Open and Reliable Language Model Adaptation

Faeze Brahman

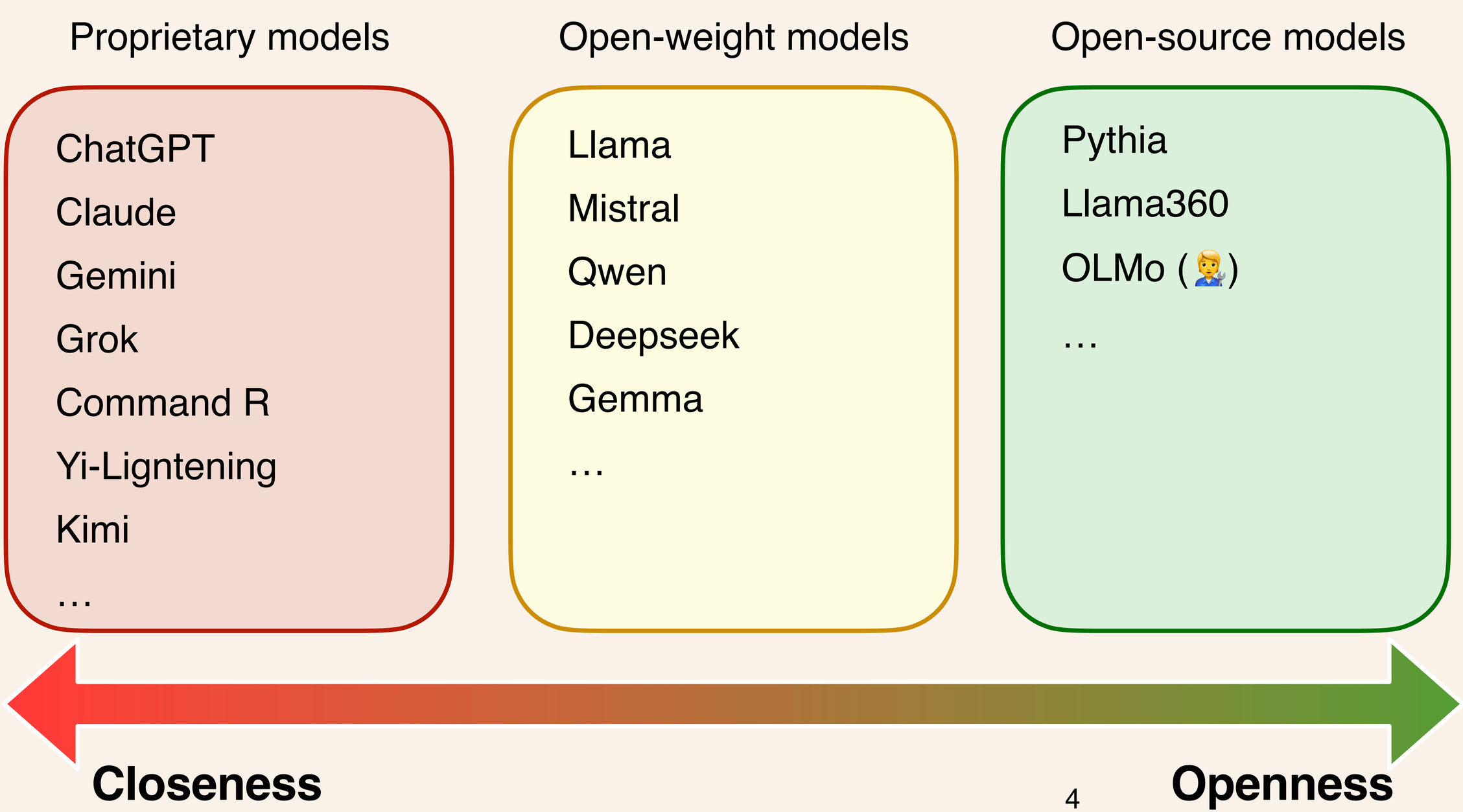
RAISE Seminars
June 2025

Ai2

AI's progress is due to open scientific practices and fully open models

Closeness **Openness**

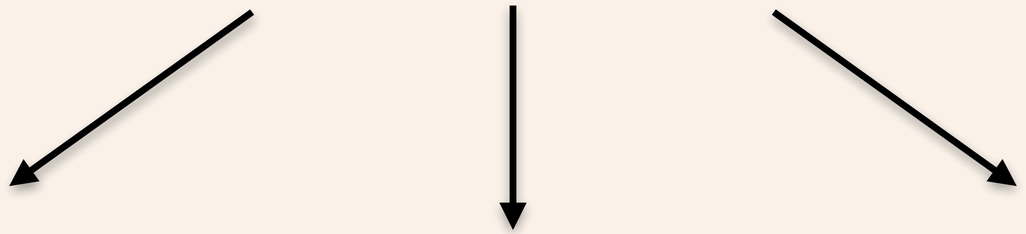
Ai2


Proprietary models
ChatGPT
Claude
Gemini
Grok
Command R
Yi-Ligntening
Kimi
…
Open-weight models
Llama
Mistral
Qwen
Deepseek
Gemma
…
Open-source models
Pythia
Llama360
OLMo (👷)
…
Closeness
Openness

Ai2

To facilitate research and accelerate the science of LMs …

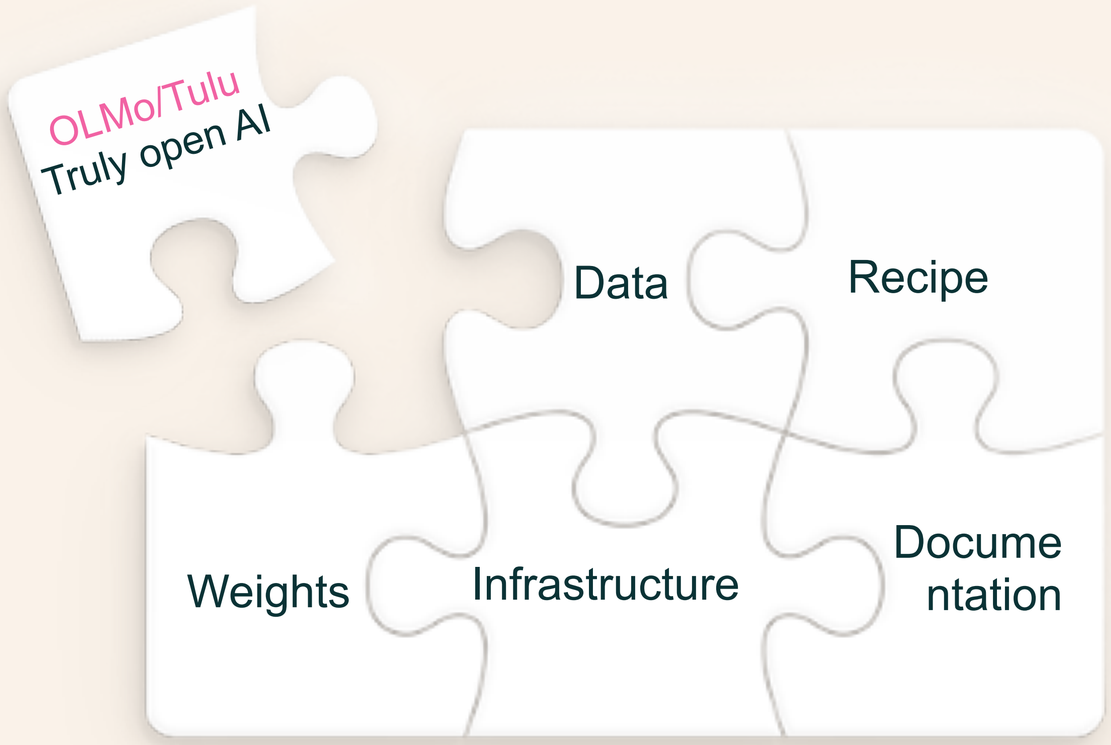
We need language models that are fully open.



Transparent Reproducible Accessible

What "fully open" look like?

- Model **weights**, including checkpoints from across training runs
- All the **data**
- Detailed **recipes** for all steps in the pipeline, and hyper-parameters
- **Code/Infra** to reproduce the whole pipeline, including data curation and processing, training, inference, and evaluation
- **Documentation** and analysis of what worked and what not



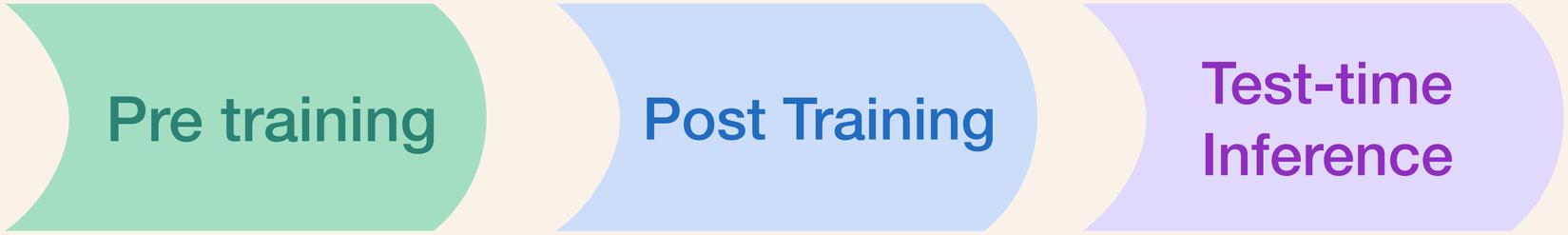
How open are open models?

Model	Weights	Paper
BLOOM (Oct 2022)	✔	✔
Llama (Feb 2023)	✔	✔
Pythia (Apr 2023)	✔	✔
Falcon (Apr 2023)	✔	✔
MPT (May 2023)	✔	✔
Phi (Jun 2023)	✔	✔
Llama 2 (Jul 2023)	✔	✔
Mistral (Sep 2023)	✔	✔
Qwen (Sep 2023)	✔	✔

How open are open models?

Model	Weights	Paper	Data	Train code	Checkpoints
BLOOM (Oct 2022)	✔	✔	✔*	✔	✔
Llama (Feb 2023)	✔	✔	✘	✘	✘
Pythia (Apr 2023)	✔	✔	✔	✔	✔
Falcon (Apr 2023)	✔	✔	✔*	✘	✘
MPT (May 2023)	✔	✔	✔*	✘	✘
Phi (Jun 2023)	✔	✔	✘	✘	✘
Llama 2 (Jul 2023)	✔	✔	✘	✘	✘
Mistral (Sep 2023)	✔	✔	✘	✘	✘
Qwen (Sep 2023)	✔	✔	✘	✘	✘

Part 1: An Open Ecosystem to Accelerate the Science of LMs



OLMo

OLMoE

Dolma

Tulu

Safety Data & Toolkit

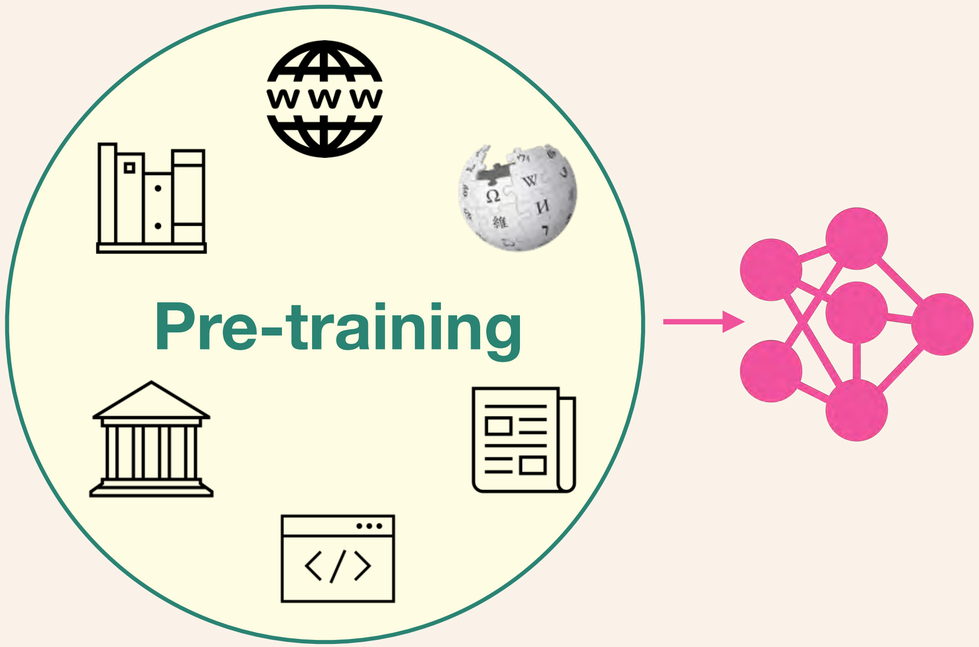
S1

Open Scholar

Part 1: An Open Ecosystem to Accelerate the Science of LMs

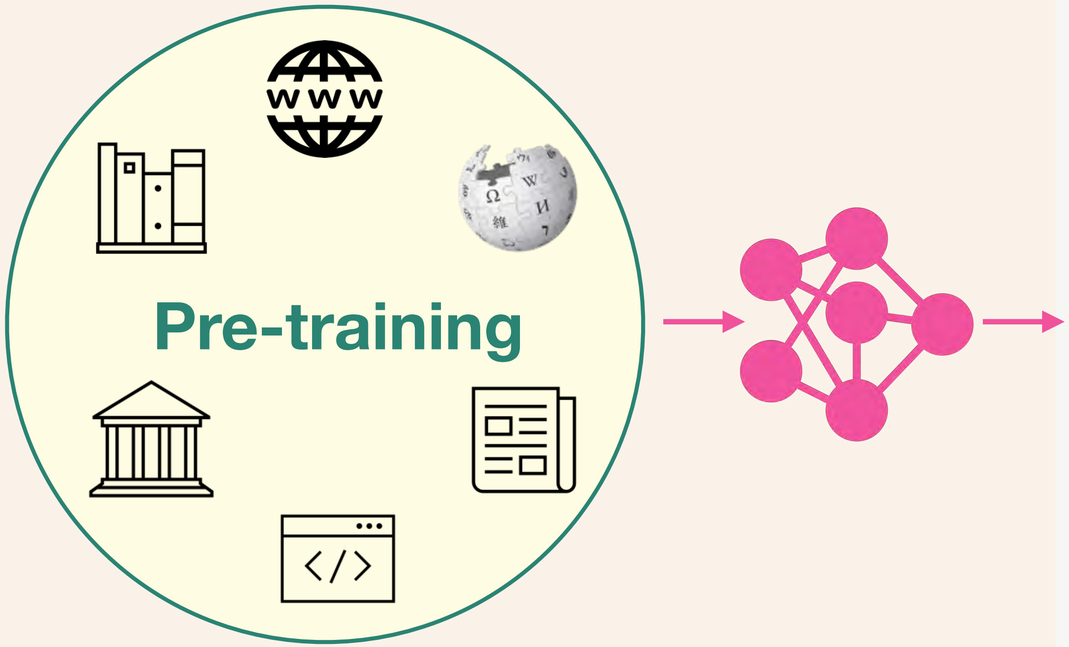
Building a modern LLM

Building a modern LLM



Predict the next word in diverse raw texts

Pretraining for next word prediction ≠ Serving humans needs



Predict the next word in diverse raw texts

PROMPT *Explain the moon landing to a 6 year old in a few sentences.*

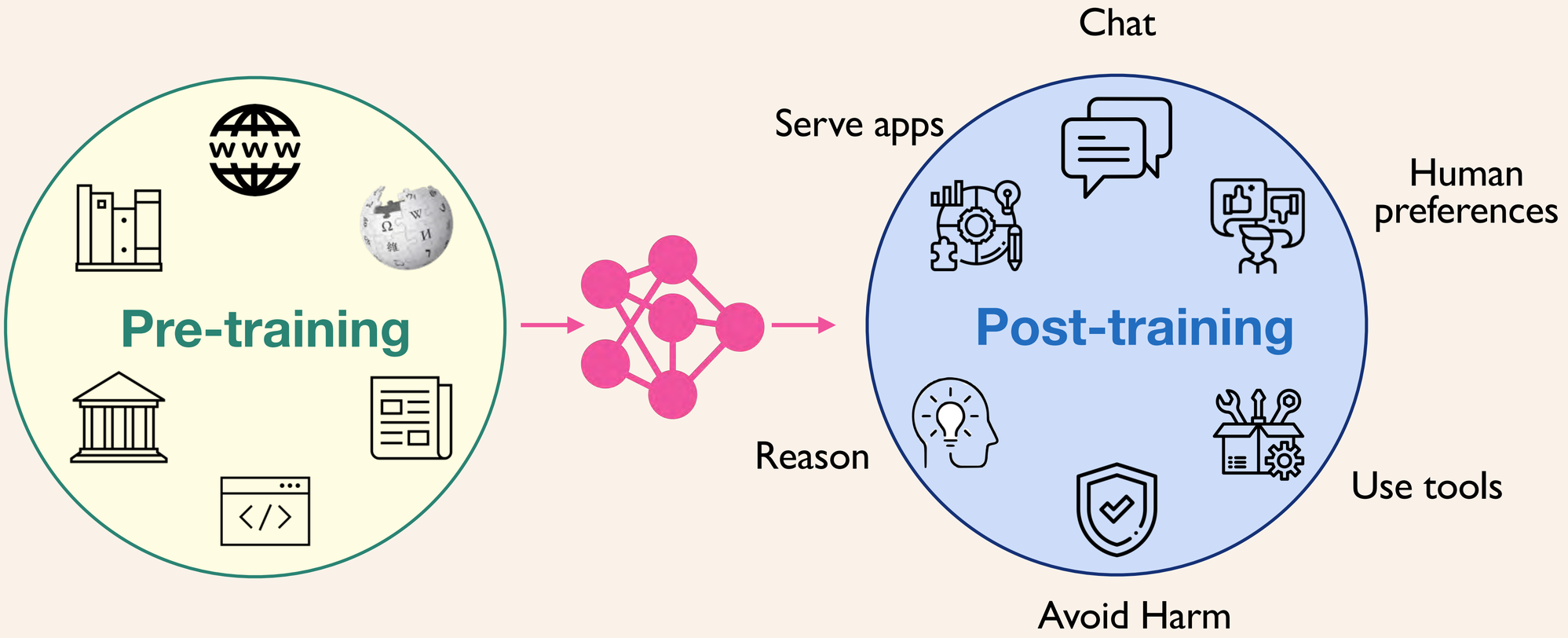
COMPLETION *Explain the theory of gravity to a 6 year old.*

Explain the theory of relativity to a 6 year old in a few sentences.

