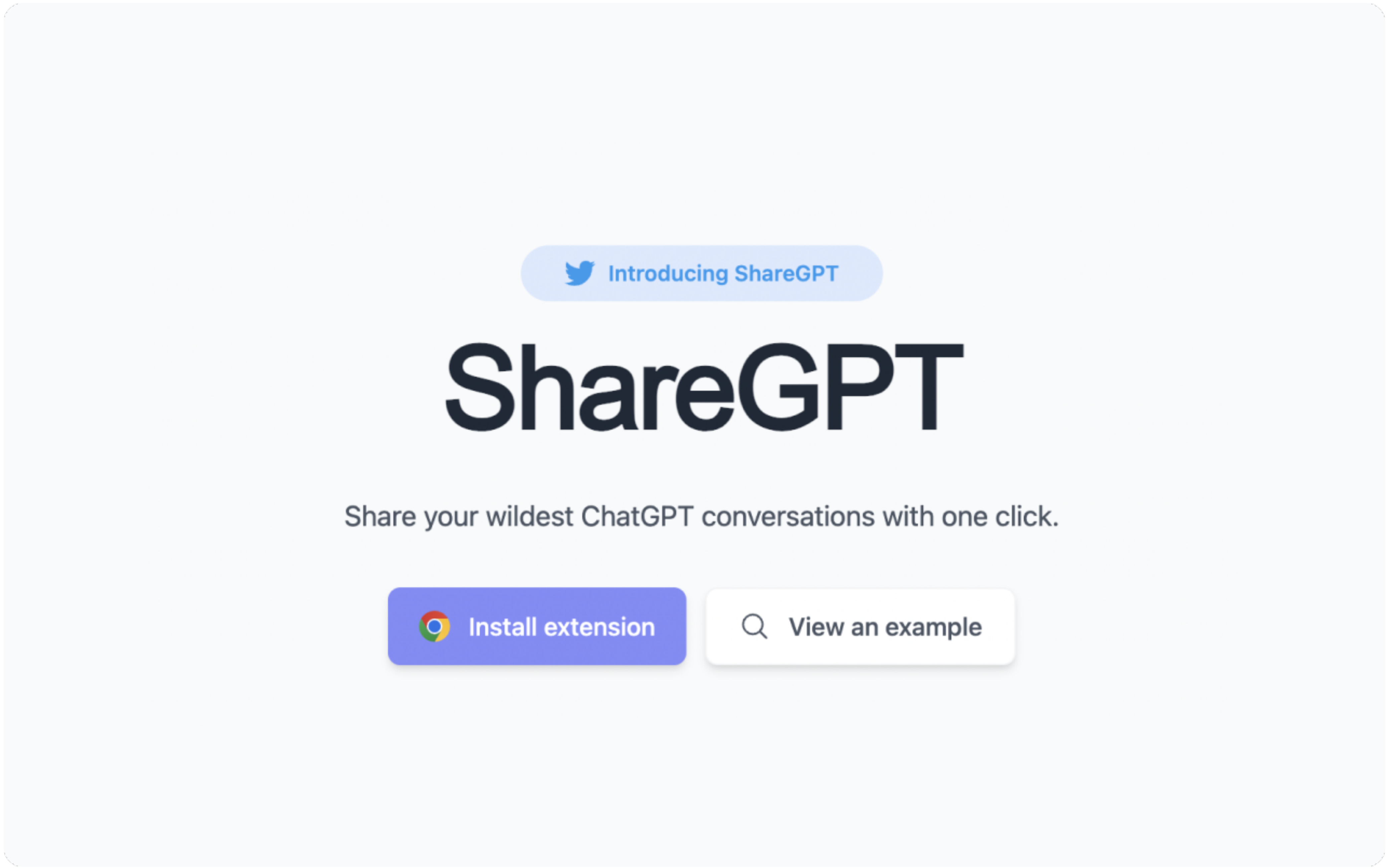


56k examples

An explosion of instruction datasets

- How can get prompts?
- How can get completions?

