Reinforcement Learning for Large Language Models

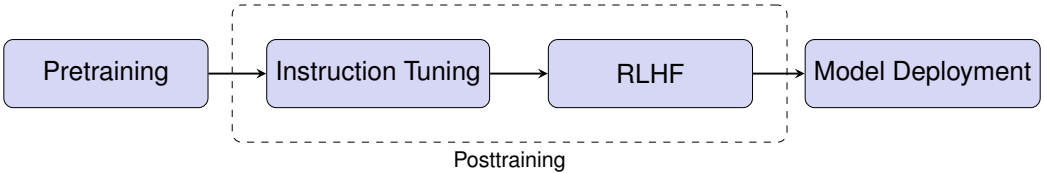
Instructor: Tong Zhang

Part 1: Introduction to Foundation Model Posttraining

Outline

- Overview of Post-training
- Instruction Tuning: Data Curation
- Scaling in SFT

Pretraining to Posttraining Pipeline



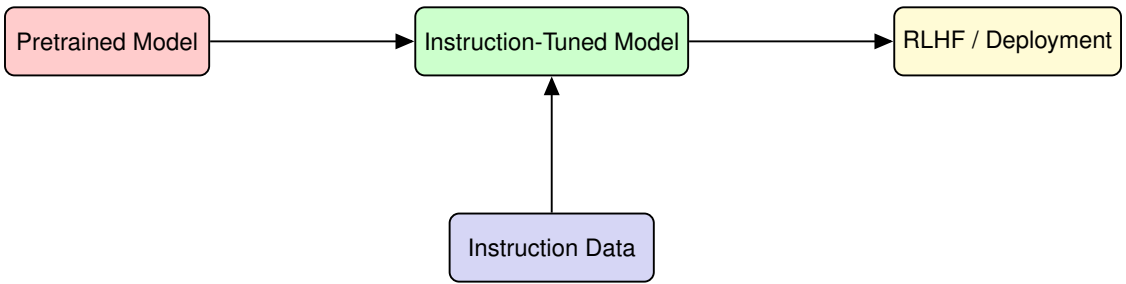
- Pretraining: Train on large text datasets.
- Instruction tuning (SFT): fine-tune on human instructions.
- RLHF: refine with human feedback and reinforcement learning.
- Deployment: use the model for real-world applications.

More recent reasoning models also require RL training of reasoning capability.

Instruction Tuning: What it Achieves

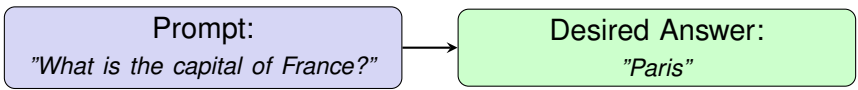
Instruction tuning aligns a pretrained LLM to follow human instructions by learning from prompt–response examples.

- Behavior alignment: outputs become more consistent with user intent.
- Generalization: can transfer to new tasks described in natural language.



Supervised Fine-Tuning (SFT): Data Format

SFT adapts a pretrained language model to follow human instructions using labeled prompt–response pairs.



- Applied after pretraining and before RLHF.
- Learn instruction following and task adaptation.
- Quickly adds useful capabilities before RL refinement.

How do we construct instruction data with good quality, quantity, and diversity?

Instruction Tuning Task Example

To achieve diversity, instruction tuning data should include a diverse set of tasks, each with its own prompt-response pairs.

Example task: summarization

- Input: Article
- Output: Human-written summary
- Train model to predict summary from article

Benefits

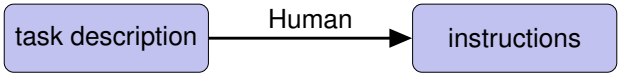
- Better instruction following
- Task specialization
- Much cheaper than full pretraining

Task diversity is essential, and diversity in tasks can lead to task generalization ability beyond seen tasks.