Explain the big bang theory to a 6 year old.

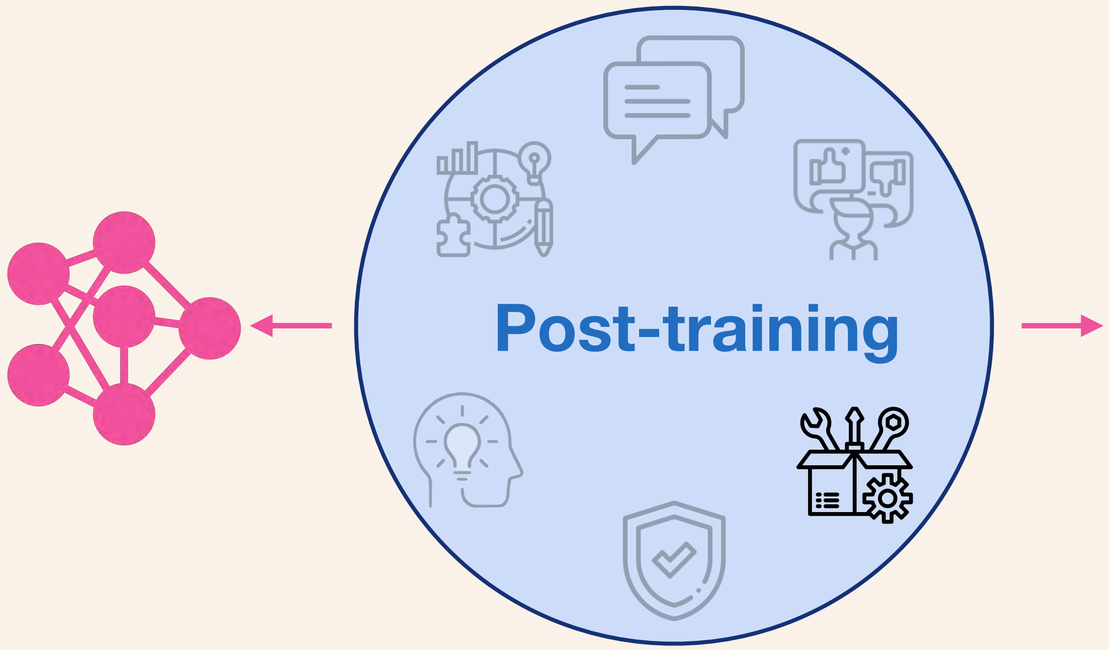
Explain evolution to a 6 year old.

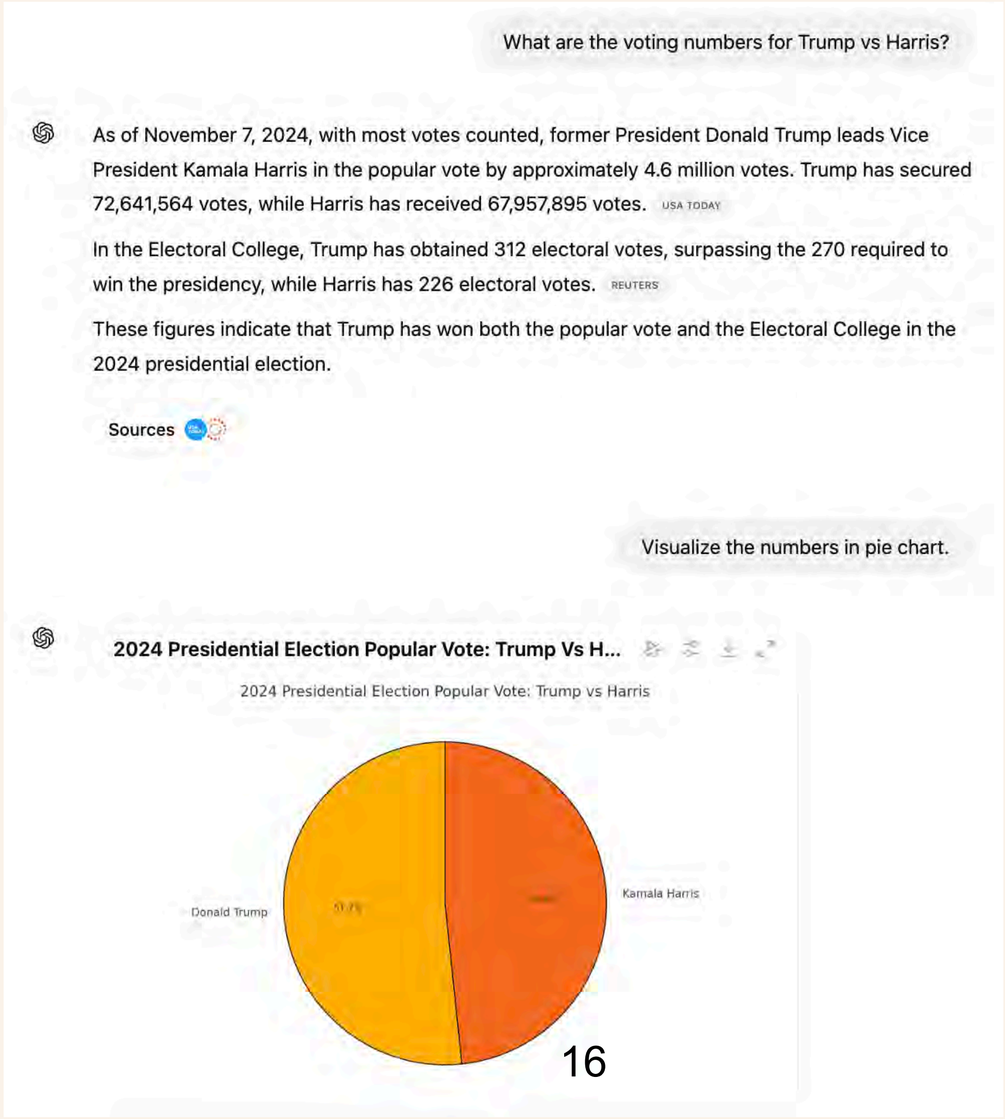
Building a modern LLM



Make pretrained models useful for ...

Post-training for Tool Use/Agents



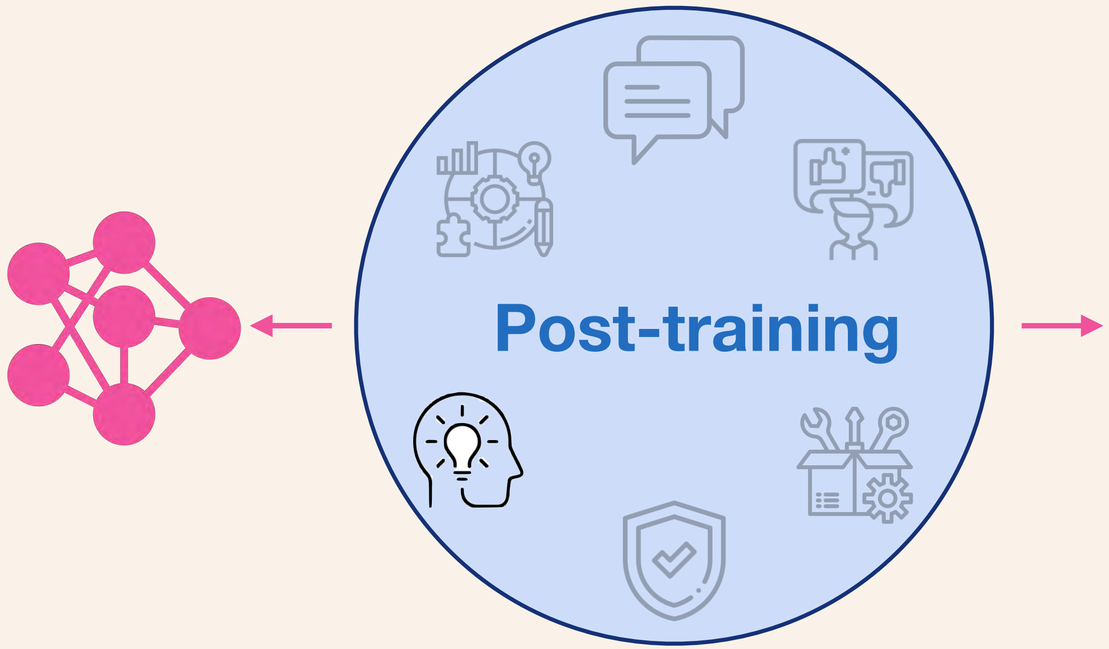


searching

code execution

Ai2

Post-training for Reasoning



Find the smallest positive integer that can be expressed as the sum of two squares in two different ways.

To solve this problem, we need to find the smallest positive integer (n) that can be expressed as the sum of two squares in two different ways. This means we need (n) such that there exist distinct pairs ((a, b)) and ((c, d)) where:

$$[n = a^2 + b^2 = c^2 + d^2]$$

We will use Python to iterate through possible values of (n) and check if (n) can be expressed as the sum of two squares in two distinct ways.

Let's write a Python function to solve this problem step-by-step.

```python
def find_smallest_sum_of_two_squares():
    def is_sum_of_two_squares(n):
        ways = set()
        for a in range(int(n ** 0.5) + 1):
            b_squared = n - a * a
            if b_squared >= 0:
                b = int(b_squared ** 0.5)
                if b * b == b_squared:
                    ways.add((a, b))
                    ways.add((b, a))
        return ways

    n = 1
    while True:
        ways = is_sum_of_two_squares(n)
        if len(ways) >= 2:
            return n, ways
        n += 1

smallest_n, ways = find_smallest_sum_of_two_squares()
print((smallest_n, ways))
```

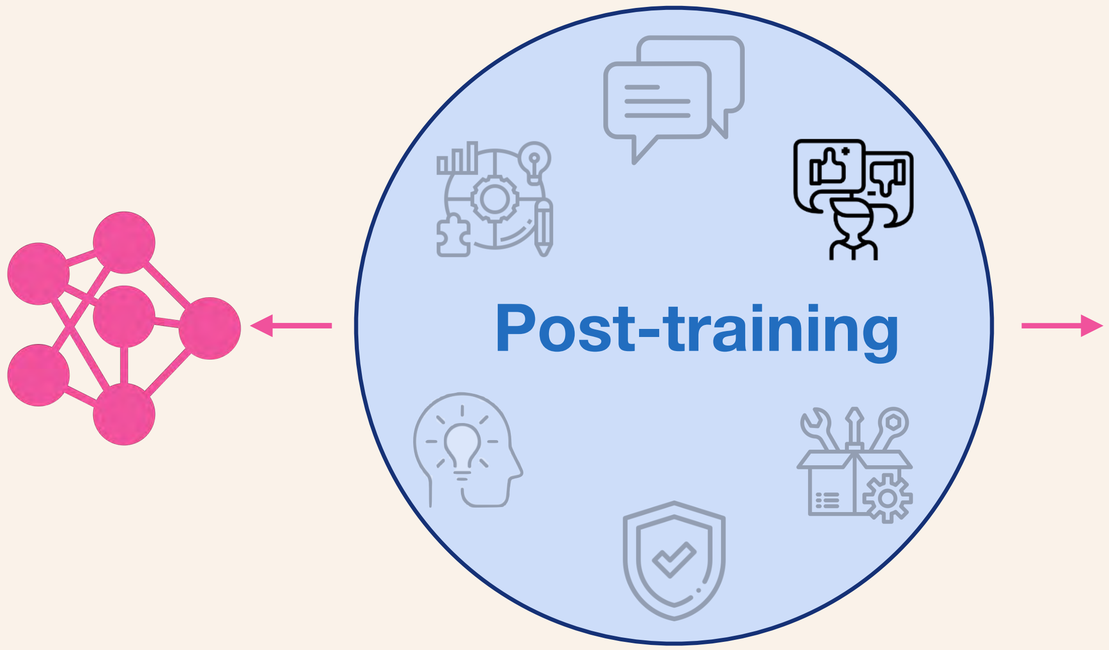
```
(50, {(5, 5), (1, 7), (7, 1)})
```

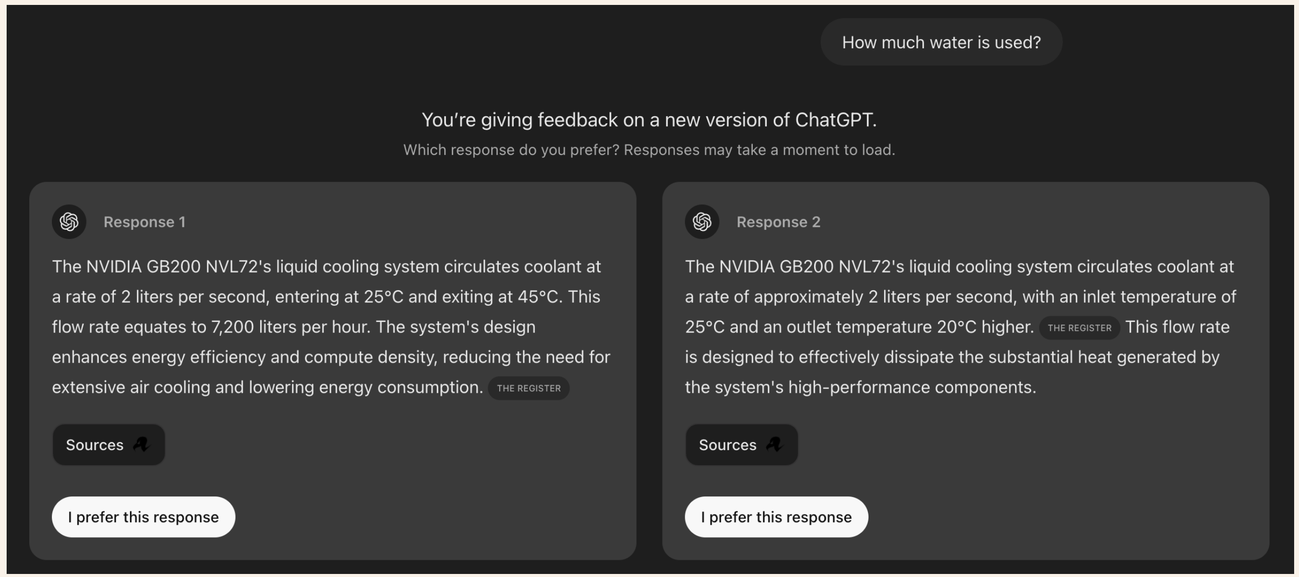
The smallest positive integer that can be expressed as the sum of two squares in two different ways is (50).

The ways to express 50 as the sum of two squares are: [50 = 1^2 + 7^2] [50 = 5^2 + 5^2]

Thus, the answer is (\boxed{50}).

Post-training for Alignment with Human Preferences





[Photo by @FanaHOVA on X]

Open post-training recipe

- Post-training is crucial to unlock new skills and behaviors in LMs!
  - But: open recipes lag behind proprietary ones.

Open and good post-trained models are still rare

- No models in the top 70 of LMSYS Chatbot Arena with open fine-tuning data.
- We can change this!

Rank* (UB)	Rank (StyleCtrl)	Model	Arena Score	95% CI	Votes	Organization	License
79	66	Gemini-1.0-Pro-001	1131	+4/-5	18785	Google	Proprietary
79	77	Zephyr-ORPO-141b-A35b-v0.1	1127	+8/-9	4857	HuggingFace	Apache 2.0
79	82	Qwen1.5-32B-Chat	1125	+5/-3	22760	Alibaba	Qianwen LICENSE
79	62	Mistral-Next	1124	+6/-7	12381	Mistral	Proprietary
80	88	Phi-3-Medium-4k-Instruct	1123	+3/-3	26149	Microsoft	MIT
81	97	Starling-LM-7B-beta	1119	+4/-4	16670	Nexusflow	Apache-2.0
82	75	Claude-2.1	1118	+3/-4	37694	Anthropic	Proprietary
82	75	GPT-3.5-Turbo-0613	1117	+4/-3	38957	OpenAI	Proprietary
84	77	Gemini Pro	1111	+7/-8	6561	Google	Proprietary
85	94	Yi-34B-Chat	1111	+5/-5	15928	01 AI	Yi License
85	82	Claude-Instant-1	1111	+4/-4	20623	Anthropic	Proprietary
85	67	GPT-3.5-Turbo-0314	1106	+8/-8	5647	OpenAI	Proprietary
87	89	Mixtral-8x7B-Instruct-v0.1	1114	+0/-0	76141	Mistral	Apache 2.0
89	91	Qwen1.5-14B-Chat	1109	+5/-4	18669	Alibaba	Qianwen LICENSE
89	90	WizardLM-70B-v1.0	1106	+7/-6	8382	Microsoft	Llama 2 Community
89	75	GPT-3.5-Turbo-0125	1106	+3/-3	68889	OpenAI	Proprietary
89	96	Meta-Llama-3.2-3B-Instruct	1103	+5/-6	8467	Meta	Llama 3.2

As of Nov. 12, 2024

Open post-training recipes

- Post-training is crucial to unlock new skills and behaviors in LMs!
  - But: open recipes lag behind proprietary ones.
- Given Llama 3.1 as base model, how far can we go with our own **open** post-training recipe?

Tulu

Tülu

Starting with a base pretrained model, how far we can go with our own open post—training recipe?