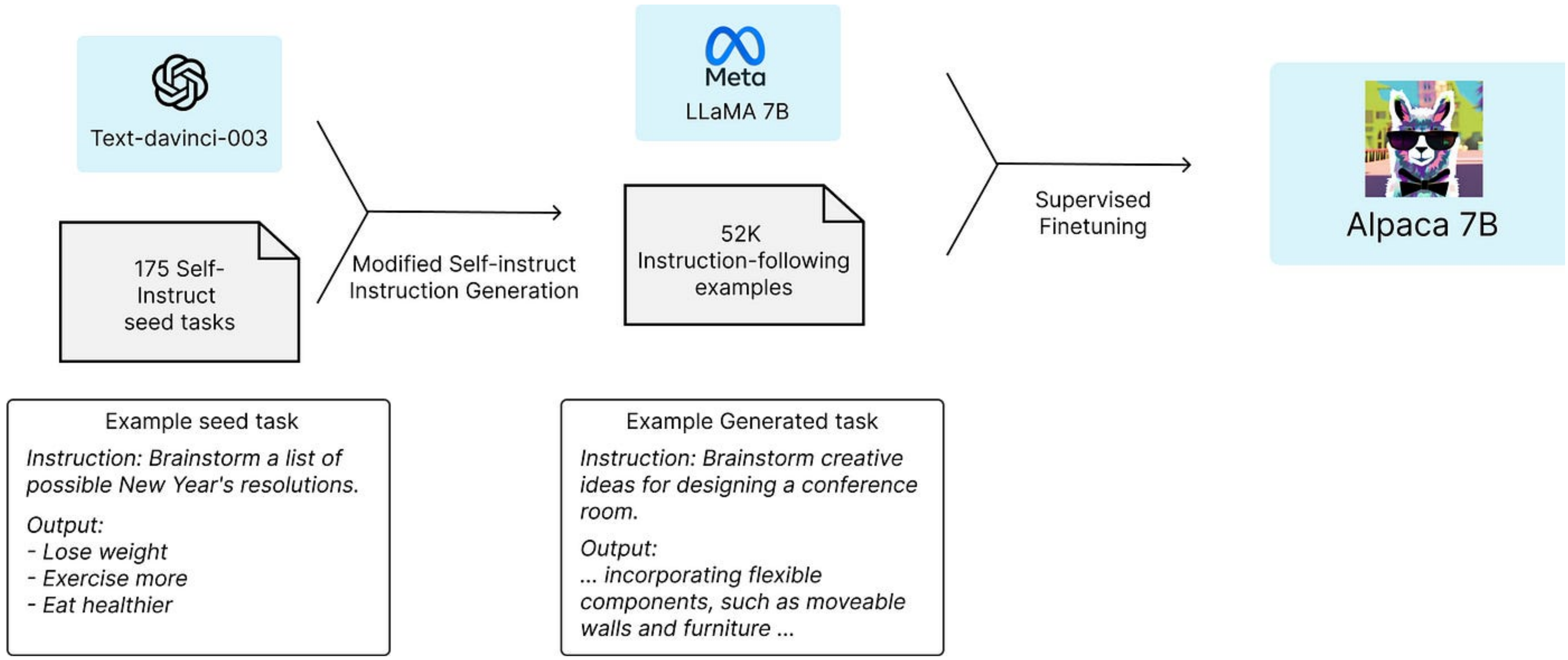
- **Option #2:** the prompts are human-written, and the completions are generated by LLMs (viewed as distillation)



