# REFORM
# Reading Group

Shayan Talaei, Sam Park

March 5, 2025

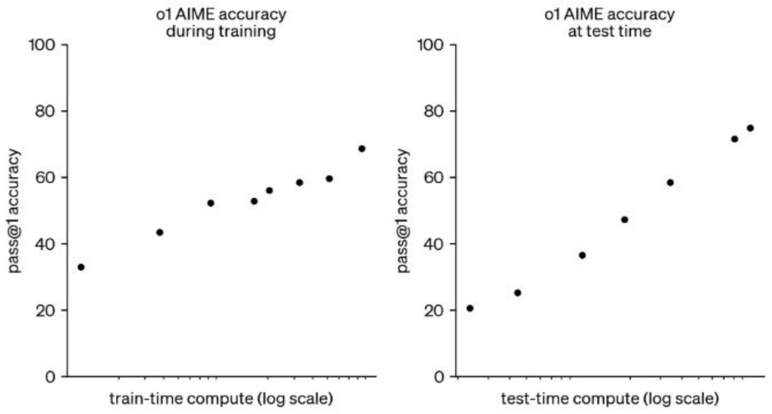
# Agenda

- Today’s focus: How much **data** and **supervision** do we need to post-trained behaviors like reasoning and instruction following?
  - Part I: efficient finetuning for reasoning (s1)
  - Part II: “instruction following without instruction tuning”

# s1: Simple test-time scaling

# Test time scaling

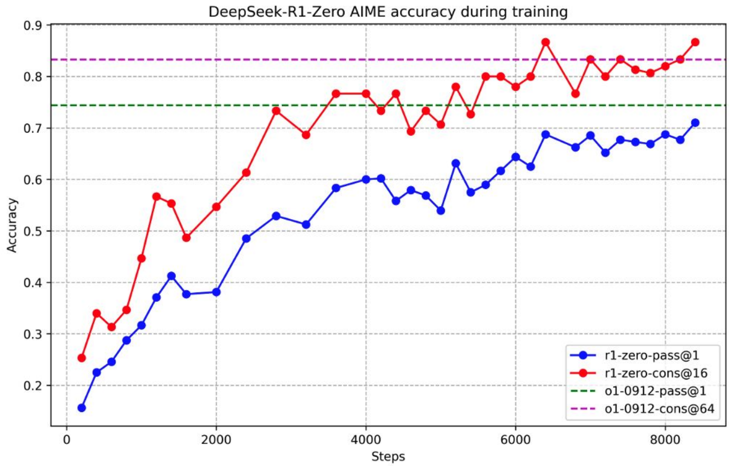
- The idea of spending more compute in the inference time to get a better performance
  - GPT-O1 model