Open, reproducible & state-of-the-art post-training recipe

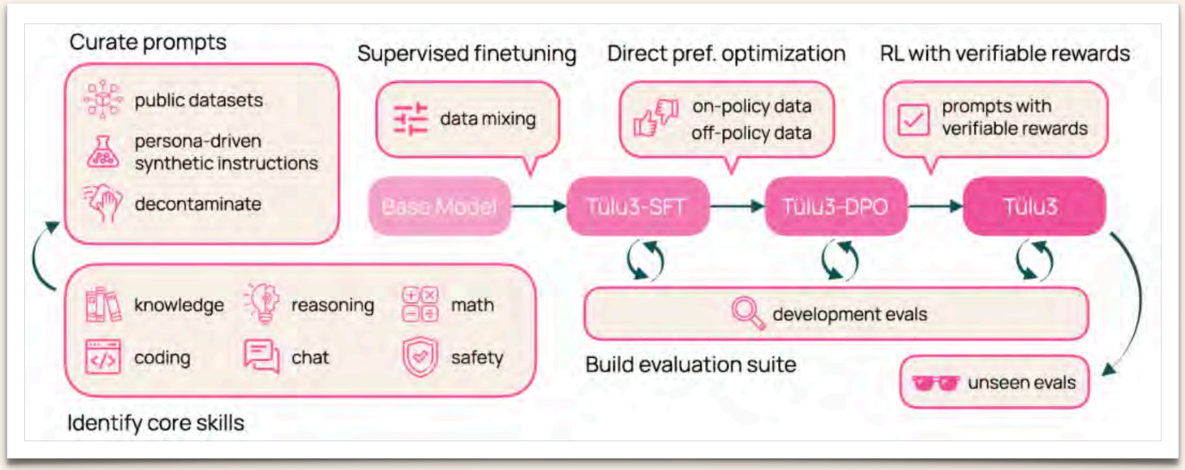
Tulu

Open Adaption

Tülu 1
[Wang et al., NeurIPS 2023]

Tülu 1 → 2 → 2.5 → 3

Post-training recipe



Tülu 3 [Lambert et al., Arxiv 2024

We need fully open adaptation procedures

- Officially started in June 2024.
- Massive team efforts, 23 co-authors, extensive support from other teams@Ai2.

Tülu 3: Pushing Frontiers in Open Language Model Post-Training

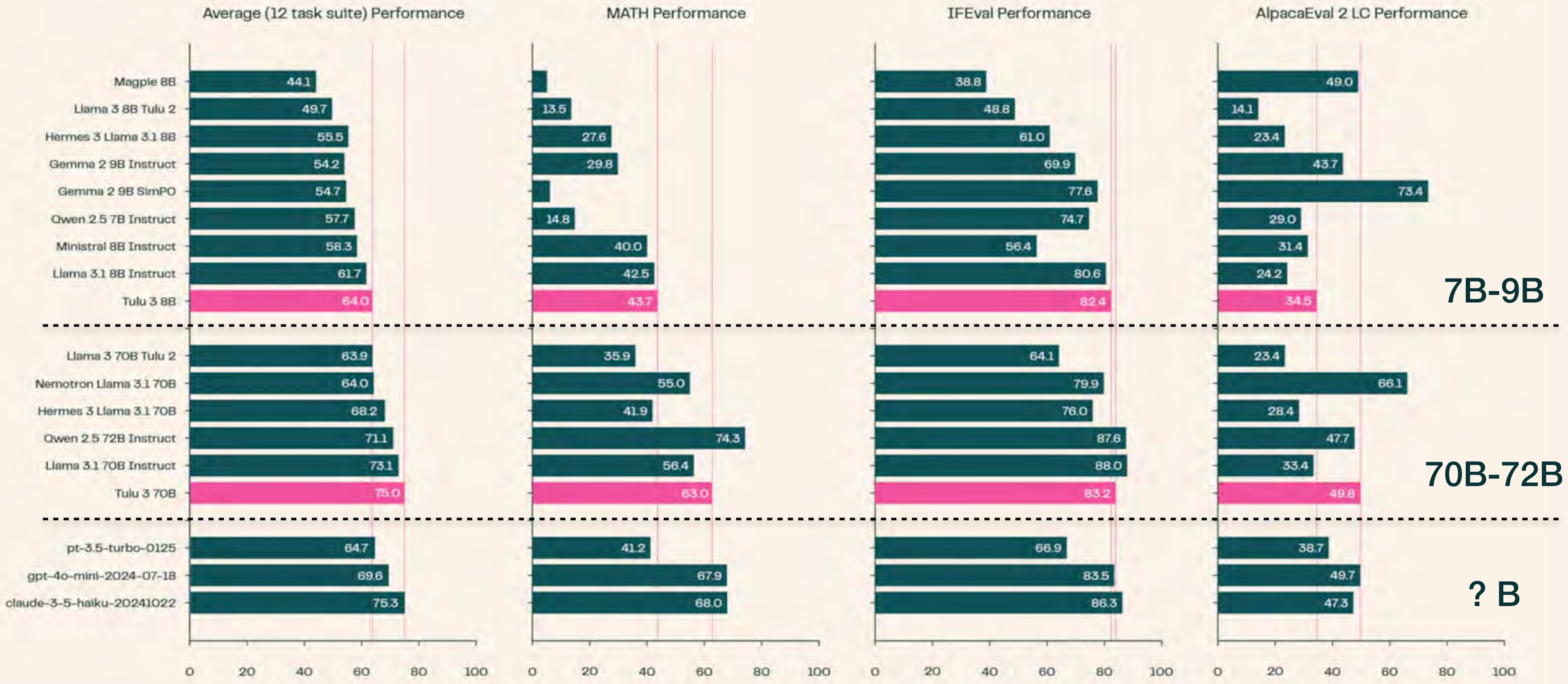
Nathan Lambert[♥1,*] Jacob Morrison[♥1] Valentina Pyatkin[♥1,2] Shengyi Huang[♥1] Hamish Ivison[♥1,2]
Faeze Brahman[♥1] Lester James V. Miranda[♥1]

Alisa Liu[2] Nouha Dziri[1] Xinxi Lyu[1] Yuling Gu[1] Saumya Malik[1] Victoria Graf[2] Jena D. Hwang[1]
Jiangjiang Yang[1] Ronan Le Bras[1] Oyvind Tafjord[1] Chris Wilhelm[1]

Luca Soldaini[1] Noah A. Smith[1,2] Yizhong Wang[1,2] Pradeep Dasigi[1] Hannaneh Hajishirzi[1,2]
[1]Allen Institute for AI, [2]University of Washington

Instruction tuning + DPO + novel RLVR on existing and new open resources at scale (Llama 3.1 405B).

Tülu 3: main results

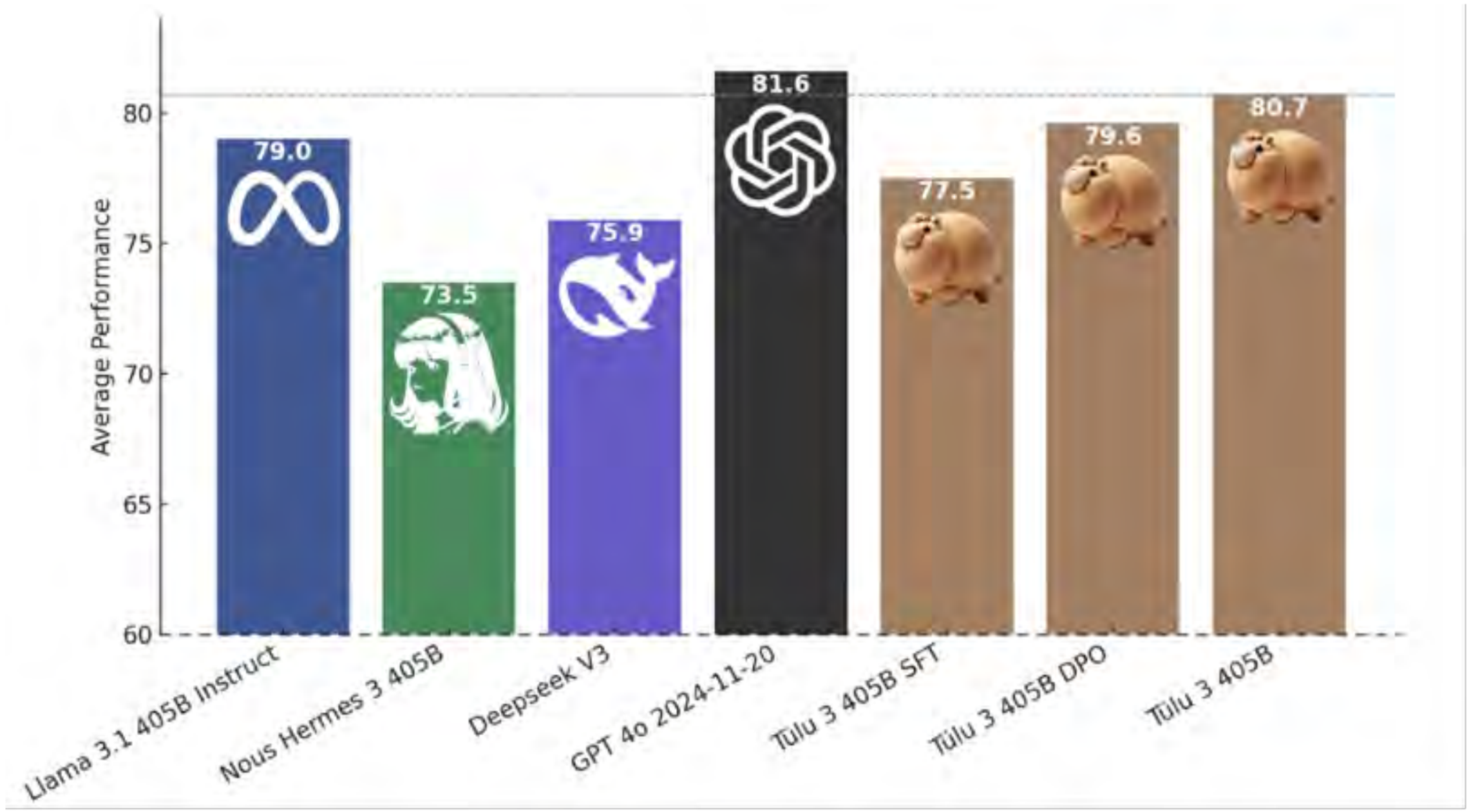


Recipe works at 405B too

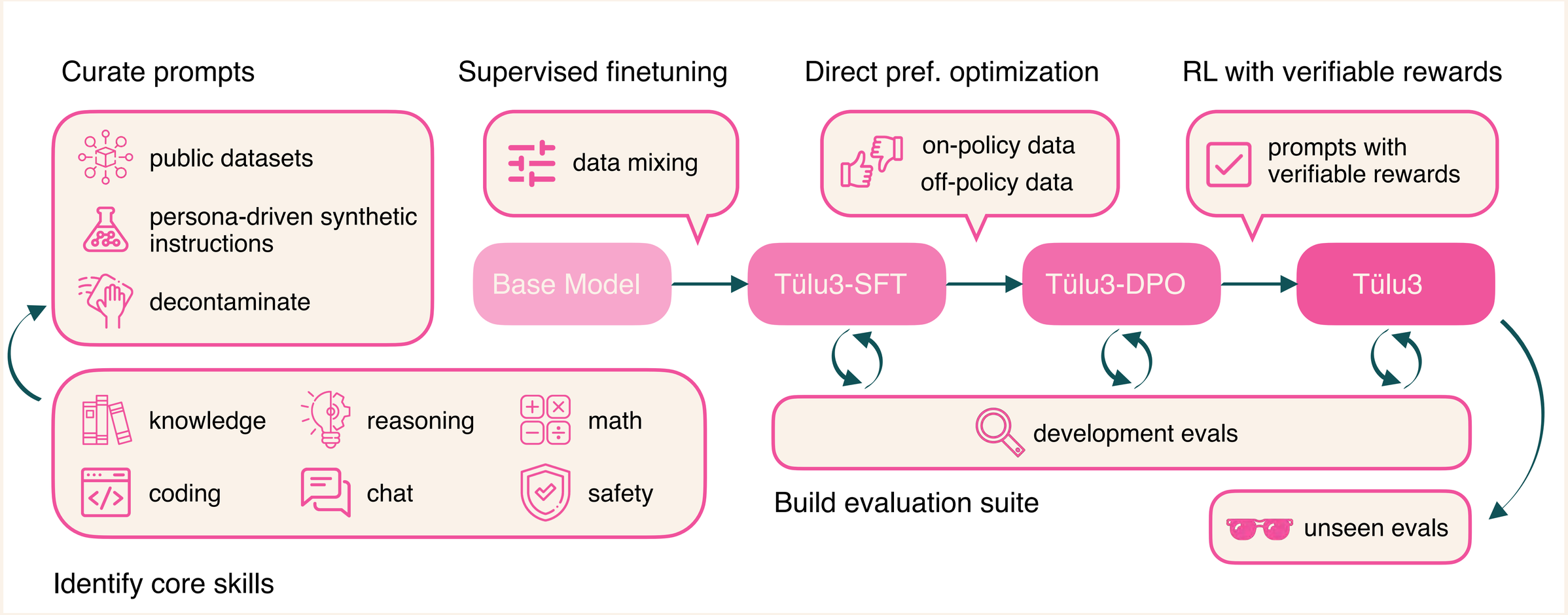
Benchmark$_{(eval)}$	Llama 3.1 405B Instruct	Nous Hermes 3 405B	Deepseek V3	GPT 4o (11-24)	Tülu 3 405B SFT	Tülu 3 405B DPO	Tülu 3 405B RLVR
Avg w/o Safety.	78.1	74.4	79.0	**80.5**	76.3	79.0	80.0
MMLU$_{(5\ shot,\ CoT)}$	**88.0**	84.9	82.1	87.9	84.4	86.6	87.0
PopQA$_{(3\ shot)}$	52.9	54.2	44.9	53.6	55.7	55.4	**55.5**
BigBenchHard$_{(0\ shot,\ CoT)}$	87.1	87.7	**89.5**	83.3	88.0	88.8	88.6
MATH$_{(4\ shot,\ Flex)}$	66.6	58.4	**72.5**	68.8	63.4	59.9	67.3
GSM8K$_{(8\ shot,\ CoT)}$	95.4	92.7	94.1	91.7	93.6	94.2	**95.5**
HumanEval$_{(pass@10)}$	95.9	92.3	94.6	**97.0**	95.7	97.2	95.9
HumanEval+$_{(pass@10)}$	90.3	86.9	91.6	92.7	93.3	**93.9**	92.9
IFEval$_{(loose\ prompt)}$	**88.4**	81.9	88.0	84.8	82.4	85.0	86.0
AlpacaEval 2$_{(LC\ \%\ win)}$	38.5	30.2	53.5	**65.0**	30.4	49.8	51.4
Safety$_{(6\ task\ avg.)}$	86.8	65.8	72.2	**90.9**	87.7	85.5	86.7

Table 4 Summary of TÜLU 3 results relative to peer 405B models. The best-performing model on each benchmark (i.e., in each row) is **bolded**. TÜLU 3-405B outperforms prior state-of-the-art models finetuned from Llama 3.1 405B Base and rivals some leading, closed models. Progress across various checkpoints highlight the contribution of each stage of the training in improving core skills. Note that TruthfulQA and MMLU multiple choice numbers are not compatible with our infrastructure for running evaluations (via log-probs).

Tulu rivals DeepSeek-V3 and GPT-4o

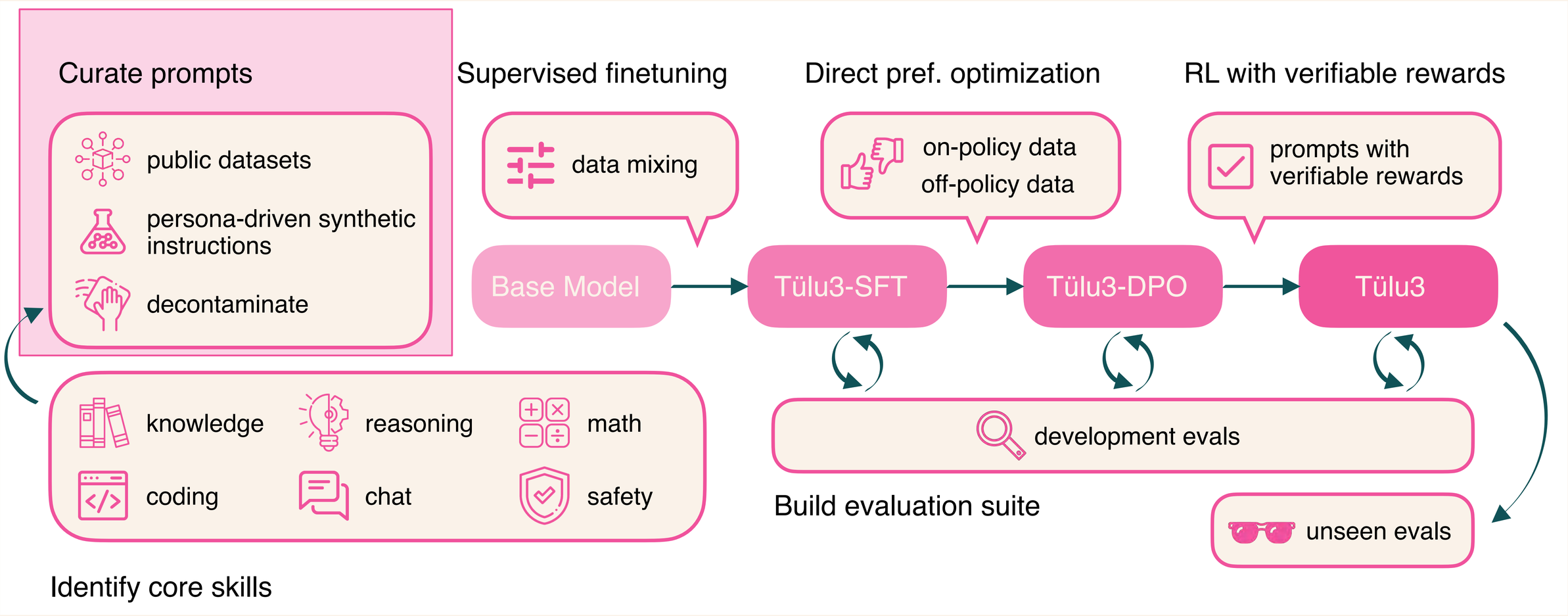


Tulu 3: Our current best recipe



Ai2

Tulu 3: Our current best recipe



Ingredients to start with—Curate targeted set of prompts

Knowledge recall	FLAN v2; SciRIFF; TableGPT
Math and reasoning	OpenMathInstruct 2; NuminaMath
Coding	Evol CodeAlpaca
Safety and non-compliance	CoCoNot; WildJailbreak; WildGuardMix
Multilinguality	Aya
General	OpenAssistant; NoRobots; WildChat; UltraFeedback

1. Find relevant public datasets.

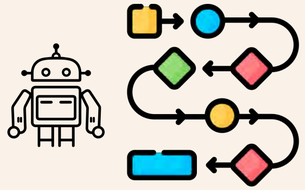
Ingredients to start with—Curate targeted set of prompts

Data mixing & selection from existing resources

1. Find relevant public datasets.

Ingredients to start with—Curate targeted set of prompts

Data mixing & selection from existing resources



Persona-driven Data Synthesis

1. Find relevant public datasets.
2. Synthesize data to fill gaps.

- ☑ Enable targeting specific skills (e.g., math, code, precise instruction following)
- ☑ Ensure high diversity
- ☑ Enable Scaling

Scaling Synthetic Data Creation with 1,000,000,000 Personas

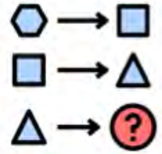
32

Tao Ge* Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, Dong Yu

Curate targeted set of prompts—Persona-drive data synthesis

Create {data} with {persona}

a math problem



a logical reasoning problem

a chemical kinetics researcher

Dr. Smith, a chemist, is studying a reaction where compound X decomposes into products Y and Z. The reaction follows first-order kinetics with a rate constant k of 0.5 min^{-1}.