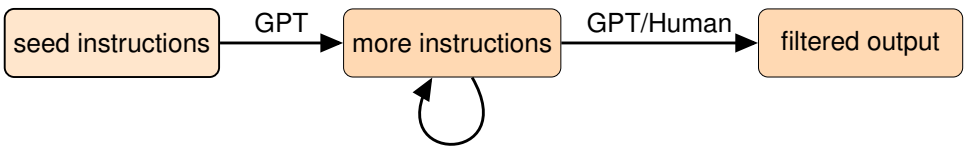
Sources of Instruction Tuning Data

Two primary sources:

- Human-curated data



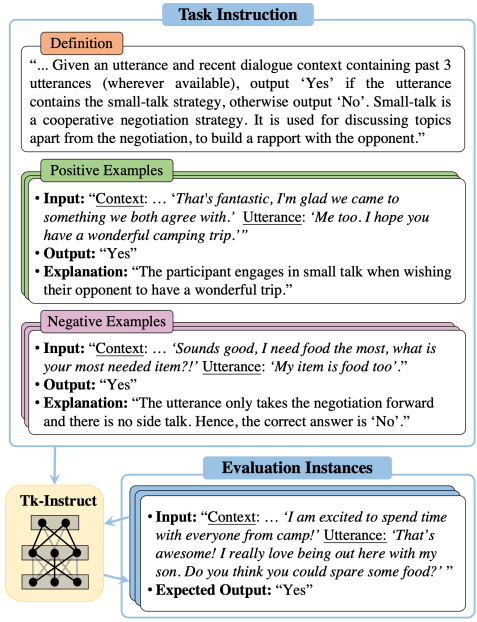
- LLM-generated data



Human-Curated Data: Pros and Cons

- Pros:
  - High-quality
  - Domain-specific
  - Often more reliable
- Cons:
  - Expensive
  - Slow to create
  - Difficult to scale

Source: Wang, Mishra, et al. (2022). "Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks". *arXiv preprint arXiv:2204.07705*



LLM-Generated Data

Pros:

- Use LLMs to produce instruction data.
- Easily scalable and cost-effective.

Cons:

- Quality control: may contain errors.
- Requires filtering and post-processing.
- Bias propagation from the LLM model.

Source: Wang, Kordi, et al. (2022). *Self-Instruct: Aligning Language Model with Self Generated Instructions.* *arXiv preprint arXiv:2212.10560*

Instruction: Given an address and city, come up with the zip code.
Input:
Address: 123 Main Street, City: San Francisco
Output: 94105

Instruction: I am looking for a job and I need to fill out an application form. Can you please help me complete it?
Input:
Application Form:
Name:_____ Age:_____ Sex:_____
Phone Number:_____ Email Address:_____
Education:_____ ...
Output:
Name: John Doe Age: 25 Sex: Male
Phone Number: ...

Instruction: How to write a code for converting degrees fahrenheit to celsius.
Input: Null
Output:

```
def convert_fahrenheit_to_celsius(fahr):
    celsius = (fahr - 32) * 5 / 9
    return celsius
```

Key Factors in Instruction Data Generation

- Quality: high-quality data enhances model performance.
- Quantity: LLMs require massive amounts of data for instruction tuning.
- Diversity: essential for generalization to unseen tasks.
  - Different tasks across multiple domains.
  - Different instruction types: Zero-shot, few-shot, and chain-of-thought examples.

Natural Instructions: Human-Curated Instruction Data

- Source: crowdsourced human-written instructions.
- Scale: 61 NLP tasks, 193k instances.
- Structure: title, definition, examples, and constraints.
- Goal: train on seen tasks and generalize to unseen tasks.

Result:

- Detailed instructions improve generalization to unseen tasks.
- About 19% improvement reported in held-out evaluation.

Human-Curated Data: Super-Natural Instructions

Human-curated tasks and instructions by NLP experts to train models to generalize to unseen tasks.

Statistic	Value
# of tasks	1616
# of task types	76
# of languages	55
# of domains	33
# of non-English tasks	576
Avg. definition length (words per task)	56.6
Avg. # of positive examples (per task)	2.8
Avg. # of negative examples (per task)	2.4
Avg. # of instances (per task)	3106.0

Source: Wang, Mishra, et al. (2022). "Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks". *arXiv preprint arXiv:2204.07705*