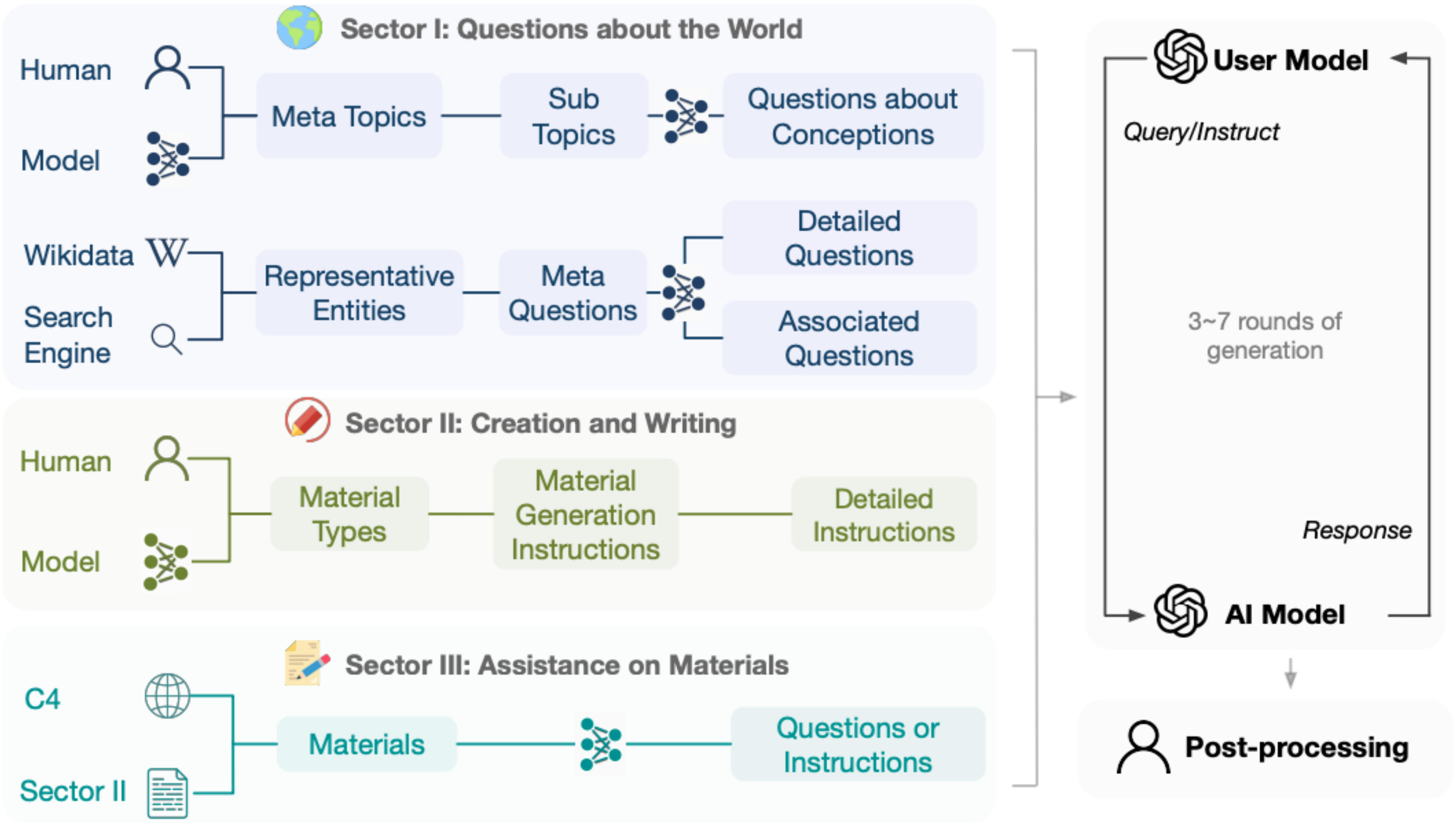
114k examples

An explosion of instruction datasets

- **Option #3:** the instructions can be model-generated too!



Alpaca uses **Self-Instruct** (Wang et al., 2022)



UltraChat (Ding et al., 2023)

The more, the better?