If the initial concentration of compound X is 1.0 M, how long will it take for the concentration of X to decrease to 0.25 M?

You are analyzing the spatial arrangement of molecules in a reaction chamber. There are three types: A, B, and C. Molecule A is always adjacent to B, but never to C. Molecule B can be adjacent to both A and C.

If molecule C is surrounded by other molecules, which ones must be present around it?

Photo from Ge et al. 2024

Curate targeted set of prompts—Persona-drive data synthesis

Create {data} with

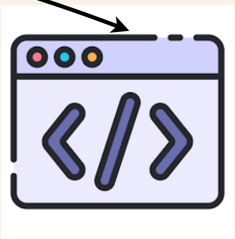
~250K Personas

PersonaHub(Ge et al. 2024)

a math problem

~150k hard math problems

~50k grade school math problems



a coding problem

~35k python coding

Precise Instruction Following

~30k IF data

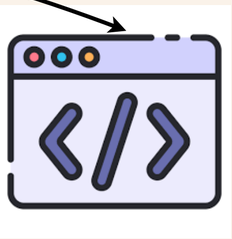
Curate targeted set of prompts—Persona-drive data synthesis

Create {data} with

~250K Personas

PersonaHub(Ge et al. 2024)

a math problem



a coding problem

Precise Instruction Following

~150k hard math problems

~50k grade school math problems

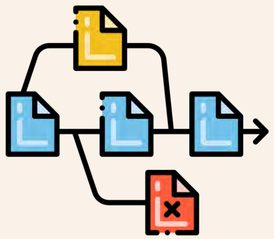
~35k python coding

~30k IF data

GPT-4o / Claude-sonnet

Generate step-by-step solutions for {a math problem}

Ingredients to start with—Curate targeted set of prompts



The Data Provenance Initiative:
A Large Scale Audit of Dataset Licensing & Attribution in AI

Shayne Longpre[1†] Robert Mahari[1,2] Anthony Chen[3] Naana Obeng-Marnu[1,4]
Damien Sileo[5] William Brannon[1,4] Niklas Muennighoff[6] Nathan Khazam[7]
Jad Kabbara[1,4] Kartik Perisetla Xinyi (Alexis) Wu[8] Enrico Shippole Kurt Bollacker[7]
Tongshuang Wu[9] Luis Villa[10] Sandy Pentland[1] Sara Hooker[11]

Evaluating Copyright Takedown Methods for Language Models

Boyi Wei*[1] Weijia Shi*[2] Yangsibo Huang*[1]
Noah A. Smith[2] Chiyuan Zhang Luke Zettlemoyer[2] Kai Li[1] Peter Henderson[1]

1. Find relevant public datasets.
2. Synthesize data to fill gaps.
3. Provenance and copyright

Ingredients to start with—Curate targeted set of prompts

1. Find relevant public datasets.
2. Synthesize data to fill gaps.
3. Provenance and copyright
4. Decontaminate against evaluation suite.

Ingredients to start with—Curate targeted set of prompts

Dataset	🤗 Link	Eval.	% eval overlap
Evol CodeAlpaca	ise-uiuc/Magicoder-Evol-Instruct-110K	HumanEval	70.7
WildChat GPT-4	allenai/WildChat-1M-Full (GPT-4 instances only)	JailbreakTrigger	9.0
		Do-Anything-Now	54.0
WildJailbreak	allenai/wildjailbreak	WildGuardTest	8.2
		HarmBench	6.3
WildGuardmix	allenai/wildguardmix	JailbreakTrigger	19.0
		Do-Anything-Now	39.7
NuminaMath-TIR	AI-MO/NuminaMath-TIR	MATH	18.2
DaringAnteater	nvidia/Daring-Anteater	MATH	30.7
ShareGPT	anon8231489123/ShareGPT_Vicuna_unfiltered	AlpacaEval	19.2
		TruthfulQA	19.1
LMSys Chat 1M	lmsys/lmsys-chat-1m	MMLU	10.3
		HumanEval	17.7
		GSM8K	8.9
		AlpacaEval	46.5
		BBH	10.6
		TruthfulQA	9.2
		JailbreakTrigger	75.0
		HarmbenchEval	9.4
		Do-Anything-Now	90.3
		AGIEval English	18.7
OpenAssistant 2	OpenAssistant/oasst2 (English only)	AlpacaEval	18.3

1. Find relevant public datasets.
2. Synthesize data to fill gaps.
3. Provenance and copyright
4. Decontaminate against evaluation suite.

Many public datasets have high overlaps with popular benchmarks! Especially those containing real conversations with chat bots.

Ingredients to start with—Curate targeted set of prompts

Exact full-prompt matches: too strict

Embedding-based matches: hard to distinguish between contamination and distributional similarity

N-gram matching with heuristics: useful middle-ground

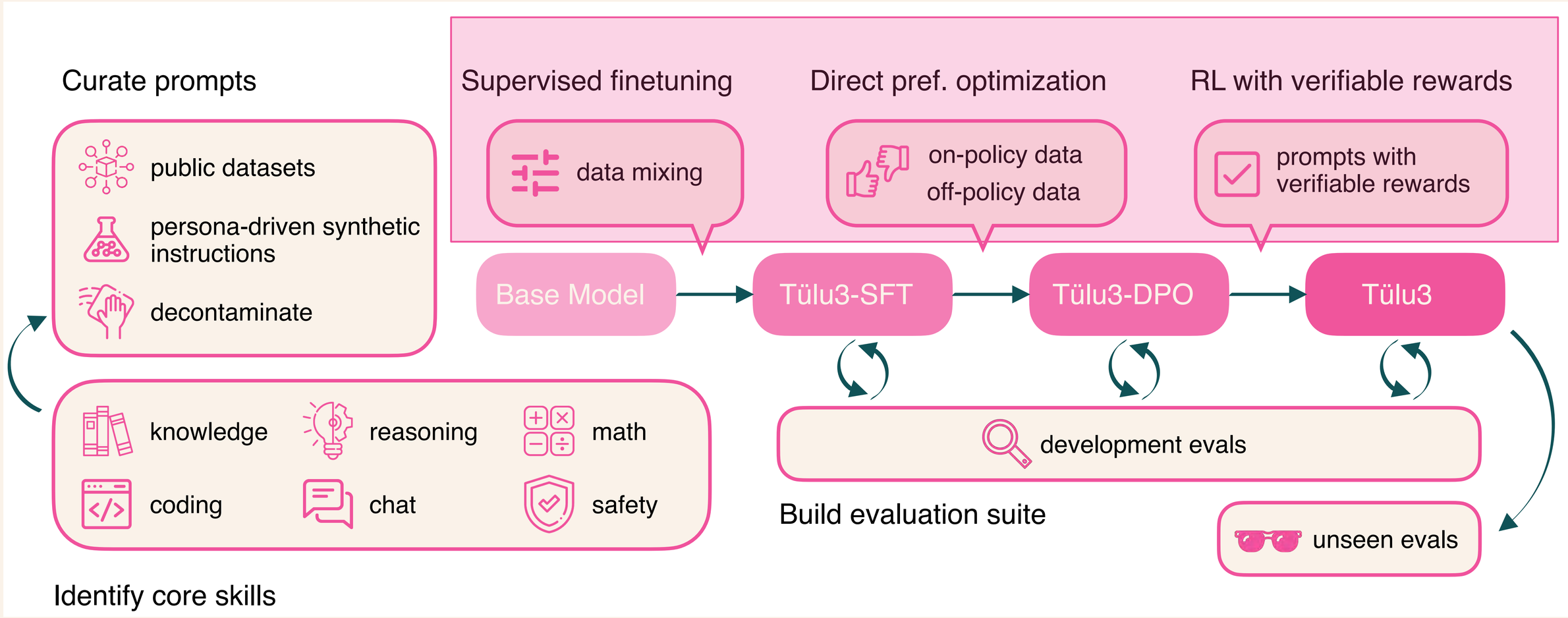
≥50% of test instance tokens have 8-gram overlap with the training instance ⇒ match

1. Find relevant public datasets.
2. Synthesize data to fill gaps.
3. Provenance and copyright
4. Decontaminate against evaluation suite.

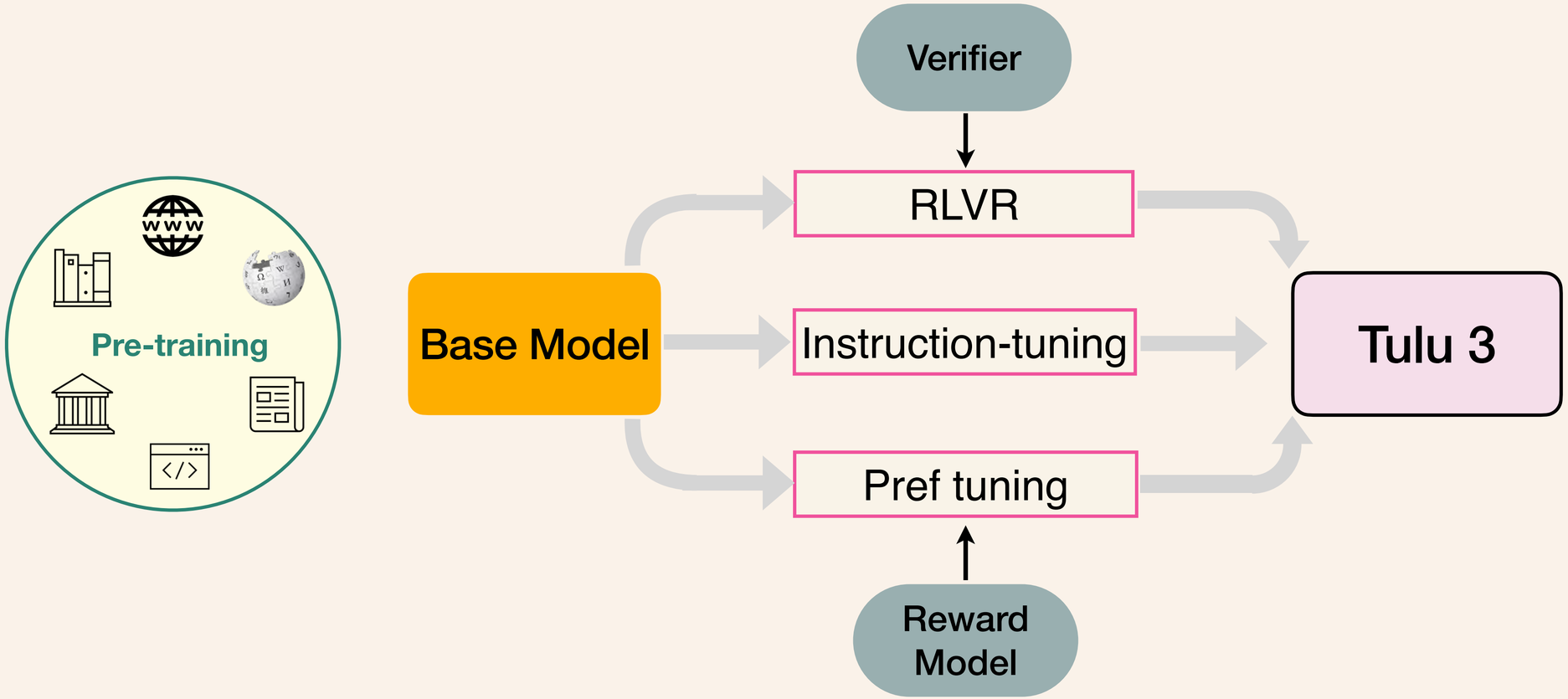
Ingredients to start with—Curate targeted set of prompts

Category	Prompt Dataset	Count	# Prompts used in SFT	# Prompts used in DPO	Reference
General	**Tülu 3 Hardcoded**[↑]	24	240	–	–
	OpenAssistant[1,2,↓]	88,838	7,132	7,132	Köpf et al. (2024)
	No Robots	9,500	9,500	9,500	Rajani et al. (2023)
	WildChat (GPT-4 subset)[↓]	241,307	100,000	100,000	Zhao et al. (2024)
	UltraFeedback[α,2]	41,635	–	41,635	Cui et al. (2023)
Knowledge	FLAN v2[1,2,↓]	89,982	89,982	12,141	Longpre et al. (2023)
Recall	SciRIFF[↓]	35,357	10,000	17,590	Wadden et al. (2024)
	TableGPT[↓]	13,222	5,000	6,049	Zha et al. (2023)
Math	**Tülu 3 Persona MATH**	149,960	149,960	–	–
Reasoning	**Tülu 3 Persona GSM**	49,980	49,980	–	–
	Tülu 3 Persona Algebra	20,000	20,000	–	–
	OpenMathInstruct 2[↓]	21,972,791	50,000	26,356	Toshniwal et al. (2024)
	NuminaMath-TIR[α]	64,312	64,312	8,677	Beeching et al. (2024)
Coding	**Tülu 3 Persona Python**	34,999	34,999	–	–
	Evol CodeAlpaca[α]	107,276	107,276	14,200	Luo et al. (2023)
Safety	**Tülu 3 CoCoNot**	10,983	10,983	10,983	Brahman et al. (2024)
& Non-Compliance	**Tülu 3 WildJailbreak**[α,↓]	50,000	50,000	26,356	Jiang et al. (2024)
	Tülu 3 WildGuardMix[α,↓]	50,000	50,000	26,356	Han et al. (2024)
Multilingual	Aya[↓]	202,285	100,000	32,210	Singh et al. (2024b)
Precise IF	**Tülu 3 Persona IF**	29,980	29,980	19,890	–
	Tülu 3 IF-augmented	65,530	–	65,530	–
Total		23,327,961	939,344	425,145[γ]	

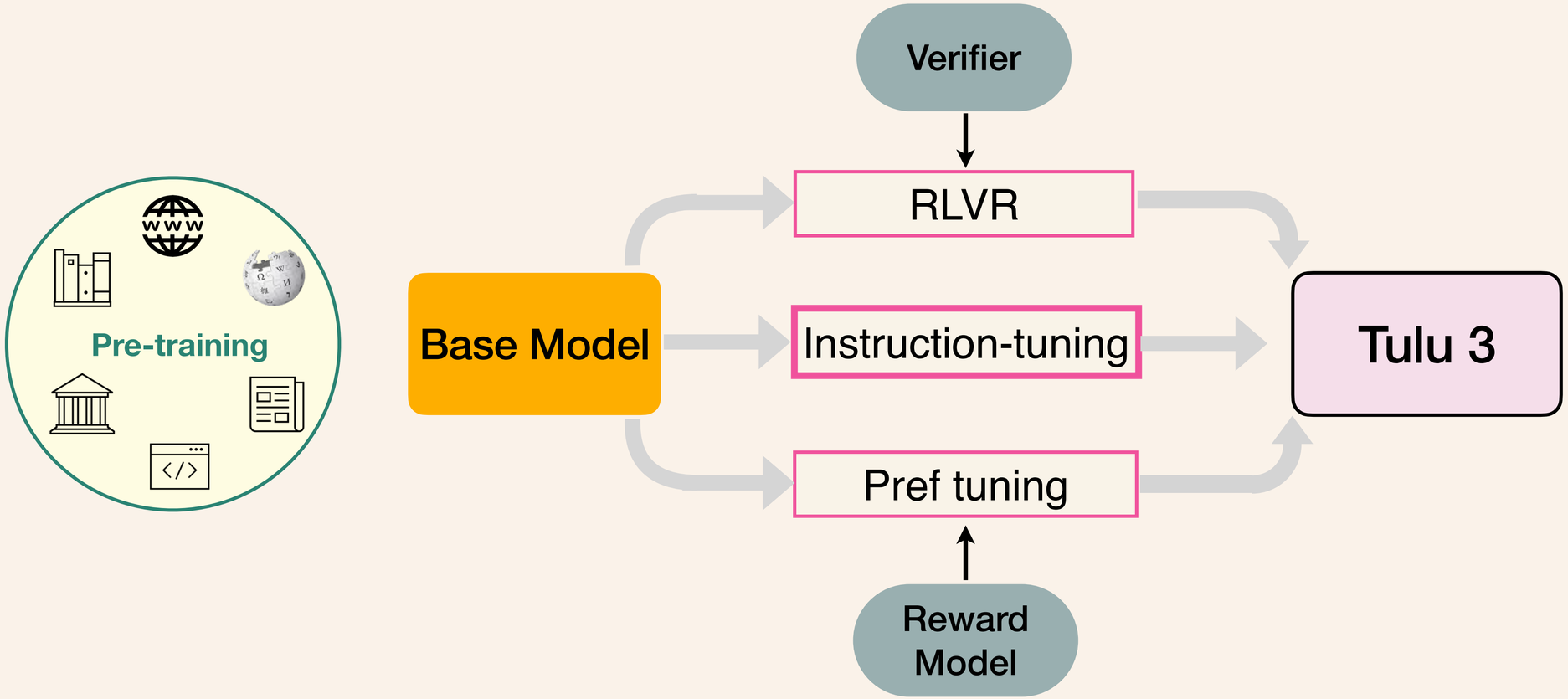
Tulu 3: Training Recipe



Tülu 3 Training Recipe



Tülu 3 Step 1: Supervised Finetuning (aka Instruction Tuning)