Comparison to Natural Instructions

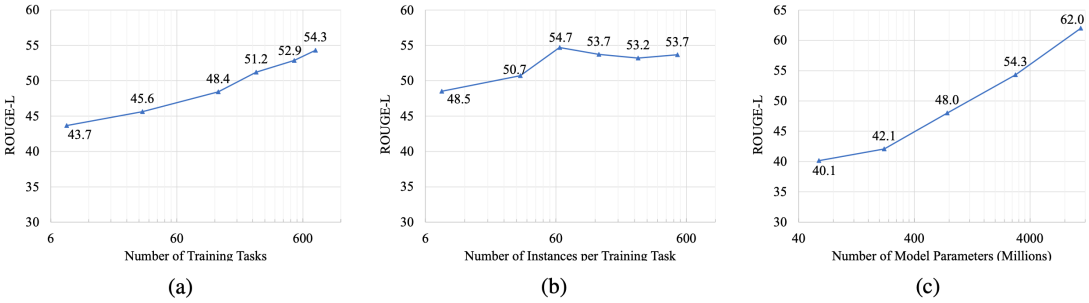
Resource $\rightarrow$	**Sup-NatInst**	**NatInst**
Has task instructions?	✓	✓
Has negative examples?	✓	✓
Has non-English tasks?	✓	**x**
Is public?	✓	✓
# of tasks	1616	61
# of annotated task types	76	6
Avg. task definition length (words)	56.6	134.4

Table: Comparison of Super-Natural and Natural Instructions

Example Task for Human Expert: Cause Effect Classification

Task Type	Cause Effect Classification
Task ID	task828_copa_cause_effect_classification
Definition	In this task you are given two statements. You must judge whether the second sentence is the cause or effect of the first one. Label the instances as “cause” or “effect” based on your judgment. The sentences are separated by a newline character.
Positive Example	**Input:** The women met for coffee. They wanted to catch up with each other. **Output:** cause **Explanation:** The women met for coffee because they wanted to catch up with each other.
Negative Example	**Input:** My body cast a shadow over the grass. The sun was rising. **Output:** effect **Explanation:** The rising of the sun isn’t an effect of casting a shadow over the grass.
Instance	**Input:** The woman tolerated her friend’s difficult behavior. The woman knew her friend was going through a hard time. **Valid Output:** [“cause”]

Results: Importance of Task Diversity



Source: Wang, Mishra, et al. (2022). "Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks". *arXiv preprint arXiv:2204.07705*

ROUGE-L measures the similarity between a system-generated response and a reference response (human created) based on the longest common subsequence (LCS) between them. It is highly strongly correlated with human evaluation.

Human-Curated Data: FLAN Collection

FLAN Collection is a collection of public datasets designed to support generalization across many NLP tasks released by Google.

Focus on scaling up tasks and response enrichment techniques

- Task scaling
- Mixture of zero-shot, few-shot, and chain-of-thought prompts
- Dataset mixing

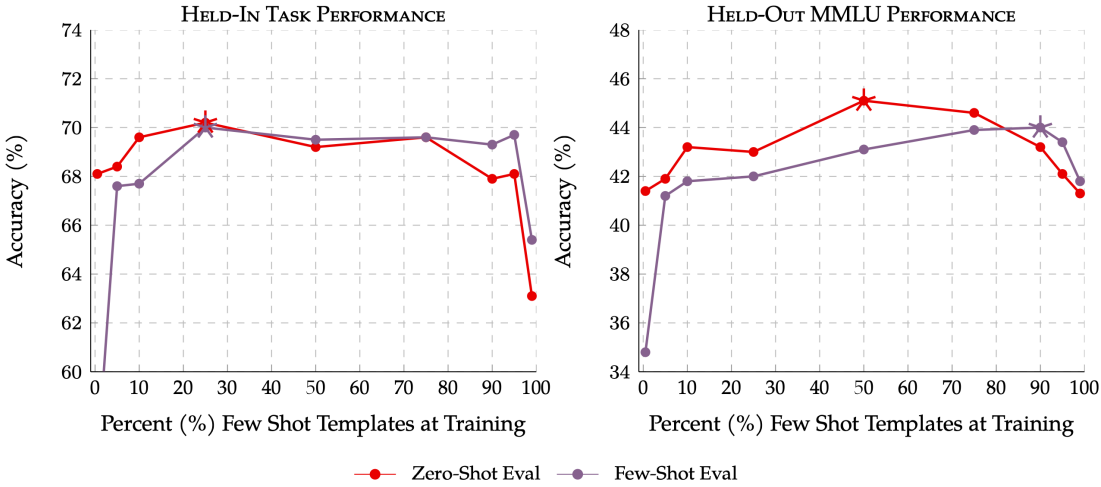
Longpre et al. (2023). "The flan collection: designing data and methods for effective instruction tuning". *ICML 2023*