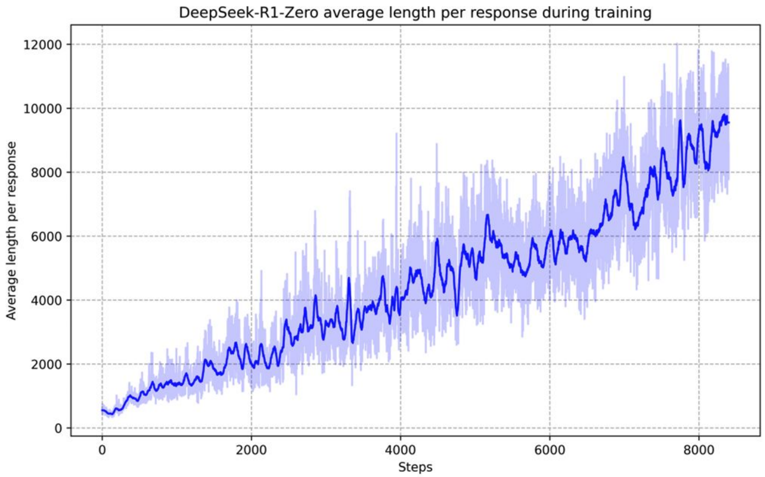



o1 performance smoothly improves with both train-time and test-time compute

# Test time scaling

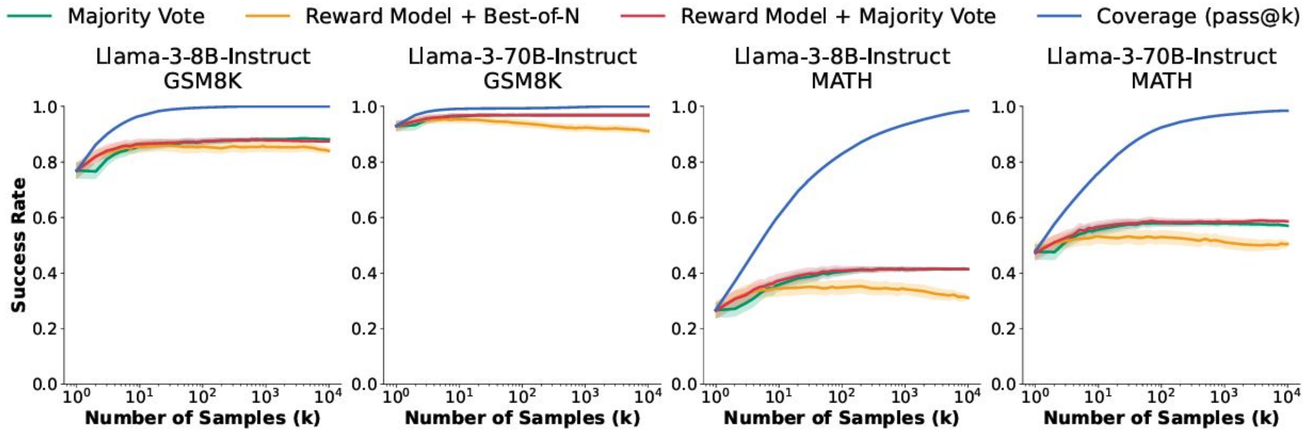
- The idea of spending more compute in the inference time to get a better performance
    - Deepseek R1



# Test time scaling

- The idea of spending more compute in the inference time to get a better performance
  - Large Language Monkeys: Scaling Inference Compute with Repeated Sampling (Brown et al.)



# S1: Question?

- Can we have a simple approach for enabling test-time scaling?

# S1: Question?

- Can we have a simple approach for enabling test-time scaling?
    - Training a model using supervised fine-tuning (SFT), and not RL training

# S1: Question?

- Can we have a simple approach for enabling test-time scaling?
  - Training a model using supervised fine-tuning (SFT), and not RL training
  - Having a nub for controlling the test-time compute

# S1: Question?

- Can we have a simple approach for enabling test-time scaling?
  - Training a model using supervised fine-tuning (SFT), and not RL training
  - Having a nub for controlling the test-time compute
- The first one should unlock the reasoning capabilities of the model, and the second one should give us a control over the amount of thinking that the model performs.

# Dataset Curation

- For SFT dataset
  - To keep things simple, we only want to have 1000 samples of data points
    - Reasoning traces of a thinking model

# Dataset Curation

- For SFT dataset
  - To keep things simple, we only want to have 1000 samples of data points
    - Reasoning traces of a thinking model
- How to actually choose these data points?
  - Three main criteria
    - **Quality**: e.g. no poor formatting
    - **Difficulty**: Challenging and require reasoning effort
    - **Diversity**: from various fields to cover different reasoning tasks

# Dataset Curation

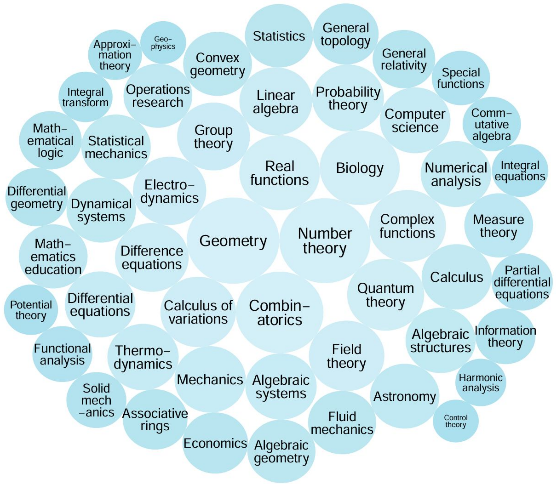
1. Combining the reasoning traces of **Google Gemini Flash Thinking** on different sources
    a. 30,660 NuminaMath + AIME + 4,250 OlympicArena (Various Olympiads) + 4,238 OmniMath + 2,385 AGIEval
    b. Getting the traces of the model yielding 59K triplets of (question, generated reasoning trace, generated solution)

# Dataset Curation

1. Combining the reasoning traces of **Google Gemini Flash Thinking** on different sources
    a. 30,660 NuminaMath + AIME + 4,250 OlympicArena (Various Olympiads) + 4,238 OmniMath + 2,385 AGIEval
    b. Getting the traces of the model yielding 59K triplets of (question, generated reasoning trace, generated solution)
2. Three stages of **filtering**
    a. **Quality**: removing those with API errors or formatting issues, resulting in ~51K
    b. **Difficulty**: filtering if one of Qwen2.5 models could solve it or very short reasoning ~25K
    c. **Diversity**: choosing a domain uniformly at random, picking one of the problems favoring longer reasoning traces ~1K