Capability-driven Data Mixing for SFT

Two repeated and parallelizable tracks:

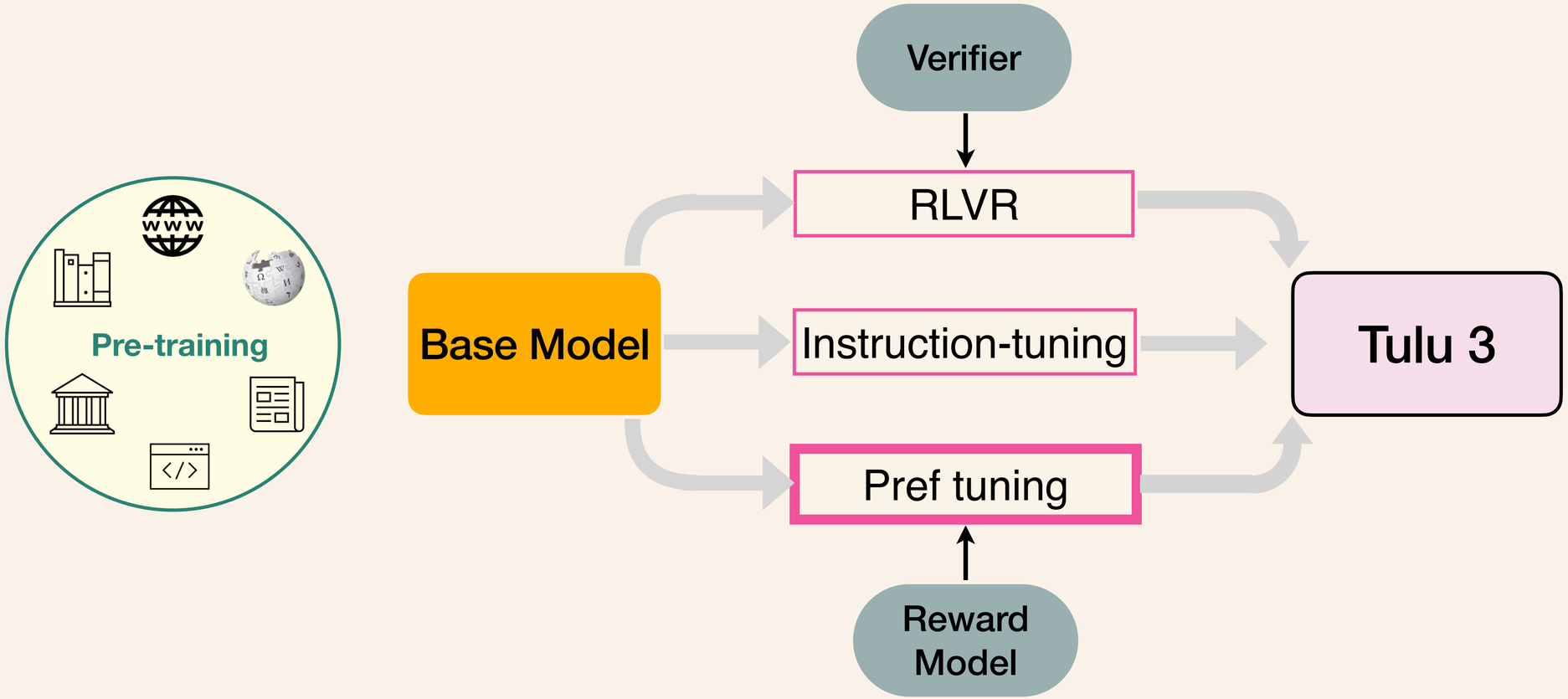
1. **Data curation**: Curate data given targeted capabilities
2. **Data mixing**: Mix data across capabilities
    a. Substantial effort in filtering data while maintaining performance.
    b. Start fully with mixing before curation.

SFT Data Ablations

Model	Avg.	MMLU	TQA	PopQA	BBH	CHE	CHE+	GSM	DROP	MATH	IFEval	AE 2	Safety
Tülu 3 8B SFT	**60.1**	62.1	46.8	29.3	67.9	**86.2**	**81.4**	76.2	61.3	31.5	**72.8**	12.4	93.1
→ w/o WildChat	58.9	61.0	45.2	28.9	65.6	85.3	80.7	75.8	59.3	31.8	70.1	7.5	**95.2**
→ w/o Safety	58.0	62.0	45.5	**29.5**	68.3	84.5	79.6	**76.9**	59.4	**32.6**	71.0	12.4	74.7
→ w/o Persona Data	58.6	**62.4**	**48.9**	29.4	68.3	84.5	79.0	76.8	**62.2**	30.1	53.6	**13.5**	93.9
→ w/o Math Data	58.2	62.2	47.1	**29.5**	**68.9**	86.0	80.5	64.1	60.9	23.5	70.6	12.0	93.5

- Training on real user interactions with strong models is helpful almost across the board.
- Safety training is largely orthogonal to the other skills.
- Persona-based data synthesis is very useful for targeting *new* skills.

Tülu 3 Step 2: Preference Tuning



Why Preference Learning for LLMs?

- For LLMs generating text, what's "good" text? It's not just about grammar or facts, it is about human taste, the coherence of thought, the correctness of reasoning, the removal of undesired percolation of biases in the outputs and much more.
- These are subjective! Trying to write a formula for "good text" is super hard.

Preference Learning to the Rescue!

Preference judgments

Input: `Write a haiku about AI`

Output 1: `Sure, here's a haiku: …`

Output 2: `Sorry, I cannot help you with that.`

Preference Learning to the Rescue!

Preference judgments

Input: `Write a haiku about AI`

Output 1: `Sure, here's a haiku: …`

Output 2: `Sorry, I cannot help you with that.`

Aligning to human preferences gives:

- Stronger training influence for style and chat evaluations (e.g. ChatBotArena).
- Continue building capabilities of skills from SFT, but lower absolute magnitude of improvements.

The Reward Model— Your AI Judge

- We can’t have humans judge every LLM response during training — that’s too slow.
- So, we train a reward model — an AI judge that learns to mimic human preferences.

RL Algorithms use Reward Model: Algorithms like **PPO**, **DPO** & **GRPO** then use this reward model to guide the LLM’s learning.

RLHF Algorithms— PPO

$$\max_{\pi_\theta} \mathbb{E}_{y\sim\pi_\theta(x)}\left[R(x,y)\right] = \left[r_\phi(x,y) - \beta \mathrm{KL}[\pi_\theta(y|x)||\pi_{\mathrm{ref}}(y|x)]\right]$$

Optimize “reward” *inspired* ▲ by human preferences

▲ Constrain the model to stay close to the base LM (preferences are hard to model)

π: LLM policy

π_θ: base LLM

x: prompt

y: completion

PPO vs. Direct Optimization & Friends

$$\max_{\pi_\theta} \mathbb{E}_{y\sim\pi_\theta(x)}\left[R(x,y)\right] = \left[r_\phi(x,y) - \beta \mathrm{KL}[\pi_\theta(y|x)||\pi_{\text{ref}}(y|x)]\right]$$

$$\mathcal{L}_{\text{DPO}}(\pi_\theta;\pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w\mid x)}{\pi_{\text{ref}}(y_w\mid x)} - \beta\log\frac{\pi_\theta(y_l\mid x)}{\pi_{\text{ref}}(y_l\mid x)}\right)\right]$$

$$\mathcal{L}_{\text{SimPO}}(\pi_\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\frac{\beta}{|y_w|}\log\pi_\theta(y_w|x) - \frac{\beta}{|y_l|}\log\pi_\theta(y_l|x) - \gamma\right)\right]$$

Proximal Policy Optimization (PPO; Schulman et al., 2017) first trains a reward model and then uses RL to optimize the policy to maximize those rewards.

Direct Preference Optimization (DPO; Rafailov et al., 2024) directly optimizes the policy on the preference dataset; no explicit reward model.

SimPO (Meng et al., 2024) does not use a reference model.

Length-normalized DPO normalizes log-likelihoods of preferred and rejected responses by their lengths.

RL (PPO, Reinforce, …) vs. DPO

Most important factor: high-quality data

PPO consistently outperforms DPO (~1%), but at the cost of:

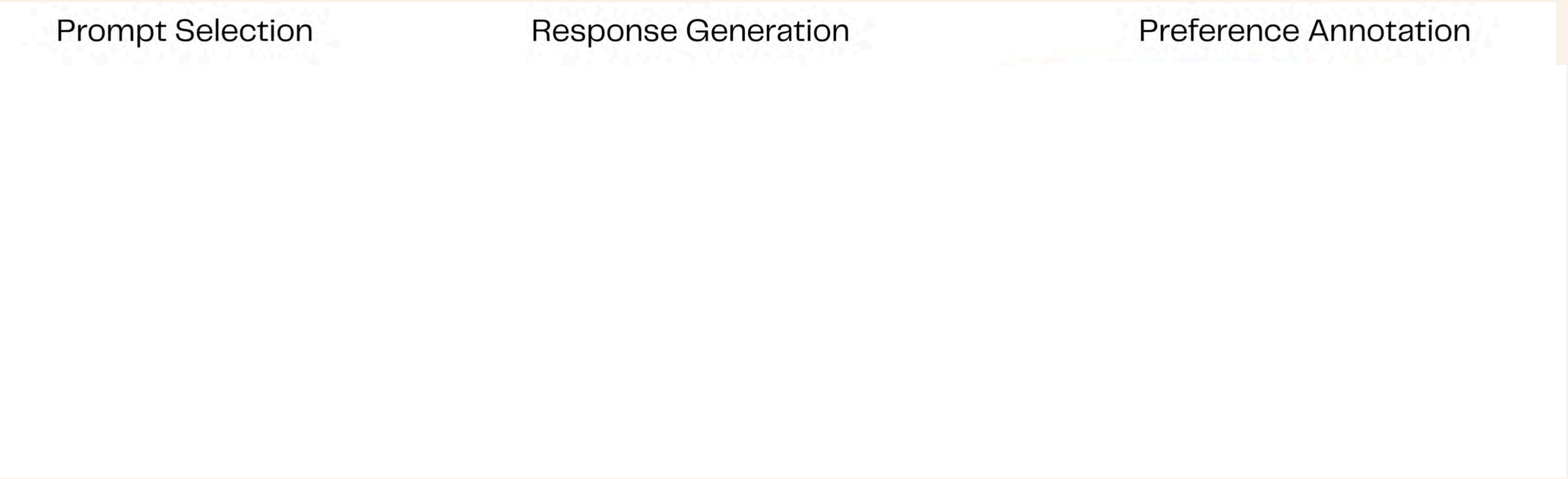
- Implementation complexity
- Memory usage, and
- Throughput (slower training)

Unpacking DPO and PPO: Disentangling Best Practices for Learning from Preference Feedback

Hamish Ivison[♣♠] Yizhong Wang[♣♠] Jiacheng Liu[♣♠]
Zeqiu Wu[♠] Valentina Pyatkin[♣♠] Nathan Lambert[♣]
Noah A. Smith[♣♠] Yejin Choi[♣♠] Hannaneh Hajishirzi[♣♠]

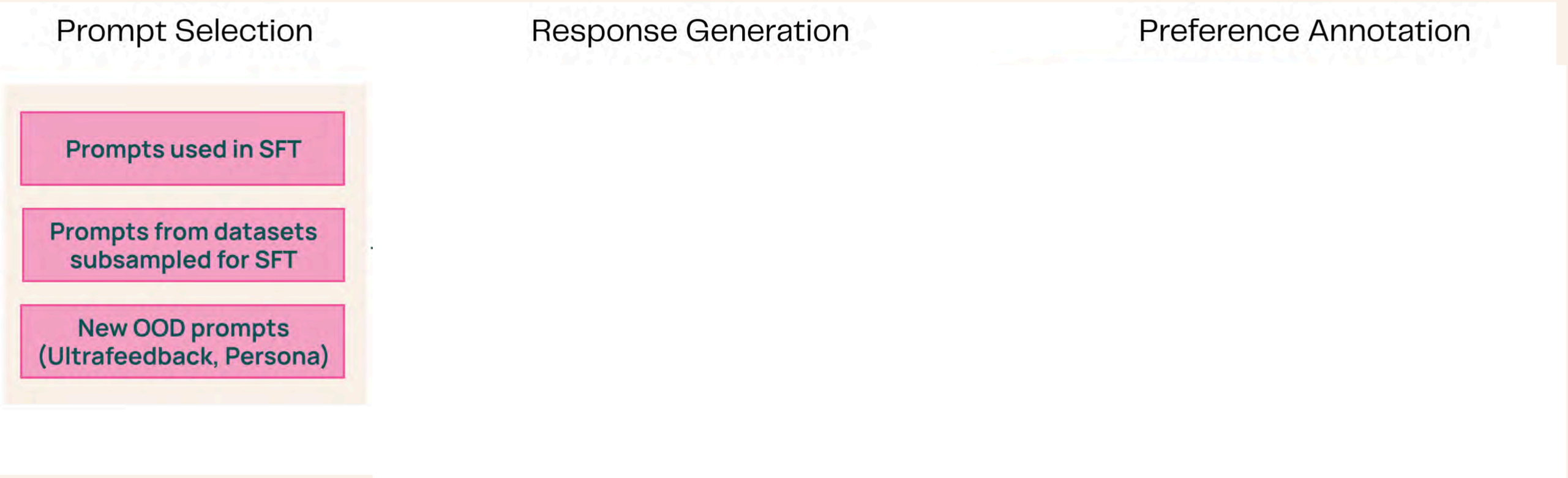
[♣]Allen Institute for AI [♠]University of Washington
hamishiv@cs.washington.edu

Preference Data



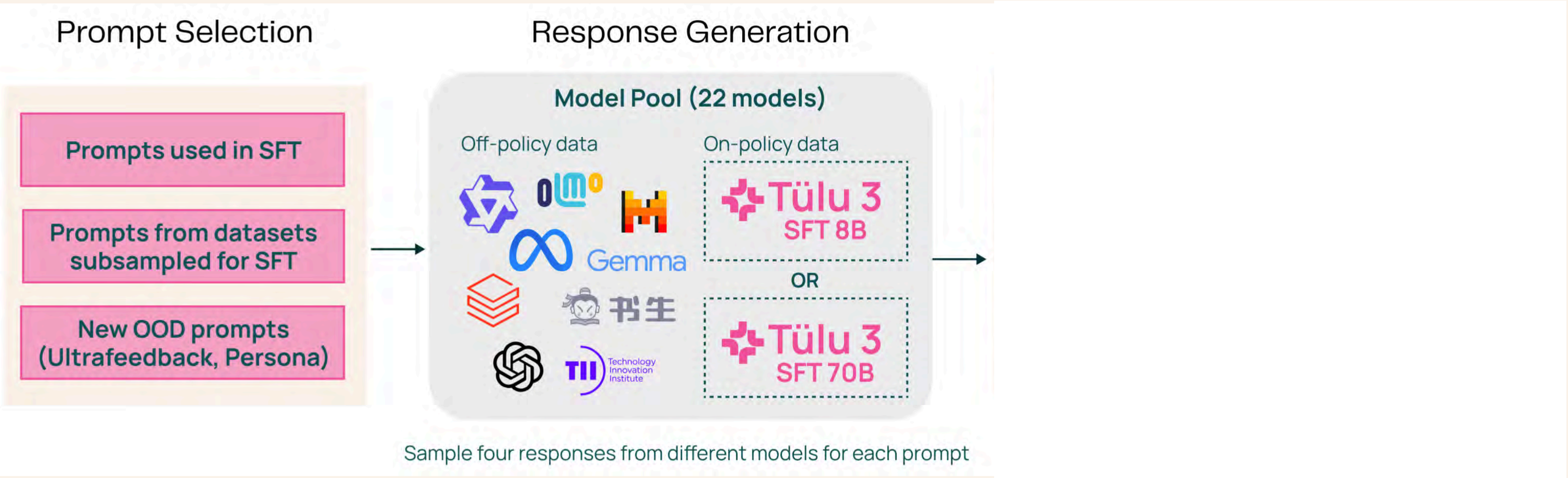
- We adapted and scaled up the UltraFeedback [Cui et al., 2023] for preference data generation.