Comparison of Data and Models

		Model Details				Data Collection & Training Details			
Release	Collection	Model	Base	Size	Public?	Prompt Types	Tasks in Flan	# Exs	Methods
2020 05	UnifiedQA	UnifiedQA	RoBerta	110-340M	P	ZS	46 / 46	750k	
2021 04	CrossFit	BART-CrossFit	BART	140M	NP	FS	115 / 159	71.M	
2021 04	Natural Inst v1.0	Gen. BART	BART	140M	NP	ZS / FS	61 / 61	620k	+ Detailed k-shot Prompts
2021 09	Flan 2021	Flan-LaMDA	LaMDA	137B	NP	ZS / FS	62 / 62	4.4M	+ Template Variety
2021 10	P3	T0, T0+, T0++	T5-LM	3-11B	P	ZS	62 / 62	12M	+ Template Variety + Input Inversion
2021 10	MetaICL	MetaICL	GPT-2	770M	P	FS	100 / 142	3.5M	+ Input Inversion + Noisy Channel Opt
2021 11	ExMix	ExT5	T5	220M-11B	NP	ZS	72 / 107	500k	+ With Pretraining
2022 04	Super-Natural Inst.	*Tk*-Instruct	T5-LM, mT5	11-13B	P	ZS / FS	1556 / 1613	5M	+ Detailed k-shot Prompts + Multilingual
2022 10	GLM	GLM-130B	GLM	130B	P	FS	65 / 77	12M	+ With Pretraining + Bilingual (en, zh-cn)
2022 11	xP3	BLOOMz, mT0	BLOOM, mT5	13-176B	P	ZS	53 / 71	81M	+ Massively Multilingual
2022 12	Unnatural Inst.†	T5-LM-Unnat. Inst.	T5-LM	11B	NP	ZS	~20 / 117	64k	+ Synthetic Data
2022 12	Self-Instruct†	GPT-3 Self Inst.	GPT-3	175B	NP	ZS	Unknown	82k	+ Synthetic Data + Knowledge Distillation
2022 12	OPT-IML Bench†	OPT-IML	OPT	30-175B	P	ZS + FS, CoT	~2067 / 2207	18M	+ Template Variety + Input Inversion + Multilingual
2022 10	Flan 2022 (ours)	Flan-T5, Flan-PaLM	T5-LM, PaLM	10M-540B	P NP	ZS + FS, CoT	1836	15M	+ Template Variety + Input Inversion + Multilingual

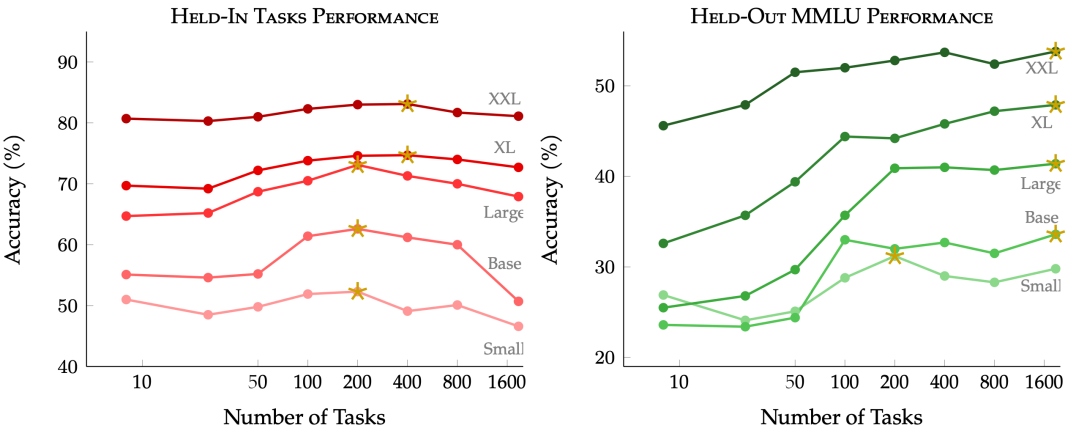
Source: Longpre et al. (2023). "The flan collection: designing data and methods for effective instruction tuning". *ICML 2023*