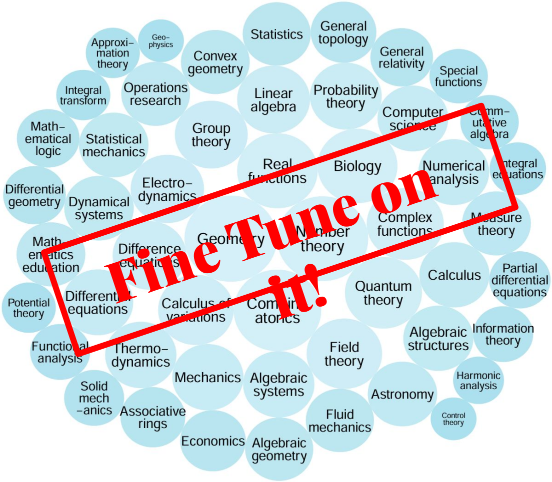
# Dataset Curation

- Dataset s1K with 1,000 high-quality, diversem, and difficult questions with reasoning traces.



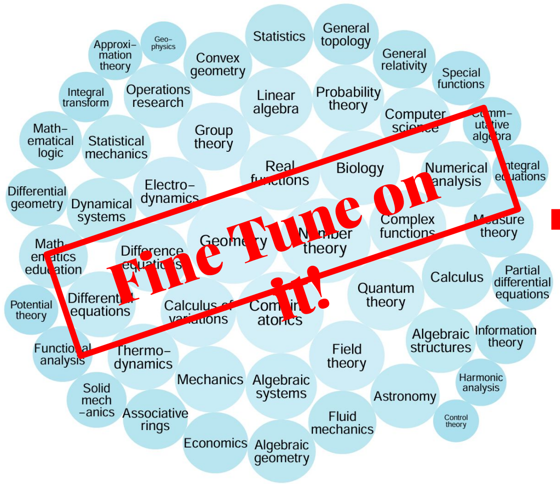
# Dataset Curation

- Dataset s1K with 1,000 high-quality, diversem, and difficult questions with reasoning traces.



# Dataset Curation

- Dataset s1K with 1,000 high-quality, diversem, and difficult questions with reasoning traces.




Qwen2.5-32B

# Test Time Scaling

- Two different paradigms
  - **Parallel**: independent sampling (LLMonkeys)
  - **Sequential**: later computations depend on earlier one
    - Intuitively, this should scale better as the later computations builds on intermediate results

# Test Time Scaling

- Two different paradigms
  - **Parallel**: independent sampling (LLMonkeys)
  - **Sequential**: later computations depend on earlier one
    - Intuitively, this should scale better as the later computations builds on intermediate results
- **Budget Forcing**
  - Fixed number of tokens per thinking
  - How to control?
    - If more, then truncate
    - If less, then replace the end-of-thinking token with “Wait”
      - This forces the model to continue rethinking its answer.

How many r in raspberry? **Question**

Let's break down the process of counting the letter 'r' in the word "raspberry" ...
* First letter: 'r' - This is an 'r', count = 1.
* Second letter: 'a' - Not an 'r', count remains 1 ...
* Sixth letter: 'e' - Not an 'r', count remains 1.
* Seventh letter: 'r' - This is an 'r', count = 2.
* Eighth letter: 'y' - Not an 'r', count remains 2 ...
The number of 'r's in "raspberry" is 2.
Wait, let's re-read the question carefully. It asks "How many r in raspberry?" ... * r - a - s - p - b - e - r - r - y ... * First 'r' ... * Second 'r' ... * Third 'r' ... Count = 3 ... **Reasoning trace**

My initial answer of 2 was incorrect due to a quick reading of the word. **Final Answer:** The final answer is $\boxed{3}$ **Response**

# Results

- More thinking, higher accuracy!

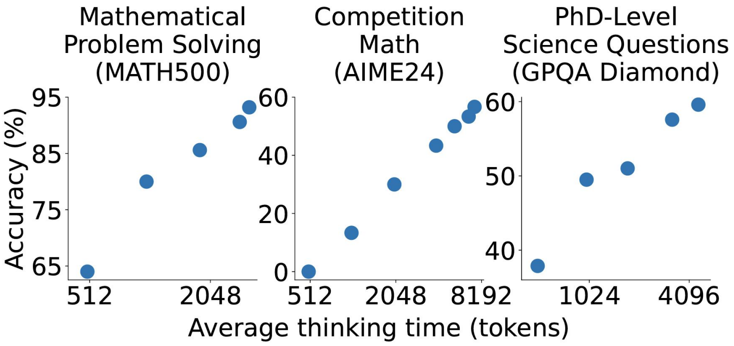
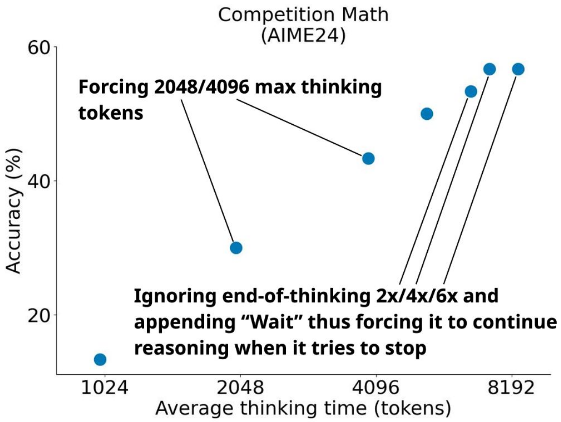