Preference Data



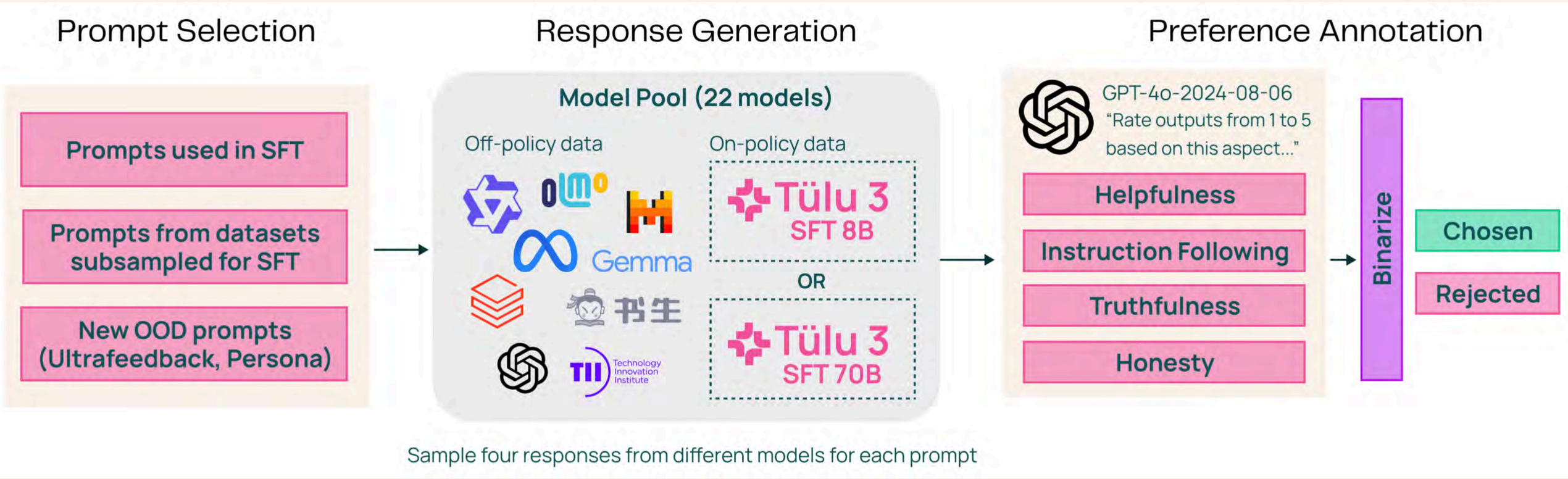
- We adapted and scaled up the UltraFeedback [Cui et al., 2023] for preference data generation.

Preference Data



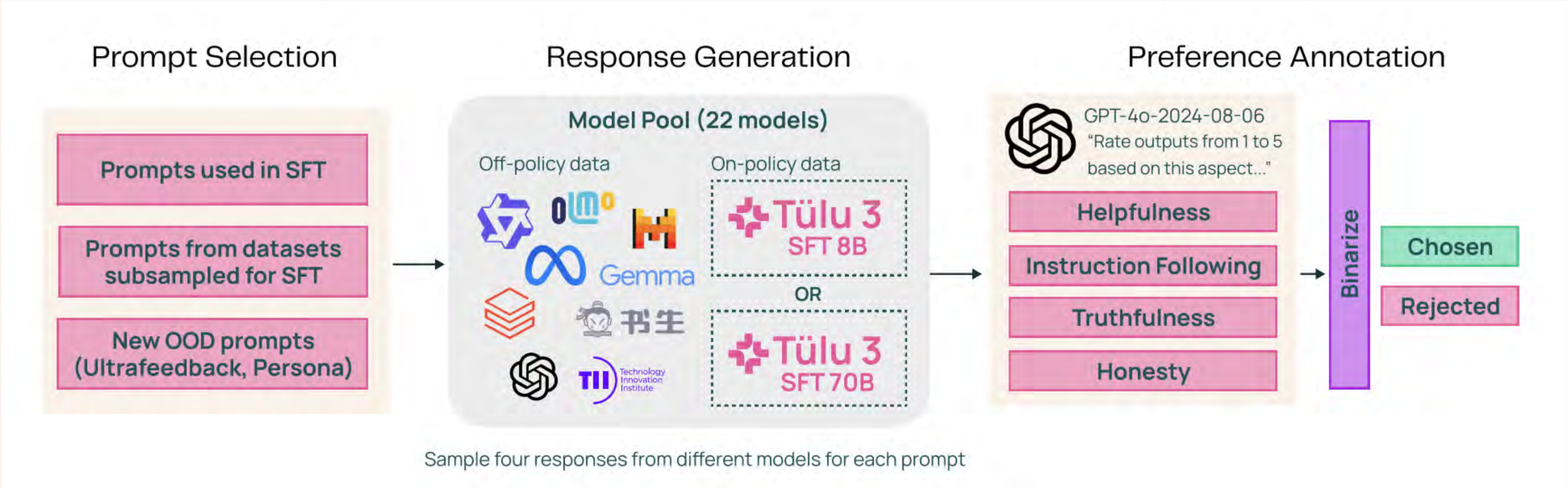
- Model pool consists of both open-source and proprietary models that vary across parameter size and model family

Preference Data

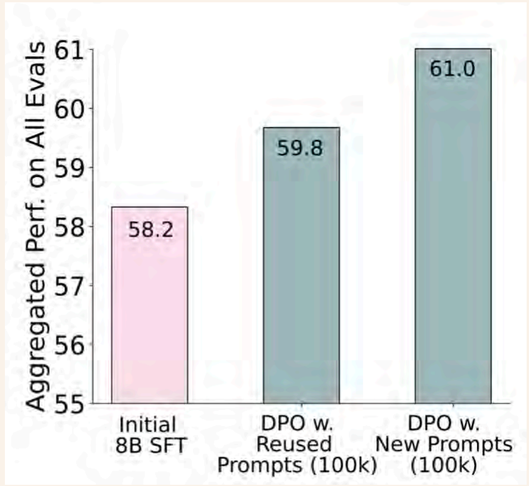


- We experimented with SimPO [Meng et al., 2024], but ended up with the **length-normalized DPO**.

🔎 Preference tuning: findings

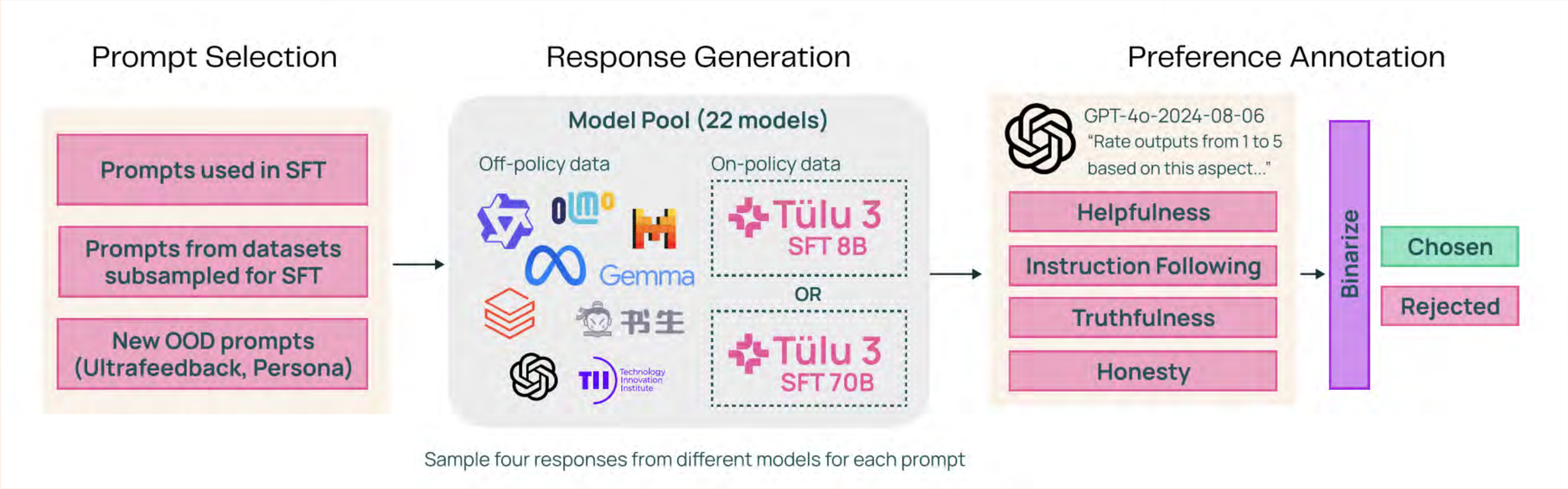


Using SFT vs. new prompts

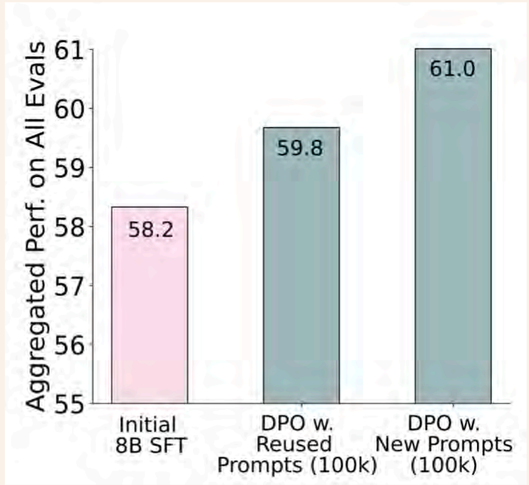


Unused prompts lead to higher performance compared to reusing prompts from SFT Mix

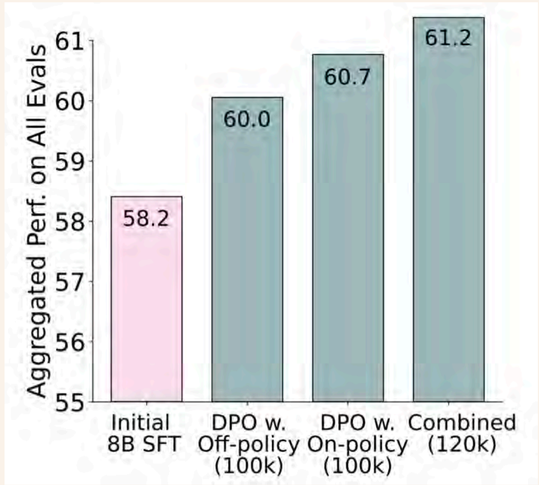
🔎 Preference tuning: findings



Using SFT vs. new prompts

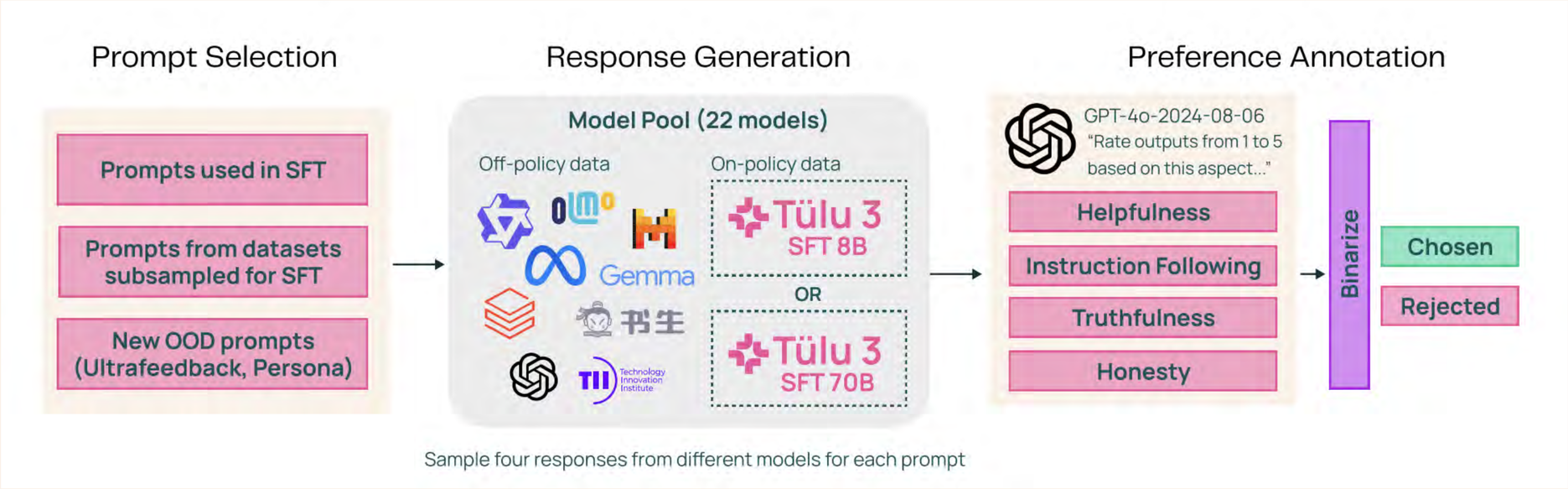


Off- vs on-policy preferences

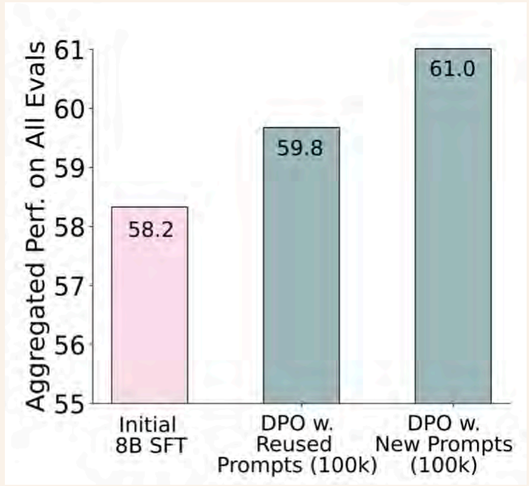


On-policy Data Improves Downstream DPO Performance

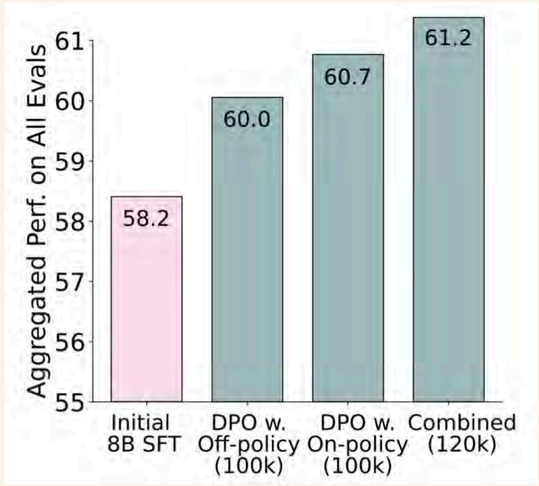
🔎 Preference tuning: findings



Using SFT vs. new prompts



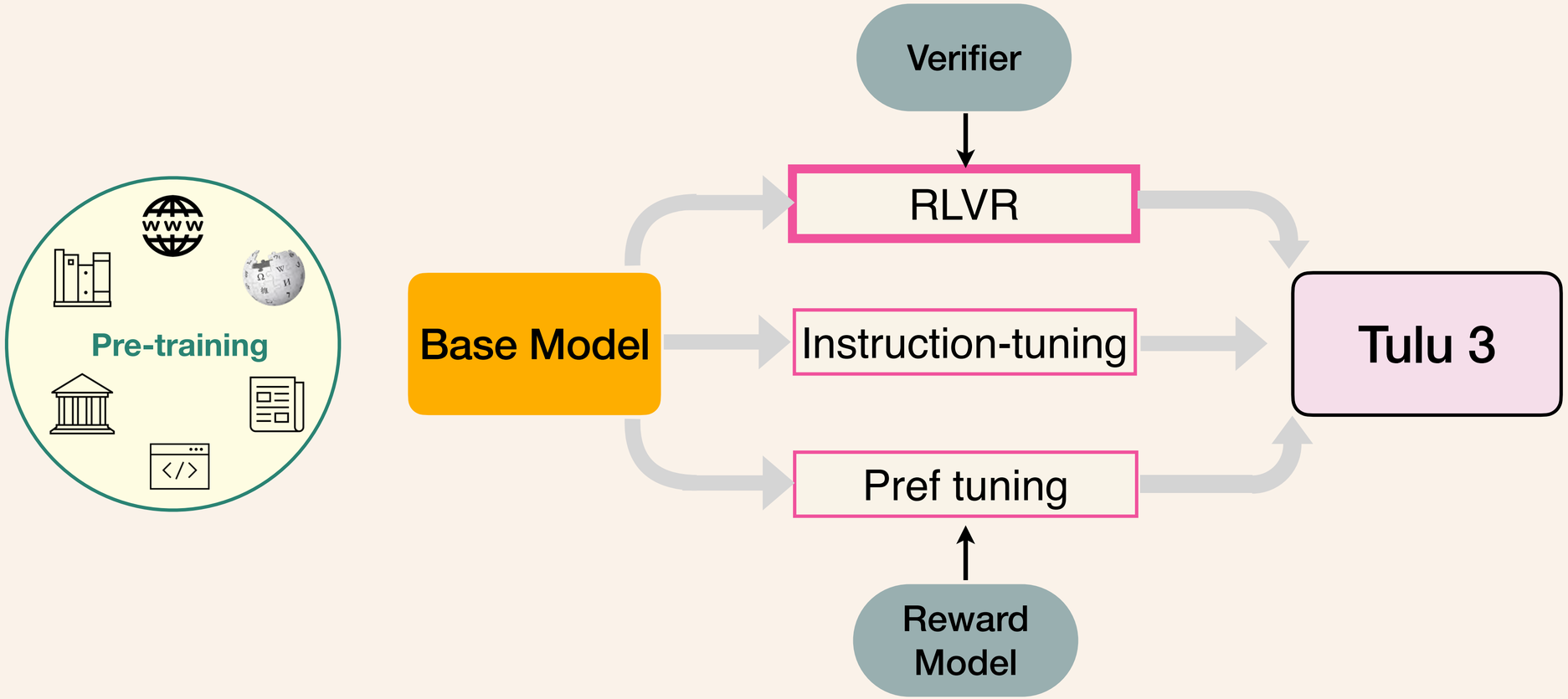
Off- vs on-policy preferences



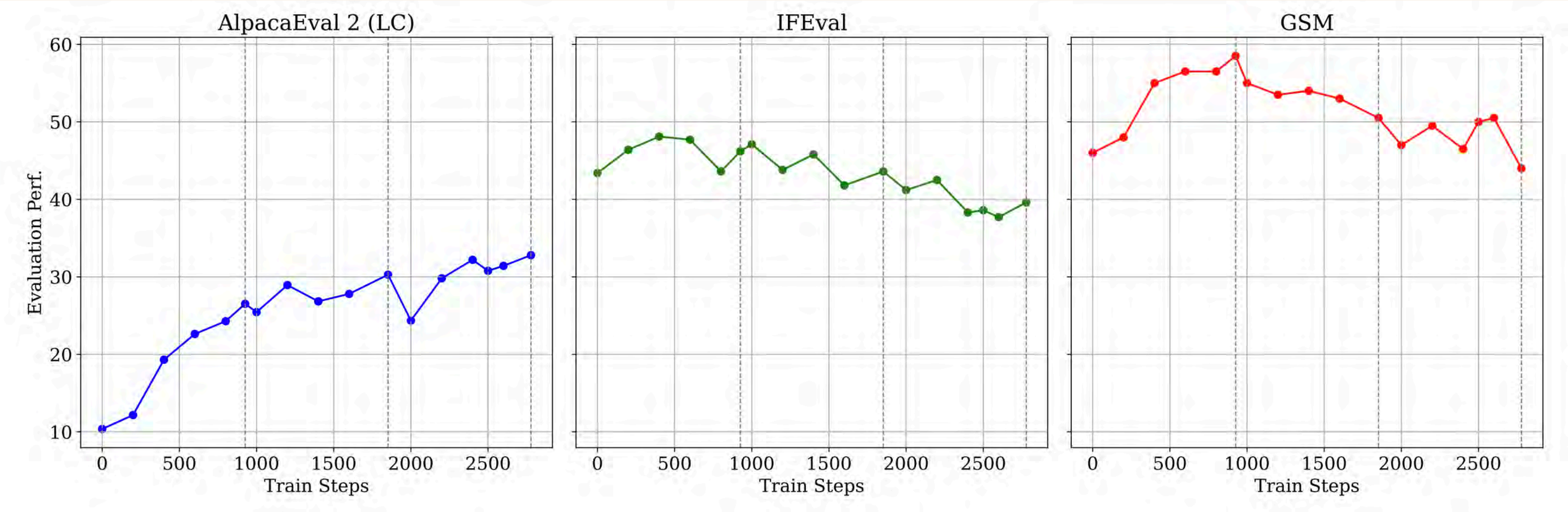
Different LM Judges

LLM Judge	Avg.
GPT-4o	57.3
LLama 3.1 405B	57.2
GPT-4 Turbo	57.0
GPT-4o Mini	56.9
LLama 3.1 70B	56.6