Adding Few-shot Prompts Improves Performance



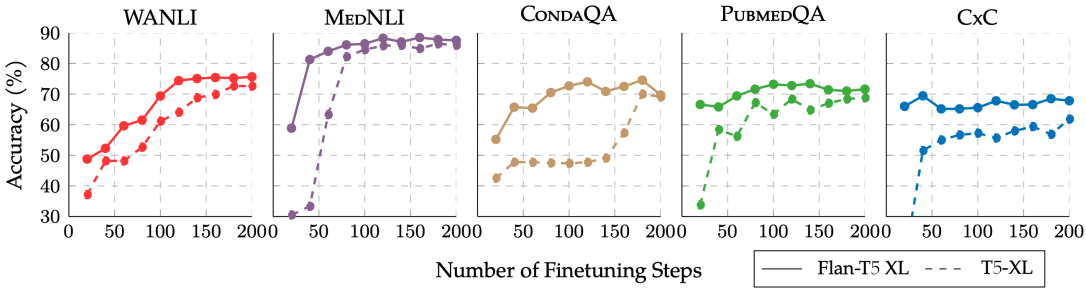
Source: Longpre et al. (2023). "The flan collection: designing data and methods for effective instruction tuning". *ICML 2023*

Scaling with Tasks and Model Size



Source: Longpre et al. (2023). "The flan collection: designing data and methods for effective instruction tuning". *ICML 2023*

Single-Task Fine-Tuning on Held-Out Tasks

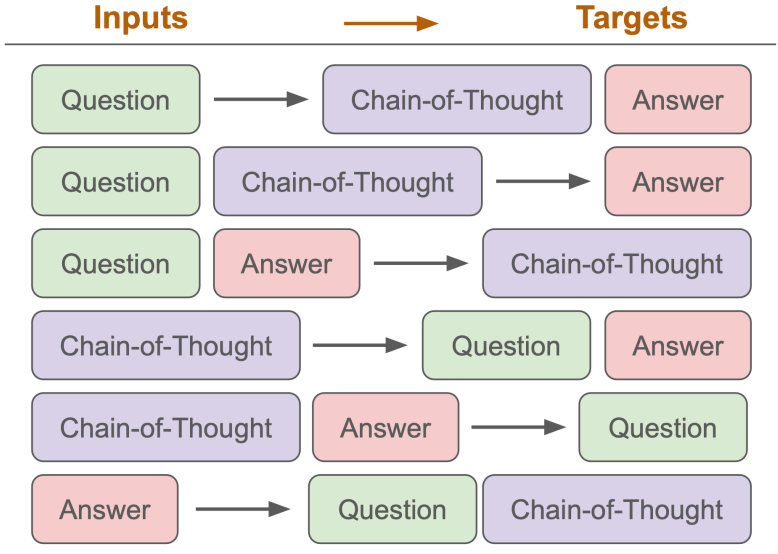


Instruction Tuning Improves Single-Task Fine-Tuning

- Flan-T5 XL: Instruction-tuned model
- T5-XL: base-model (without instruction-tuning)

Source: Longpre et al. (2023). "The flan collection: designing data and methods for effective instruction tuning". *ICML 2023*

Input Inversion Helps Held-Out Task Performance



Source: Longpre et al. (2023). "The flan collection: designing data and methods for effective instruction tuning". *ICML 2023*

Dataset Mixing

Train Mixtures	Metrics		
	Held-In	**CoT**	**MMLU**
All (Equal)	64.9	41.4	47.3
All - Flan 2021	55.3	38.6	45.7
All - T0-SF	63.2	**43.4**	44.7
All - Super-Nat. Inst.	65.9	42.2	46.8
All - CoT	65.6	29.1	46.8
All - Prog. Synth.	66.9	42.3	46.8
All - Dialog	65.4	40.3	47.1
All (Weighted)	**66.4**	40.1	**48.1**