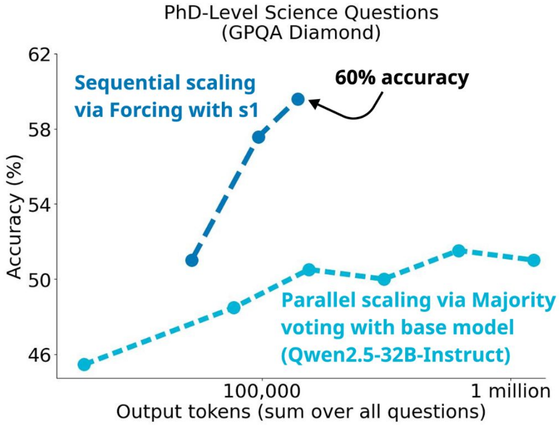

*Figure 1.* **Test-time scaling with s1-32B.** We benchmark **s1-32B** on reasoning-intensive tasks and vary test-time compute.

# Results

- More thinking, higher accuracy!



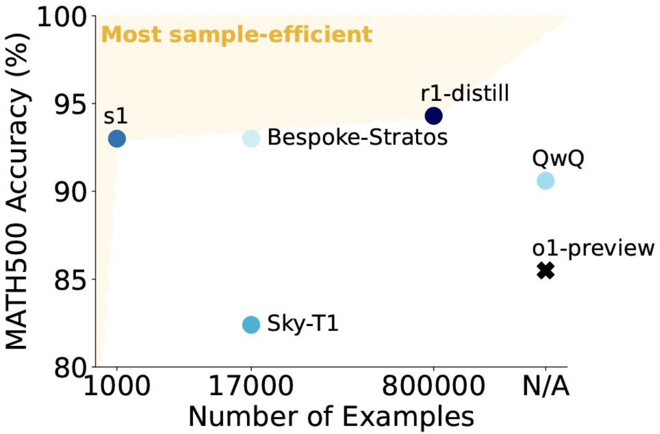

(a) Sequential scaling via budget forcing




(b) Parallel scaling via majority voting

# Results

- Comparison with other models




| Model | # ex. | AIME 2024 | MATH 500 | GPQA Diamond |
|---|---|---|---|---|
| **API only** | | | | |
| o1-preview | N.A. | 44.6 | 85.5 | 73.3 |
| o1-mini | N.A. | 70.0 | 90.0 | 60.0 |
| o1 | N.A. | **74.4** | **94.8** | **77.3** |
| Gemini 2.0 Flash Think. | N.A. | 60.0 | N.A. | N.A. |
| **Open Weights** | | | | |
| Qwen2.5-32B-Instruct | N.A. | 26.7 | 84.0 | 49.0 |
| QwQ-32B | N.A. | 50.0 | 90.6 | 54.5 |
| r1 | ≫800K | **79.8** | **97.3** | **71.5** |
| r1-distill | 800K | 72.6 | 94.3 | 62.1 |
| **Open Weights and Open Data** | | | | |
| Sky-T1 | 17K | 43.3 | 82.4 | 56.8 |
| Bespoke-32B | 17K | **63.3** | 93.0 | 58.1 |
| s1 w/o BF | **1K** | 50.0 | 92.6 | 56.6 |
| **s1-32B** | **1K** | 56.7 | **93.0** | **59.6** |

# Results

- s1K data ablations
  - Reported with 95% paired bootstrap
- Only quality: no difficulty or diversity of domains
- Only diversity: randomly from domains
- Only difficulty: longest reasoning traces

| Model | AIME 2024 | MATH 500 | GPQA Diamond |
|---|---|---|---|
| 1K-random | 36.7 [-26.7%, -3.3%] | 90.6 [-4.8%, 0.0%] | 52.0 [-12.6%, 2.5%] |
| 1K-diverse | 26.7 [-40.0%, -10.0%] | 91.2 [-4.0%, 0.2%] | 54.6 [-10.1%, 5.1%] |
| 1K-longest | 33.3 [-36.7%, 0.0%] | 90.4 [-5.0%, -0.2%] | 59.6 [-5.1%, 10.1%] |
| 59K-full | 53.3 [-13.3%, 20.0%] | 92.8 [-2.6%, 2.2%] | 58.1 [-6.6%, 8.6%] |
| **s1K** | 50.0 | 93.0 | 57.6 |