LIMA: Less is more for alignment

LIMA: Less Is More for Alignment

Source	#Examples	Avg Input Len.	Avg Output Len.
Training			
Stack Exchange (STEM)	200	117	523
Stack Exchange (Other)	200	119	530
wikiHow	200	12	1,811
Pushshift r/WritingPrompts	150	34	274
Natural Instructions	50	236	92
Paper Authors (Group A)	200	40	334
Dev			
Paper Authors (Group A)	50	36	N/A
Test			
Pushshift r/AskReddit	70	30	N/A
Paper Authors (Group B)	230	31	N/A

- Knowledge is learned during pre-training; instruction tuning teaches models which subdistribution of formats to use
- Quality and diversity matter - 1000 manually-selected examples work great!

We will have a debate on this paper next week!

Tulu v1

	MMLU (factuality)	GSM (reasoning)	BBH (reasoning)	TydiQA (multilinguality)	Codex-Eval (coding)	AlpacaEval (open-ended)	Average
	EM (0-shot)	EM (8-shot, CoT)	EM (3-shot, CoT)	F1 (1-shot, GP)	P@10 (0-shot)	Win % vs Davinci-003	
Vanilla LLaMa 13B	42.3	14.5	39.3	43.2	28.6	-	-
+SuperNI	49.7	4.0	4.5	**50.2**	12.9	4.2	20.9
+CoT	44.2	40.0	41.9	47.8	23.7	6.0	33.9
+Flan V2	**50.6**	20.0	40.8	47.2	16.8	3.2	29.8
+Dolly	45.6	18.0	28.4	46.5	31.0	13.7	30.5
+Open Assistant 1	43.3	15.0	39.6	33.4	31.9	58.1	36.9
+Self-instruct	30.4	11.0	30.7	41.3	12.5	5.0	21.8
+Unnatural Instructions	46.4	8.0	33.7	40.9	23.9	8.4	26.9
+Alpaca	45.0	9.5	36.6	31.1	29.9	21.9	29.0
+Code-Alpaca	42.5	13.5	35.6	38.9	34.2	15.8	30.1
+GPT4-Alpaca	46.9	16.5	38.8	23.5	**36.6**	63.1	37.6
+Baize	43.7	10.0	38.7	33.6	28.7	21.9	29.4
+ShareGPT	49.3	27.0	40.4	30.5	34.1	**70.5**	42.0
+Human data mix.	50.2	38.5	39.6	47.0	25.0	35.0	39.2
+Human+GPT data mix.	49.3	**40.5**	**43.3**	45.6	35.9	56.5	**45.2**

How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources

Tulu v2

- **FLAN** [Chung et al., 2022]: We use 50,000 examples sampled from FLAN v2.
- **CoT**: To emphasize chain-of-thought (CoT) reasoning, we sample another 50,000 examples from the CoT subset of the FLAN v2 mixture.
- **Open Assistant 1** [Köpf et al., 2023]: We isolate the highest-scoring paths in each conversation tree and use these samples, resulting in 7,708 examples. Scores are taken from the quality labels provided by the original annotators of Open Assistant 1.
- **ShareGPT**[2]: We use all 114,046 examples from our processed ShareGPT dataset, as we found including the ShareGPT dataset resulted in strong performance in prior work.
- **GPT4-Alpaca** [Peng et al., 2023]: We sample 20,000 samples from GPT-4 Alpaca to further include distilled GPT-4 data.
- **Code-Alpaca** [Chaudhary, 2023]: We use all 20,022 examples from Code Alpaca, following our prior V1 mixture, in order to improve model coding abilities.
- ***LIMA** [Zhou et al., 2023]: We use 1,030 examples from LIMA as a source of carefully curated data.
- ***WizardLM Evol-Instruct V2** [Xu et al., 2023]: We sample 30,000 examples from WizardLM, which contains distilled data of increasing diversity and complexity.
- ***Open-Orca** [Lian et al., 2023]: We sample 30,000 examples generated by GPT-4 from OpenOrca, a reproduction of Orca [Mukherjee et al., 2023], which augments FLAN data with additional model-generated explanations.
- ***Science literature**: We include 7,544 examples from a mixture of scientific document understanding tasks— including question answering, fact-checking, summarization, and information extraction. A breakdown of tasks is given in Appendix C.
- ***Hardcoded**: We include a collection of 140 samples using prompts such as 'Tell me about yourself' manually written by the authors, such that the model generates correct outputs given inquiries about its name or developers.