- FLAN Collection contains multiple dataset sources
- Proper mixing improves performance (requires tuning of mixing weights)
- Weighting controls quality; mixing improves quantity and diversity

Source: Longpre et al. (2023). "The flan collection: designing data and methods for effective instruction tuning". *ICML 2023*

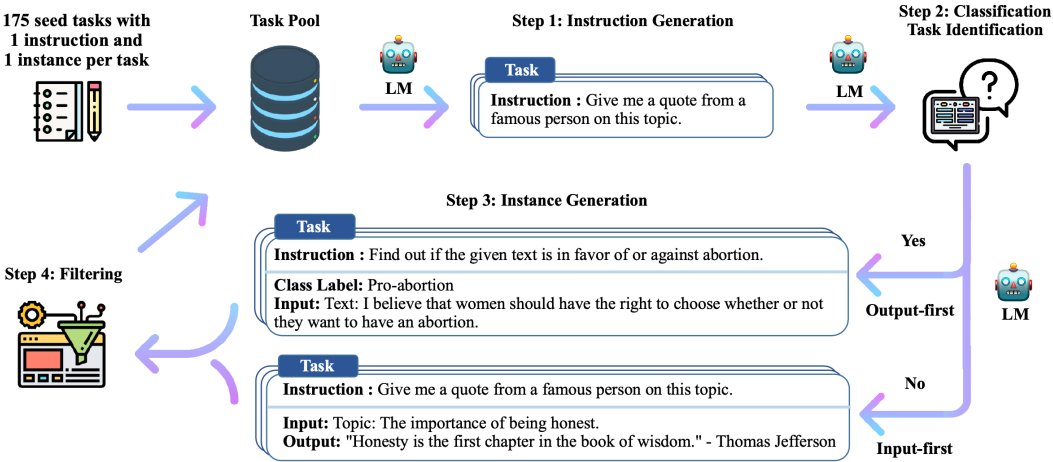
LLM Generated Data: Self-Instruct

Self-Instruct generates instruction-following data without human intervention.

Iterative Bootstrap Process:

1. Start with a small seed set of tasks as the initial task pool
2. Random tasks are sampled from the task pool, and used as in context examples to prompt an LLM to generate additional task instances
3. Filtering low-quality or similar generations
4. Add them back to the task repository

System Overview



Source: Wang, Kordi, et al. (2022). *Self-Instruct: Aligning Language Model with Self Generated Instructions.* *arXiv preprint arXiv:2212.10560*

Example: New Instruction Generation Prompt

Come up with a series of tasks:
Task 1: instruction for existing task 1
Task 2: instruction for existing task 2
Task 3: instruction for existing task 3
Task 4: instruction for existing task 4
Task 5: instruction for existing task 5
Task 6: instruction for existing task 6
Task 7: instruction for existing task 7
Task 8: instruction for existing task 8
Task 9:

Eight existing instructions are randomly sampled from the task pool for in-context demonstration. The model is allowed to generate instructions for new tasks, until it stops its generation, reaches its length limit or generates “Task 16” tokens.

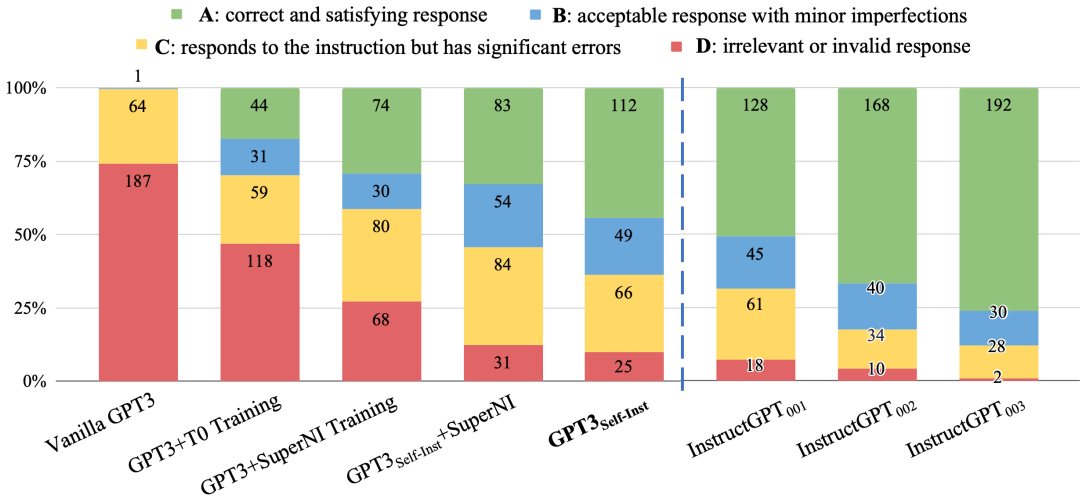
Self-Instruct Data Quantity, Diversity, and Quality