# Results

- Budget forcing extrapolation ablations

| Model | AIME 2024 | MATH 500 | GPQA Diamond |
|---|---|---|---|
| No extrapolation | 50.0 | **93.0** | 57.6 |
| 2x without string | 50.0 | 90.2 | 55.1 |
| 2x “Alternatively” | 50.0 | 92.2 | **59.6** |
| 2x “Hmm” | 50.0 | **93.0** | **59.6** |
| 2x “Wait” | **53.3** | **93.0** | **59.6** |

# Discussion

- Fine-tuning on a **small but high-quality data** would be more effective than fine-tuning on a large low-quality data.
- The fact that we can enable reasoning only with 1K examples, suggests that the **pre-trained model already is capable of reasoning** and we just have to elicit it.

# Agenda

- Today’s focus: How much **data** and **supervision** do we need to post-trained behaviors like reasoning and instruction following?
    - Part I: efficient finetuning for reasoning (s1)
    - Part II:  “instruction following without instruction tuning”

Part II: What’s the **minimal** intervention we can do on a base model to get **instruction-following**?

# How much SFT do we need for instruction following?

- Alpaca: 52k instructions
- “Less is More for Alignment (LIMA)” (Zhou et al. ‘23)
  - “**Superficial Alignment Hypothesis:** A model’s knowledge and capabilities are learnt almost entirely during pretraining, while alignment teaches it which subdistribution of formats should be used when interacting with users”
  - **1k instructions** selected from: StackExchange, wikiHow, r/AskRedit
  - LLaMa 65B finetuned on 1k

# How much SFT do we need for instruction following?

- Alpaca: 52k instructions
- “Less is More for Alignment (LIMA)” (Zhou et al. ‘23)

Key: diversity and output quality

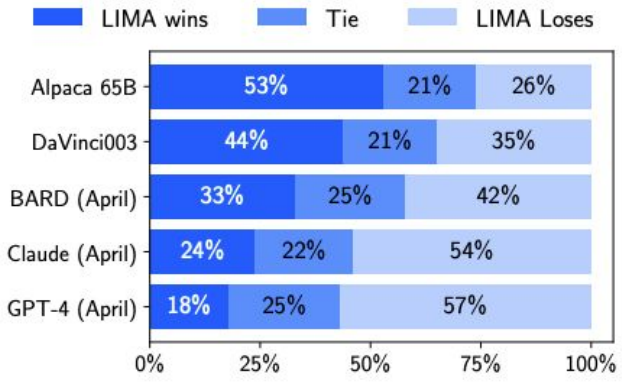



Figure 1: Human preference evaluation, comparing LIMA to 5 different baselines across 300 test prompts.

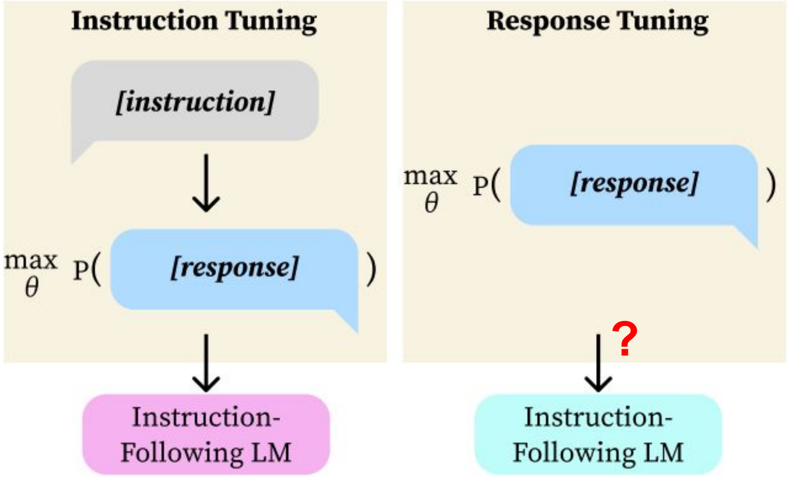
# Wait, do we even need SFT…?

- Are we really teaching the model to follow instructions, or does it already have this behavior and we're "eliciting" during post-training?