Size	Data	Average -
7B	ShareGPT	47.0
	V1 mix.	47.8
	V2 mix.	**54.2**
13B	V1 mix.	56.0
	V2 mix.	**60.8**
70B	V1 mix.	71.5
	V2 mix.	**72.4**

Camels in a Changing Climate: Enhancing LM Adaptation with Tulu 2

LESS: estimating training influence for data selection

- Choose training data to maximally reduce the validation loss: model-aware and optimizer-aware

Loss on z changes at each step: $\ell(z;\theta^{t+1}) - \ell(z;\theta^t) \approx \langle \nabla \ell(z;\theta^t), \theta^{t+1} - \theta^t \rangle$

SGD step training on x with LR η: $\ell(z;\theta^{t+1}) - \ell(z;\theta^t) \approx \eta \langle \nabla \ell(x;\theta^t), \nabla \ell(z;\theta^t) \rangle$

To maximize loss decrease,
choose x to maximize $\langle \nabla \ell(x;\theta^t), \nabla \ell(z;\theta^t) \rangle$

When training for N epochs, choose training data x to maximize aggregated influence:

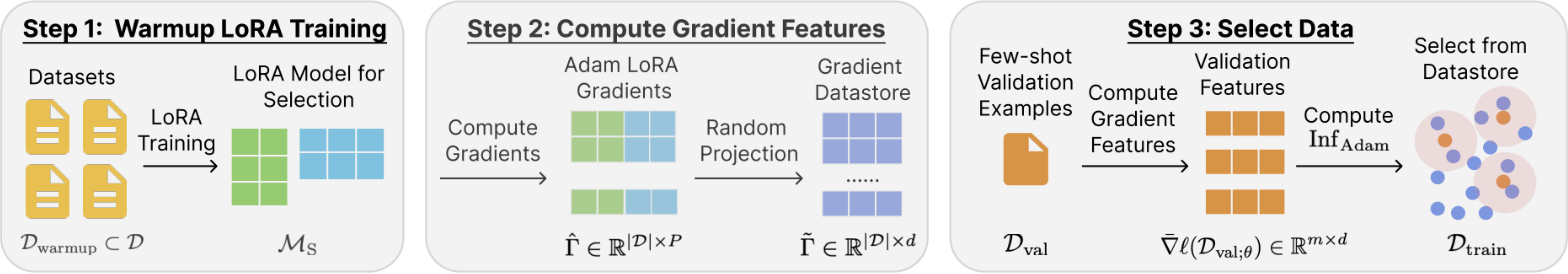
$$\mathrm{Inf}_{\mathrm{SGD}}(x,z) = \sum_{i=1}^{N} \underbrace{\eta_i}_{\text{LR in epoch } i} \langle \nabla \ell(x;\underbrace{\theta_i}_{\text{Model after epoch } i}), \nabla \ell(z;\theta_i) \rangle$$

LESS: Selecting Influential Data for Targeted Instruction Tuning

LESS: estimating training influence for data selection

- LESS made it work for **Adam optimizer** and **instruction data (varying lengths)**
- The algorithm is **practically efficient**