- 52,445 instructions and 82,439 instances generated.
- Many new instructions were generated, with limited overlap with the seeds.
- Filtering is needed to ensure correctness

Quality Review Question	**Yes %**
Does the instruction describe a valid task?	92%
Is the input appropriate for the instruction?	79%
Is the output a correct and acceptable response to the instruction and input?	58%
All fields are valid	54%

Table: Quality Review of Instruction Tasks

Self-Instruct Data Compared to Human Curated Data



Source: Wang, Kordi, et al. (2022). *Self-Instruct: Aligning Language Model with Self Generated Instructions.* *arXiv preprint arXiv:2212.10560*

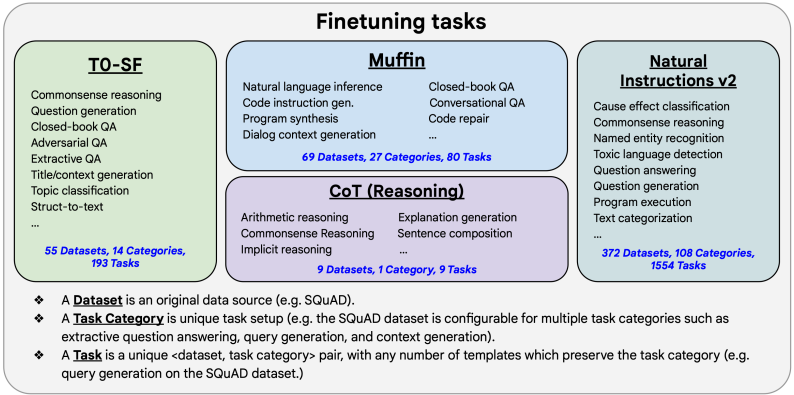
Scaling in SFT

Can we achieve better results by scaling up SFT?

1. scaling the number of tasks.
2. scaling the model size.
3. fine-tuning on chain-of-thought data.

Chung et al. (2024). "Scaling Instruction-Finetuned Language Models". *Journal of Machine Learning Research*

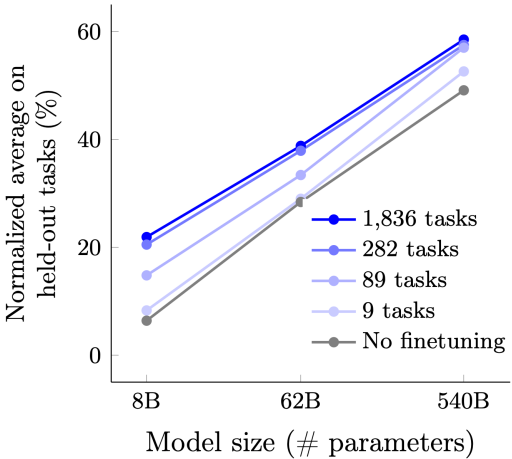
Instruction Tuning Tasks

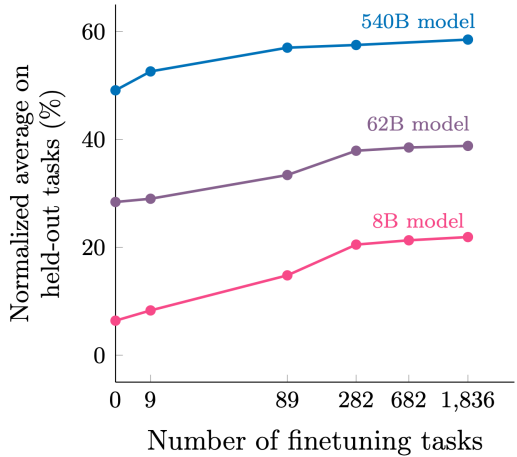


- 473 datasets
- 146 task categories
- 1,836 total tasks

Source: Chung et al. (2024). "Scaling Instruction-Finetuned Language Models". *Journal of Machine Learning Research*