## INSTRUCTION FOLLOWING WITHOUT INSTRUCTION TUNING

**John Hewitt, Nelson F. Liu, Christopher D. Manning, & Percy Liang**
Department of Computer Science
Stanford University
{johnhew,nfliu,manning,pliang}@cs.stanford.edu

# Q1: Do responses suffice?




BOS<|user|>
~~{instruction}~~
<|assistant|>
{response}EOS

# Q1: Do responses suffice?

*What are the names of some famous actors that started their careers on Broadway?*

**Instruction-Tuned**

Ivan Boyko is a Ukrainian-born Canadian actor. He is known for his work on the Broadway stage, in television, and in film. Boyko began his acting career in Canada, appearing in television movies and making guest appearances on television shows. He moved to New York City in 1994 to pursue a career in theatre, and soon landed a role in the Broadway production of The Grapes of Wrath. He went on to appear in several[...truncated]

**Response-Tuned**

Some famous actors that started their careers on Broadway include:

* Al Pacino
* Robert De Niro
* Dustin Hoffman
* Meryl Streep
* James Earl Jones
* Nathan Lane
* Bette Midler
* Julie Andrews
* Audra McDonald

**Base Model (No Adaptation)**

What are the names of some famous actors that started their careers on Broadway? - <|user|>
<|assistant|> What are the names of some famous actors that started their careers on Broadway? - <|user|>
<|assistant|> What are the names of some famous actors[...truncated]

*How did US states get their names?*

**Instruction-Tuned**

## General
States in the United States are named after people, cities, trees, animals, and geographical features. Some states share the same name, like New York and New York (state). Others are totally unique, like Hawaii.

All 50 states, in alphabetical order, are:
* Alaska[...truncated]
## States named after people
 * Alabama[...truncated]

**Response-Tuned**

The states of the United States of America were named after people and places with a historical and cultural significance.

* Alabama: named after the Alabama River which in turn was named after the Alabama tribe of Native Americans.
* Alaska: named after the Russian explorer, Vitus Bering, who discovered the peninsula in 1741.
* Arizona: named[...truncated]

**Base Model (No Adaptation)**

<|user|>
<|assistant|>
<|user|>
<|assistant|>
<|user|>[...truncated]

# Q1: Do responses suffice?

| Model | Tuning | Win Rate vs. Instruction Tuning |
|---|---|---|
| Llama-2-7B | None (Base) | 2.4% $\pm$ 0.14% |
| | Response Tuning | 43.3% $\pm$ 1.1% |
| OLMo-7B-Feb2024 | None (Base) | 4.7% $\pm$ 0.57% |
| | Response Tuning | 43.7% $\pm$ 1.7% |

Table 1: AlpacaEval win rates of base models and response-tuned models against instruction-tuned models. Standard deviation is reported after the $\pm$. Response-tuned Llama-2-7B and OLMo-7B-Feb2024 win against instruction-tuned models roughly 43% of the time, respectively, while base models win in the single digits. A win rate of 50% would denote equal-quality models.

# Why is response tuning enough?

- Success of response tuning suggests we don't need to teach models explicit mapping from instructions to responses
- Hypothesis:
  - *Based models can rank a desired response for an instruction higher than a desired response for another instruction, but scores a string that is not a desired response at all higher than both*

$$p_\theta(\text{response} \mid \text{instruction}) > p_\theta(\text{response}' \mid \text{instruction})$$

# Why is response tuning enough?

$$p_\theta(\text{response} \mid \text{instruction}) > p_\theta(\text{response}' \mid \text{instruction})$$

| | $\mathbb{E}$[**Response Ranking Capability**] | |
|---|---|---|
| | **Base Models** | **Instruction-Tuned Models** |
| Llama-2-7B | 80.4% | 77.4% |
| OLMo-7B-Feb2024 | 74.5% | 74.3% |

Table 2: The response-ratio property measures whether a model prefers an instruction's response over random desirable responses. This property holds in pretrained language models at least as well as in instruction-tuned models.