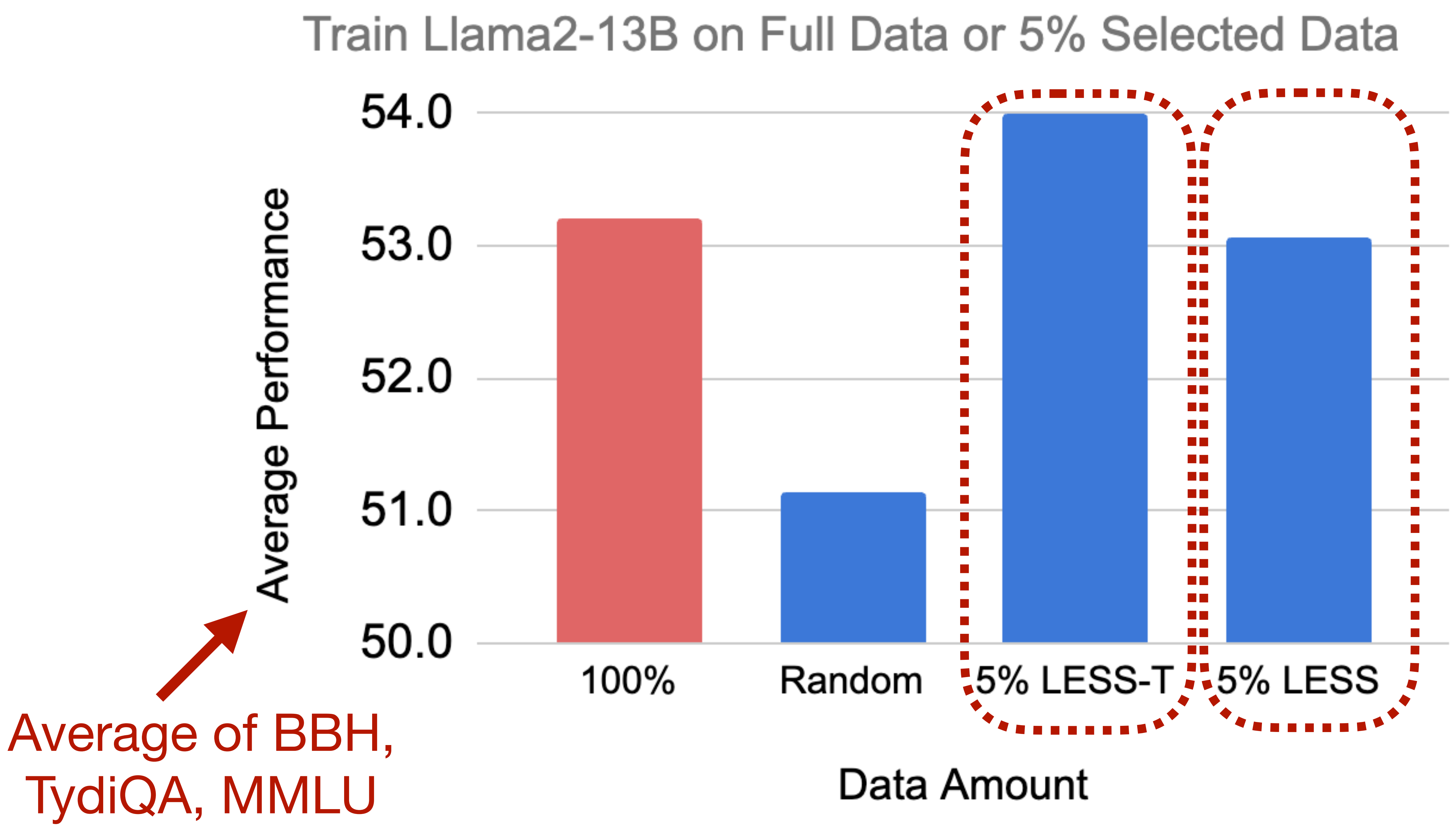
$$\mathrm{Inf}_{\mathrm{Adam}}(x, z) = \sum_{i=1}^{N} \bar{\eta}_i \cos(\nabla l(z; \theta_i), \Gamma(x; \theta_i))$$