Tülu 3 Step 3: RLVR

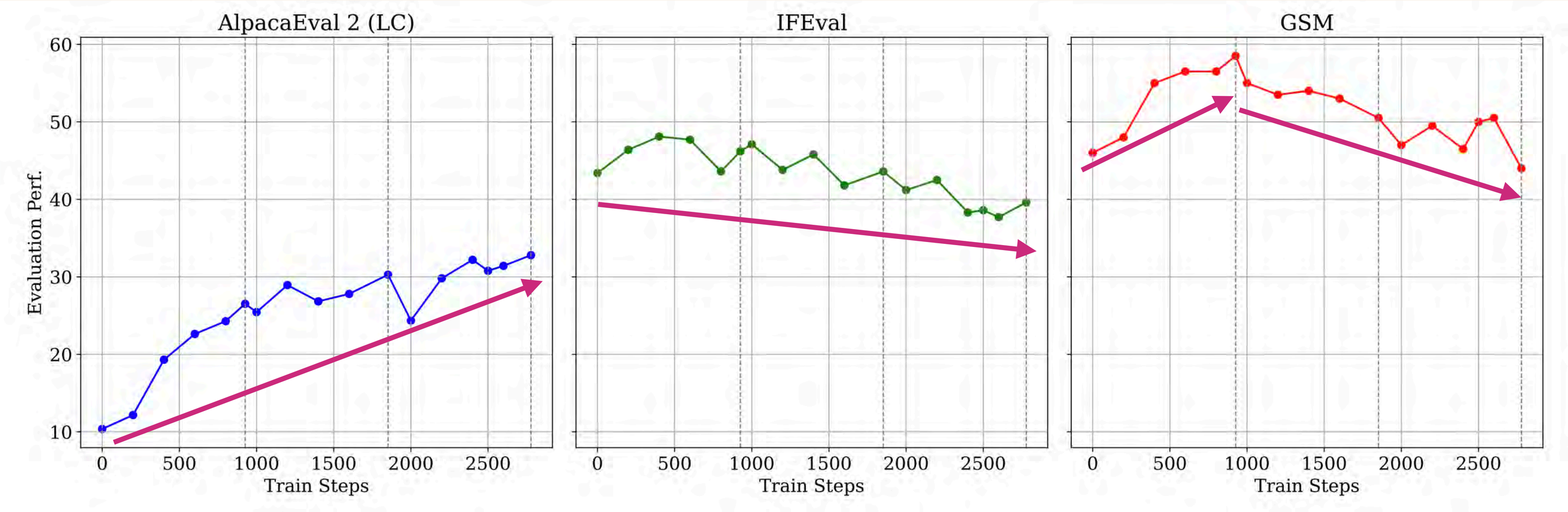


Perils of over-optimization (PPO)



Ai2

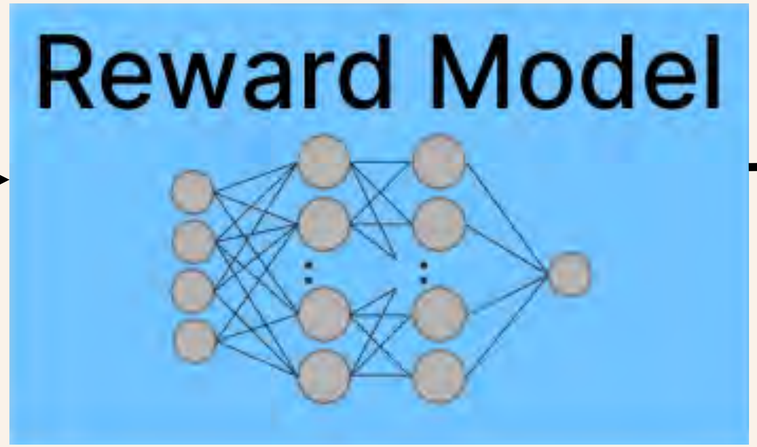
Perils of over-optimization (PPO)



Ai2

Why? Neural RM…

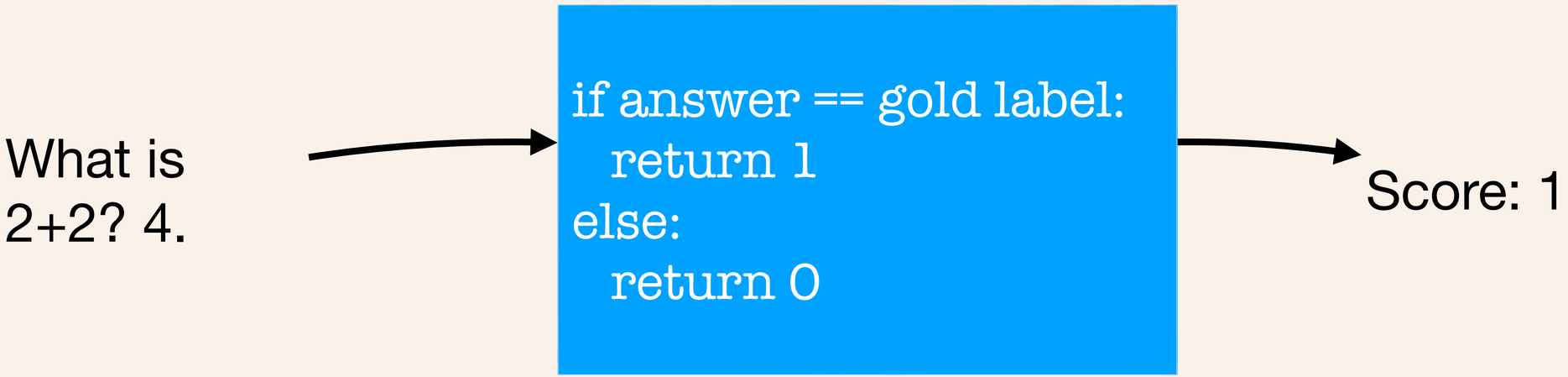
What is a Tulu? A Tulu is a camel that…



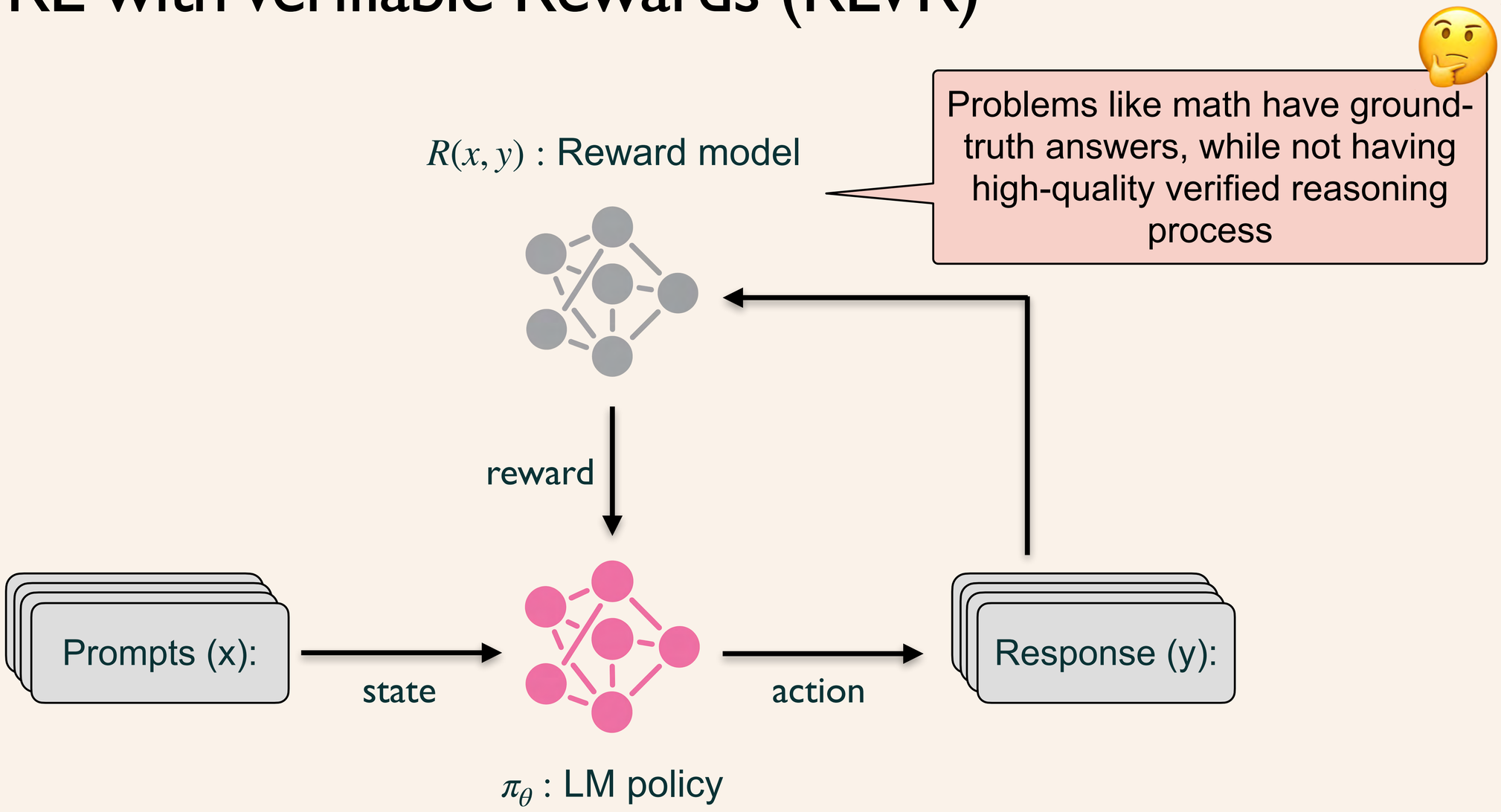
Score: 10.5

- The RM is an approximation of human preferences, and often imperfect.
- The model/policy learns to exploit the patterns and loopholes in the RM and thus don't generalize well.

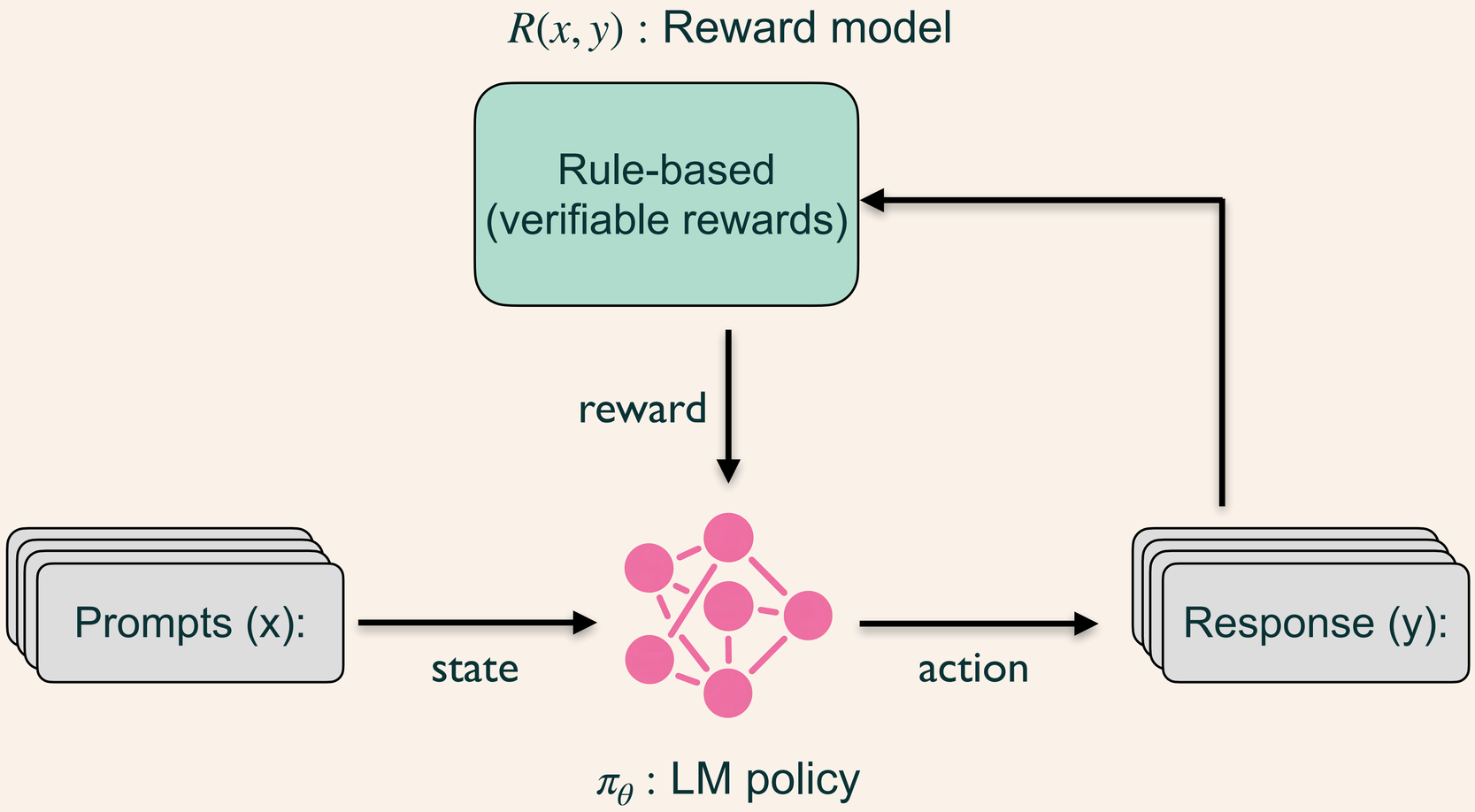
Simplifying the reward model: verifiable rewards



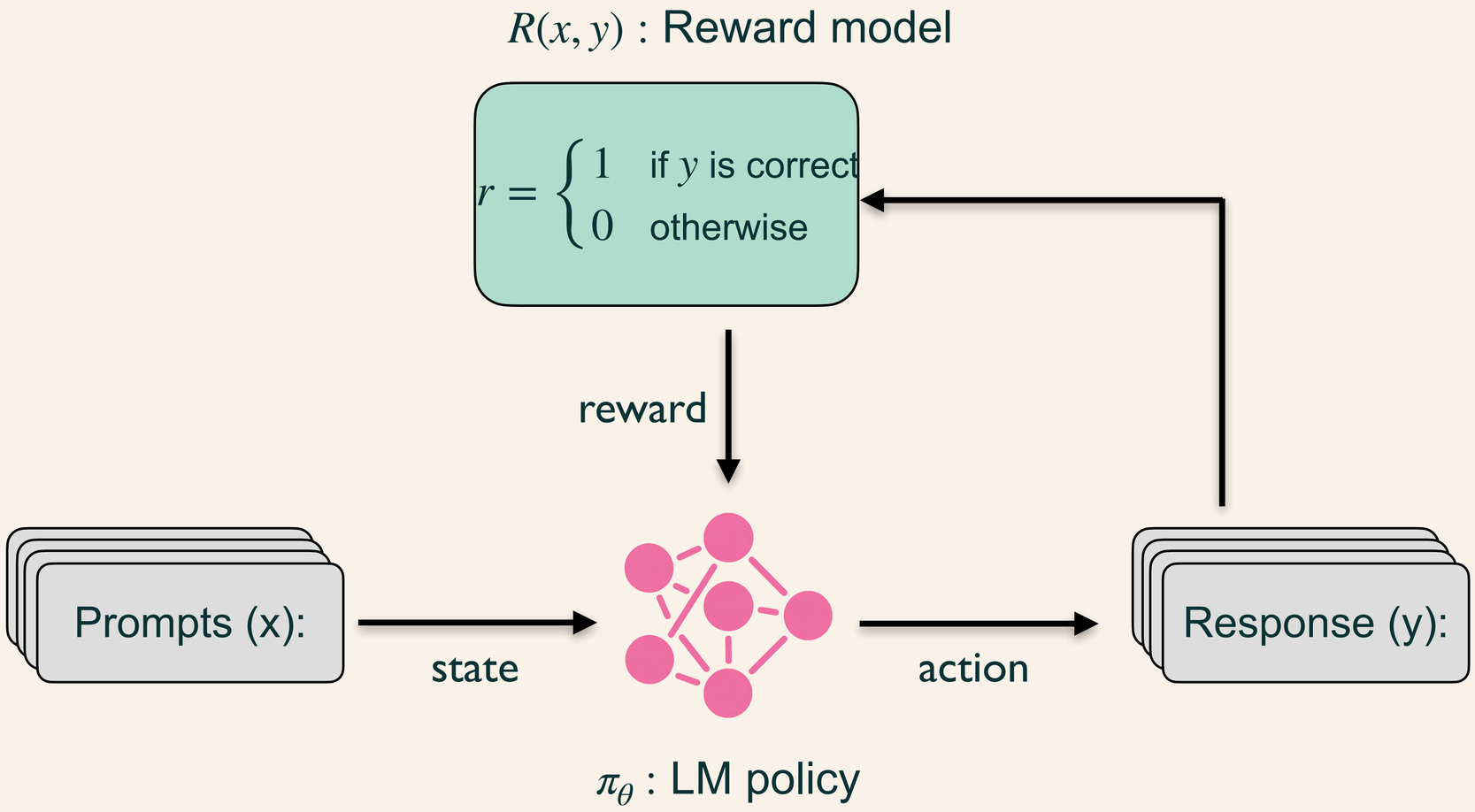
Tülu 3: RL with Verifiable Rewards (RLVR)