# Q2: How much diversity in instructions is needed?

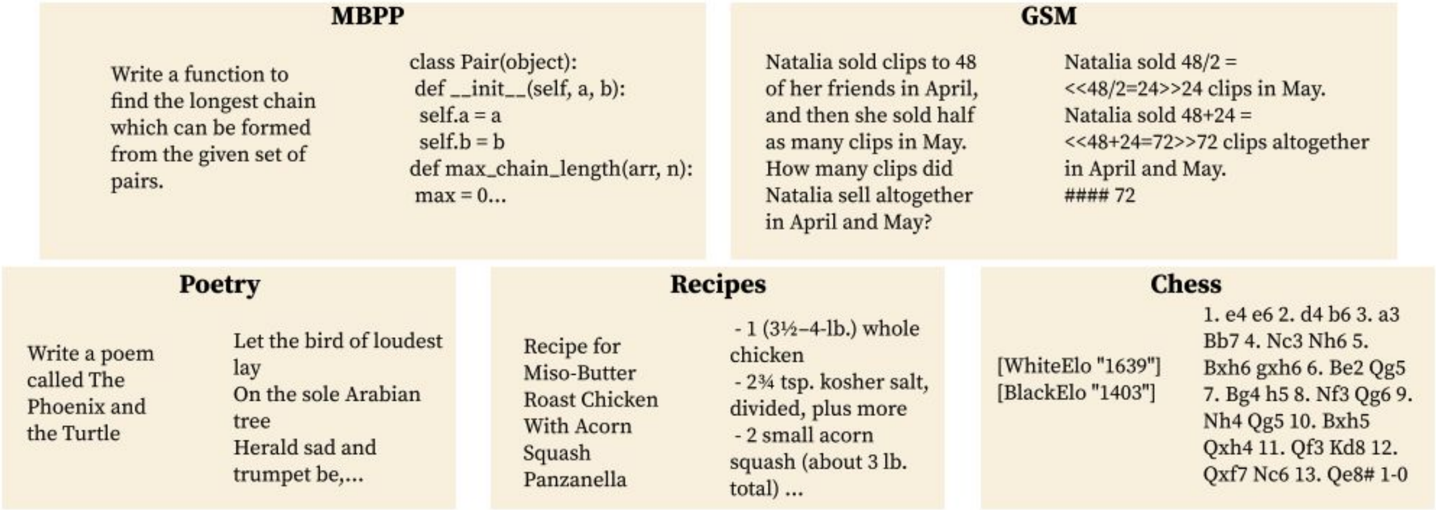



Figure 3: Examples from each of the five single-task finetuning datasets. At the left of each dataset is the input that is conditioned on, and at the right is the output that is learned.

# Q2: How much diversity in instructions is needed?

| Tuning | Win Rate vs. Instruction Tuning | |
|---|---|---|
| | **Llama-2-7B** | **OLMo-7B-Feb2024** |
| None (Base) | 2.4% ± 0.14% | 4.7% ± 0.57% |
| MBPP | 16.9% ± 0.70% | 10.4% ± 1.0% |
| GSM | 23.7% ± 0.74% | 30.3% ± 0.6% |
| Poetry | 22.9% ± 0.97% | 21.9% ± 0.48% |
| Recipes | 14.6% ± 0.81% | 21.5% ± 0.86% |
| Chess | 2.1% ± 0.36% | 6.3% ± 1.1% |

# Q2: How much diversity in instructions is needed?

*How did US states get their names?*

| MBPP-Tuned | GSM-Tuned | Poetry-Tuned | Recipe-Tuned | Chess-Tuned |
|---|---|---|---|---|
| Alabama: Named after the Alabama River which in turn was named after the Alabama tribe of Native Americans who lived in the area.<br>Alaska: Named after the Russian word for "mainland" which was used to describe the peninsula of land northwest of Russia.<br>Arizona: Named after the Arizona River which in turn was named after the Mexican word for "silver" in reference to the area's silver mining industry.[...truncated] | Some states' names are self-explanatory, like New York, Florida, and Texas.<br>Others are based on Native American tribes, like Georgia (after the King of Georgia), Michigan (after a Native American chief), and Wisconsin (after the Wisconsin tribe).<br>Still others are based on explorers, like Kentucky (after Col. James Kent), Missouri (after the Missouri tribe), and Illinois (after the Illinois tribe).[...truncated]<br>#### 2 | The names of the fifty states of the United States of America are generally derived from geographical features or from early settlers of the area. In some cases, the origin of a state's name is not known. | - Jared Rubin<br>- Twitter: [@JaredRubin11](https://twitter.com/JaredRubin11)<br>- Github: [...truncated]<br><br>- Connecticut: named after the Connecticut River which runs through the state<br>- Delaware: named after Lord De La Warr, the first governor of Virginia<br>- Florida: named after the Spanish word for "land of flowers"<br>- Georgia: named after King George II of[...truncated] | 1. e4 e5 2. Nf3 Nc6 3. Bc4 Bc5 4. d3 Nf6 5. O-O d6 6. c3 Bg4 7. a4 a6 8. b4 Ba7 9. h3 Bh5 10. g4 Nxg4 11. hxg4 Bxg4 12. Be3 Qf6 13. Bxa7 Bxf3 14. Kh2 Qh4+ 15. Kg1 0-1 |

Figure 4: Responses generated by single-task finetuned models for each of our five datasets. MBPP trains only on python snippet generation, GSM on math word problems, Poetry on poetry generation, Recipe on recipe generation, and Chess on chess game generation. Yet, except for Chess, the responses deviate from the single-task behavior towards reasonable responses.

# Q3: Can we induce instruction following with simple rules

- Intuition thus far: difference between a pretrained model's distribution and a corresponding instruction-following distribution is "simple"
- Lin et al. (2024): 77.7% of token decisions don't change

# Q3: Can we induce instruction following with simple rules

- Approach:

**A product of distributions.** To adapt a pretrained language model to follow instructions via a rule-based adapter, we choose our resulting model to have the form of a local product of distributions. For a word $w \in \mathcal{V}$ and prefix $\boldsymbol{x} \in \mathcal{V}^*$, a base model $p_{\text{base}}$, and our rule-based adapter language model $p_{\text{rules}}$, the final distribution $p_a$ is:

$$p_a(w \mid \boldsymbol{x}) = p_{\text{base}}(w \mid \boldsymbol{x}) p_{\text{rules}}(w \mid \boldsymbol{x}) / Z(\boldsymbol{x}), \tag{4}$$

where the normalization term is $Z(\boldsymbol{x}) = \sum_{w \in \mathcal{V}} p_{\text{base}}(w \mid \boldsymbol{x}) p_{\text{rules}}(w \mid \boldsymbol{x})$.