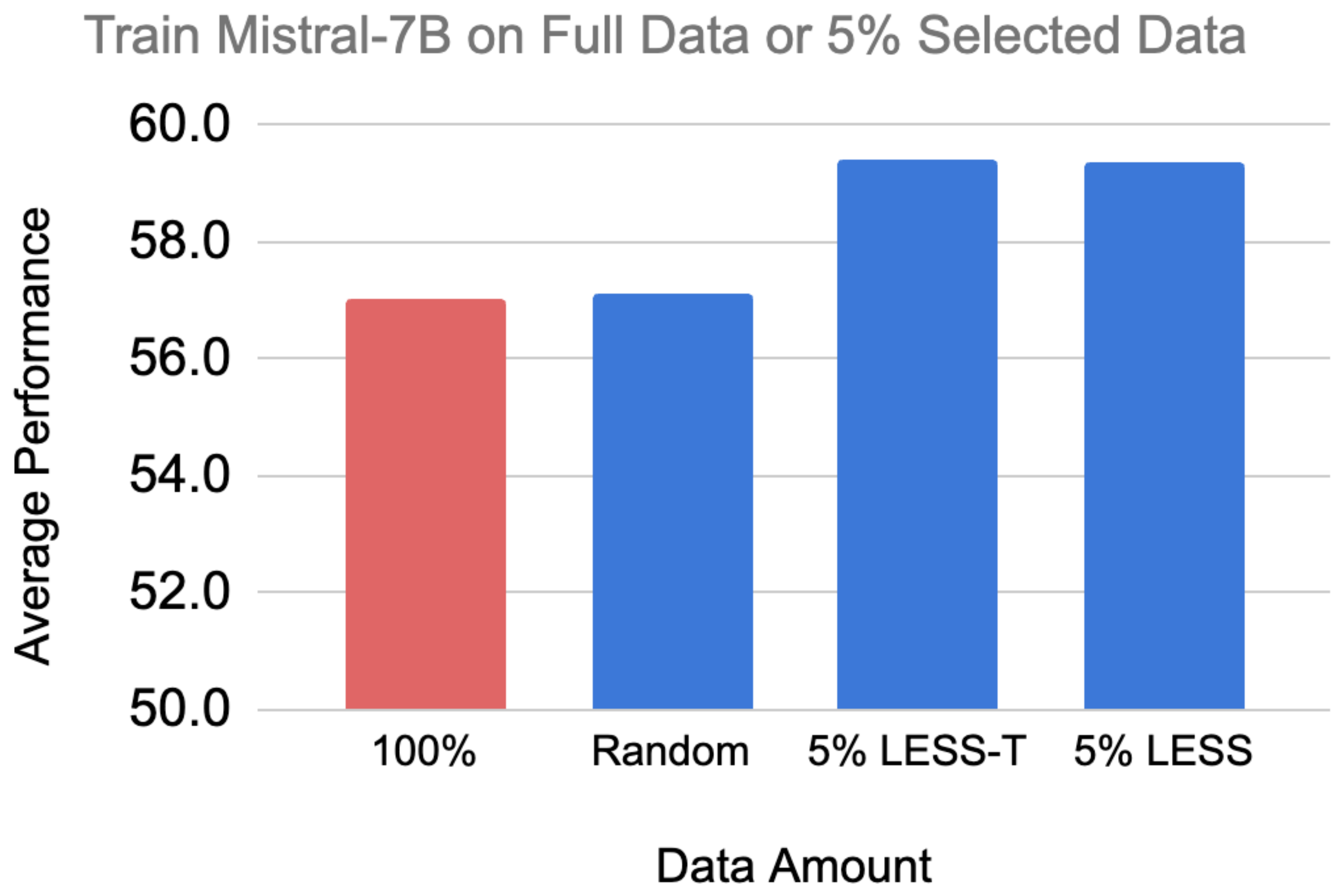


LESS: Selecting Influential Data for Targeted Instruction Tuning

LESS: estimating training influence for data selection

LESS-T: using Llama2-7B for data selection





LESS/LESS-T often outperform using the full datasets.

Data selected using smaller models can transfer!

LESS: Selecting Influential Data for Targeted Instruction Tuning