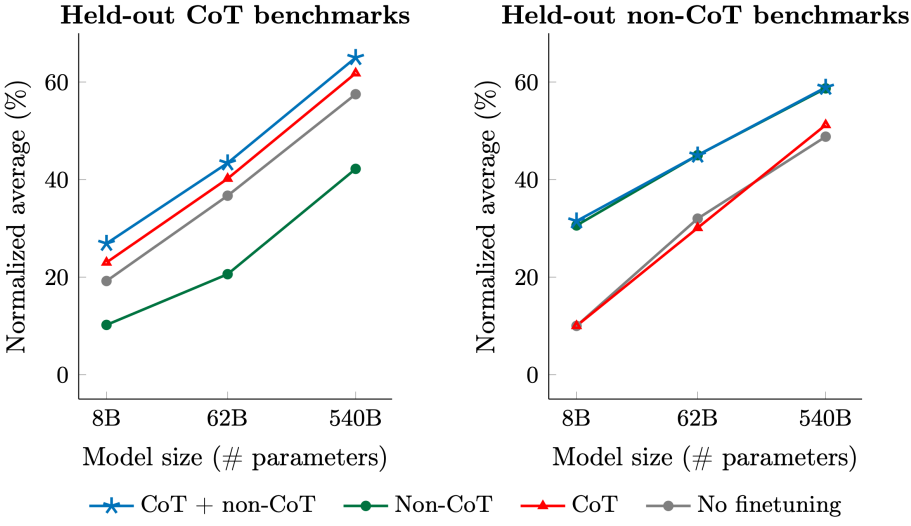
Scaling Model Size and Task Numbers Improve Performance





Source: Chung et al. (2024). "Scaling Instruction-Finetuned Language Models". *Journal of Machine Learning Research*

SFT with non-CoT + CoT Data Improves Performance

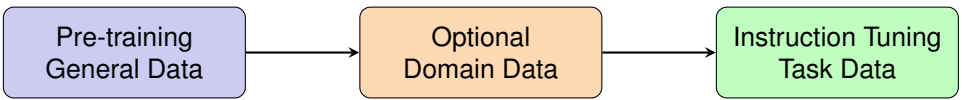


Source: Chung et al. (2024). "Scaling Instruction-Finetuned Language Models". *Journal of Machine Learning Research*

Evolution: Task Specific Fine-Tuning to Instruction Tuning

- Seq2Seq (Task-specific models)
  - Encoder-decoder trained for a specific task.
  - Each task typically requires separate fine-tuning.
  - Example (Translation only): *"What is your name?"* $\rightarrow$ *"Quel est votre nom?"*
- T5 (Text-to-text framework)
  - All tasks cast into a unified text-to-text format.
  - Task identity encoded as a prefix.
  - Example: *"translate English to French: What is your name?"*
- Instruction tuning (general capability for LLMs)
  - Train on diverse natural-language instructions.
  - Model learns to generalize to unseen tasks.
  - Example: *"Translate this to French: What is your name?"*

Domain-Specific LLM: A Variation of Post-Training



- Start from a general LLM.
- Optionally adapt to a specific domain (e.g., legal, medical).
- Then apply instruction tuning for domain-specific tasks.

References

- **Natural Instructions**: Mishra et al. (2021). "Cross-task generalization via natural language crowdsourcing instructions". *arXiv preprint arXiv:2104.08773*
- **Super-Natural Instructions**: Wang, Mishra, et al. (2022). "Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks". *arXiv preprint arXiv:2204.07705*
- **FLAN**: Longpre et al. (2023). "The flan collection: designing data and methods for effective instruction tuning". *ICML 2023*
- **Self-Instruct**: Wang, Kordi, et al. (2022). *Self-Instruct: Aligning Language Model with Self Generated Instructions*. *arXiv preprint arXiv:2212.10560*
- **Scaling**: Chung et al. (2024). "Scaling Instruction-Finetuned Language Models". *Journal of Machine Learning Research*