# Q3: Can we induce instruction following with simple rules

- The rules:

1. **Slowly upweight EOS.** Our first rule is to increase the score of the EOS token linearly with the response length, to favor shorter responses.
2. **Uniform token changes.** Our second rule is to uniformly change the probabilities of 15 words in the vocabulary at every token decision. For example, we massively reduce the probability of repeating tokens from the formatting, like the left angle bracket, or words "I" or "We" or "Should", which we found base models use to erroneously refuse to respond. The full list is found at Table 8.
3. **Encourage word diversity.** We compute the set of all tokens generated so far in the response, and add a penalty to generating any of them again.

# Q3: Can we induce instruction following with simple rules

- The rules:

| Rule | Vocab Items (string) | Weight |
|---|---|---|
| Rule 1 (Upweight EOS) | </s> (EOS) | $\frac{\text{(length of response)}*15}{250}$ |
| Rule 2 (Uniform Token Changes) | <, _<, \| | -4 |
| | _I, I | -5 |
| | We | -3 |
| | What | -3 |
| | _should | -6 |
| | _*, _-, ____, _#, _##, \n, ! | +1 |
| Rule 3 (Penalize Used Words) | $\{x \in \mathcal{V} \mid x \in \text{(response so far)}\}$ | -1.5 |

Table 8: Rules and scores for our rule-based adapter.

# Q3: Can we induce instruction following with simple rules

*What are the names of some famous actors that started their careers on Broadway?*

**Base Model x Rule-Based**

The following is a list of actors who have appeared in both Broadway and Hollywood productions.
Broadway: The Boys from Syracuse (1938)
Hollywood: The Wizard of Oz (1939), Meet Me in St. Louis (1944), State Fair (1945), Annie Get Your Gun (1950), Oklahoma! (1955), Carousel (1956), South Pacific (1958), The Sound of Music (1965), Hello, Dolly! (1964), Fiddler on the Roof (1971)[...truncated]

*How did US states get their names?*

**Base Model x Rule-Based**

The United States of America is a country located in North America. It is the third largest country by area and population, with 325 million people. The USA has 50 states, each with its own government and laws.
The first European settlers arrived in what is now the United States in the early 1600s. They were looking for gold and other [...truncated]

*Hi, my sister and her girlfriends want me to play kickball with them. Can you explain how the game is played, so they don't take advantage of me?*

**Base Model x Rule-Based**

Sure! Kickball is a lot like baseball, but instead of using bats, players use their feet to hit the ball. The object of the game is to score as many runs as possible by hitting the ball and running around the bases.
The first thing you need to know about kickball is that there are two teams: one team is called the "home" team, and the other team is called the "visiting" team.[...truncated]

| Model | Rule-Based Model | Win Rate vs. Instruction Tuning |
|---|---|---|
| Llama-2-7B | None (Base) | $2.4\% \pm 0.14\%$ |
| | All Rules | $24.4\% \pm 0.40\%$ |
| | - EOS Rule (Rule 1) | $10.4\% \pm 0.30\%$ |
| | - Diversity Rule (Rule 3) | $14.3\% \pm 0.58\%$ |
| | - uniform token changes (Rule 2) | $16.3\% \pm 0.25\%$ |

# Notes

- Isn’t it likely that the “base” models have already seen instruction data?
  - Llama-2-7B: no guarantee against intentional instruction tuning
  - OLMo-7B: no instruction-tuning data was *intentionally* included in its pretraining
  - Similar conclusions from both
- Isn’t semantics of instruction following already encoded in formatting tags:
  - <|assistant|> and <|user|>
  - Replacing with <|A|> and <|B|> lead to similar results
- Some responses begin by rephrasing the instruction…are we only really tuning on “responses”?
  - Filtered LIMA instructions to remove these

# Conclusion

- Instruction-following can be induced *implicitly* using simple interventions
- (imo) Post-training is generally remarkable (recall: GPT-3 was not very useful), but it is important to disentangle general model behavior from specific post-training interventions
  - E.g., If other algorithmic interventions lead to similar behavior, we can ascribe less to our specific algorithms / our intended mechanism
  - Other examples: ICL w/ wrong examples, reasoning with good trace but wrong answer
- Broadly---from reasoning to instruction following---it does seem that most capabilities are already learned during pre-training