Tülu 3: RL with Verifiable Rewards (RLVR)



Tülu 3: RL with Verifiable Rewards (RLVR)



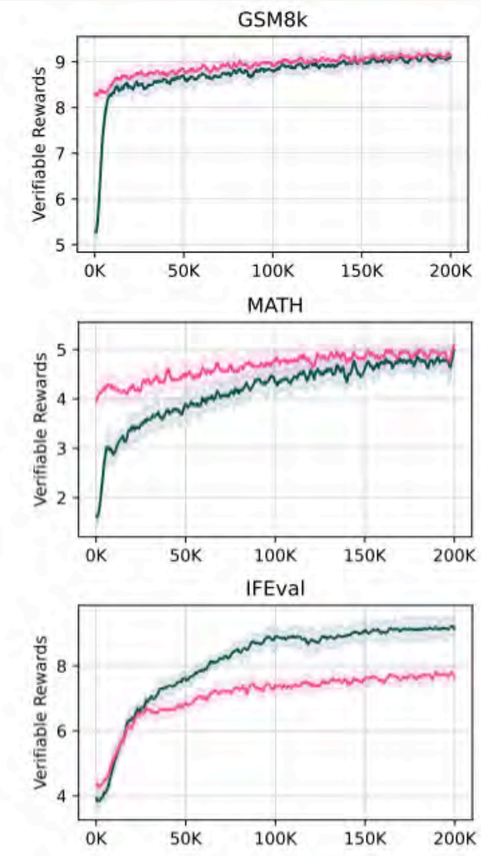
But does it work in practice?

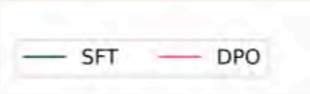
Experimental Setup

1. Start from Tulu 3 DPO and SFT
2. Use targeted datasets + paired verification functions
3. Train with PPO

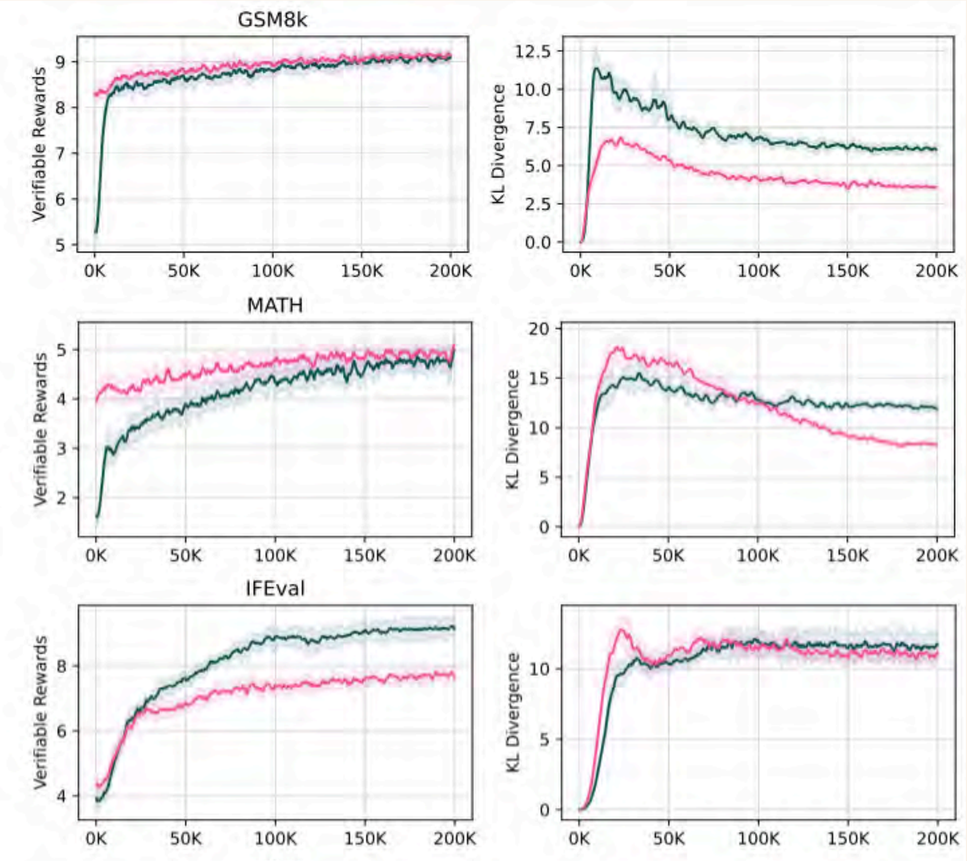
Evaluation	Training Data
GSM8k	GSM8k train set (~7k)
MATH	MATH train set (~7k)
IFEval	IFEval (~15k)

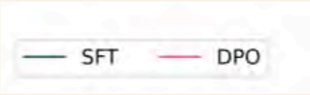
Training Curves



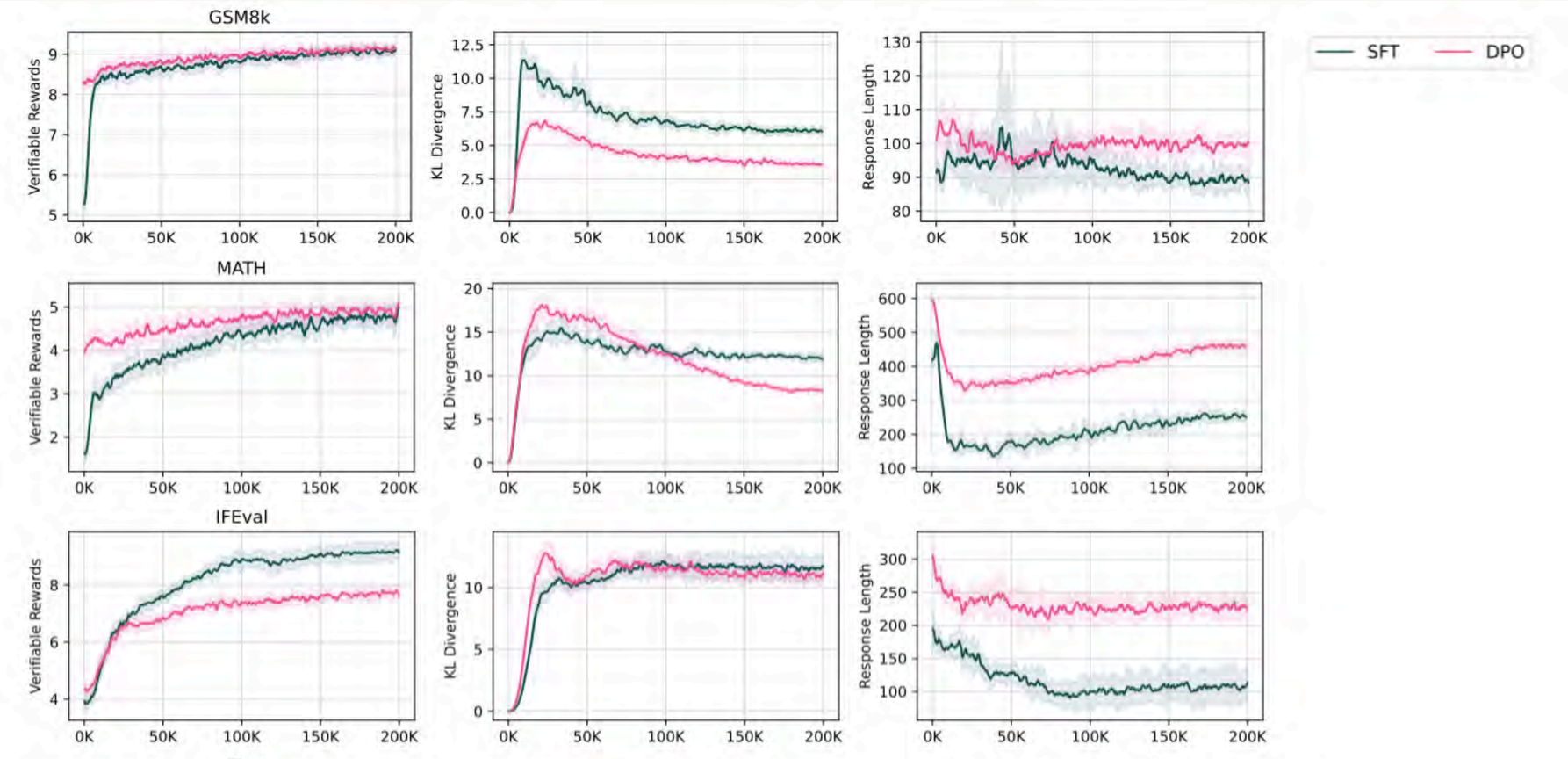


Training Curves



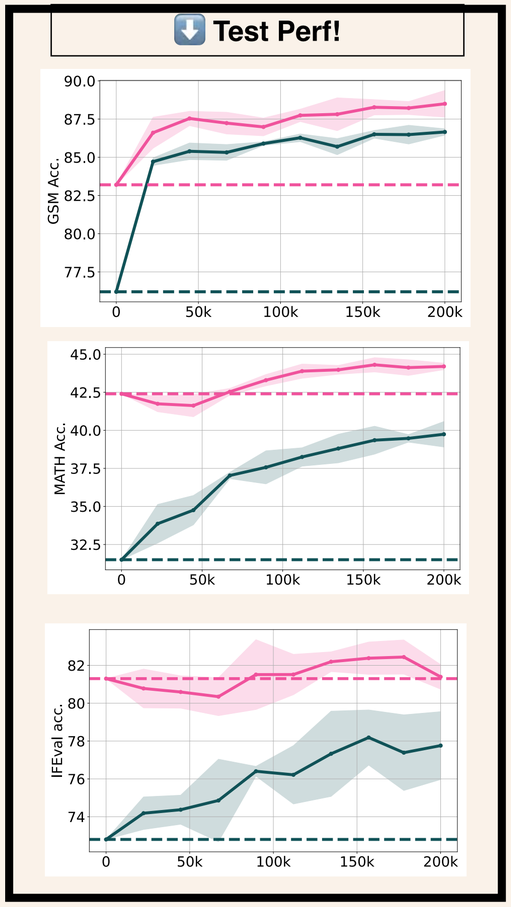


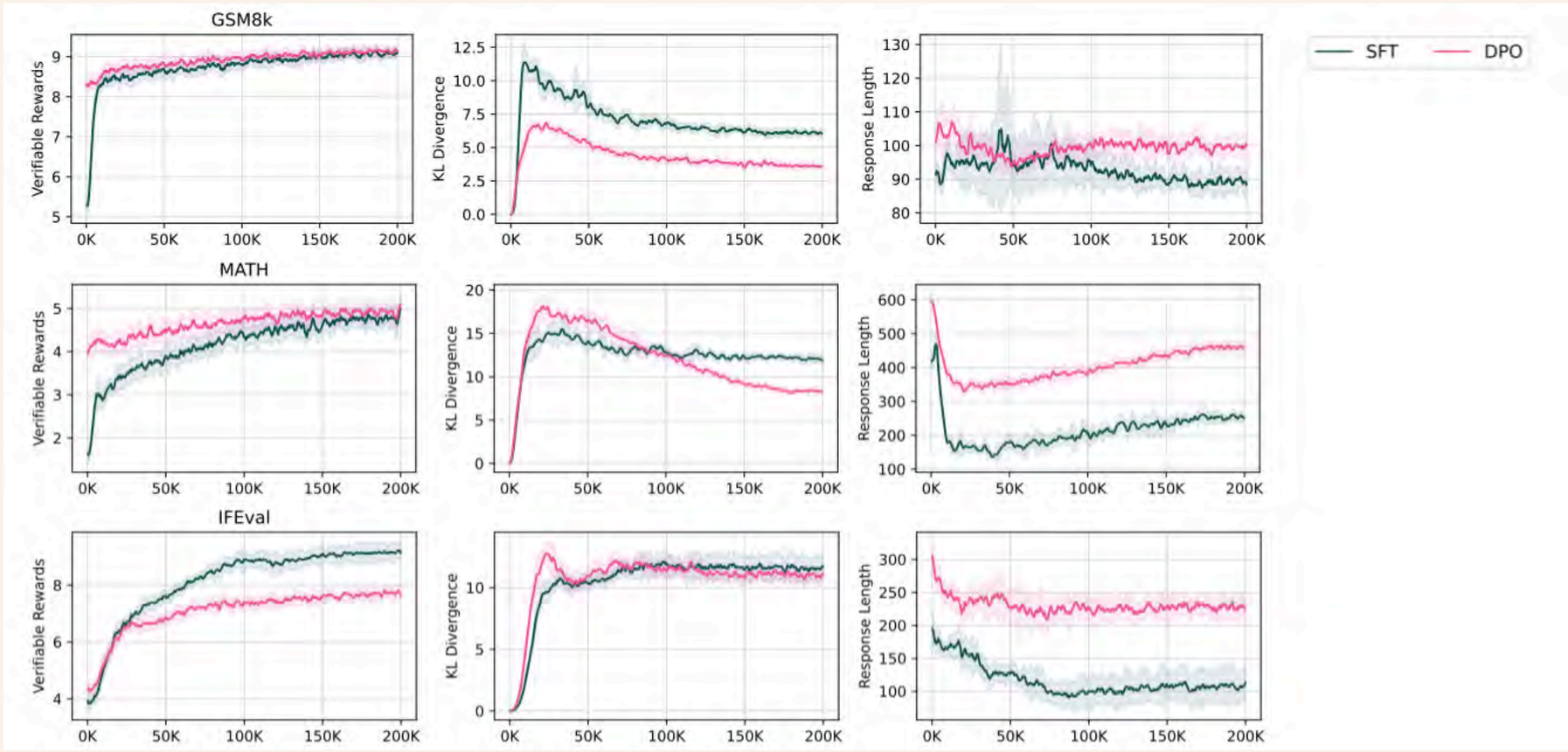
Training Curves



Ai2

Training Curves

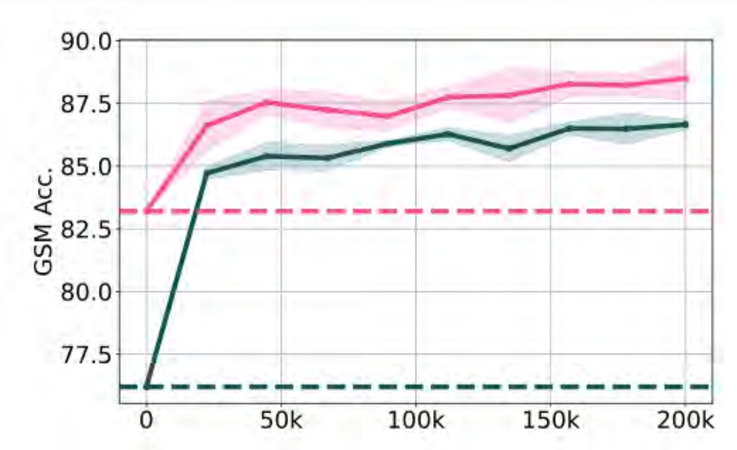




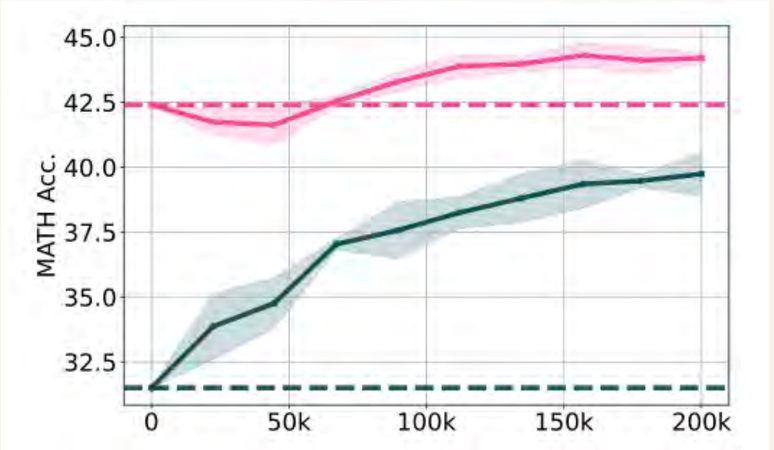
Ai2

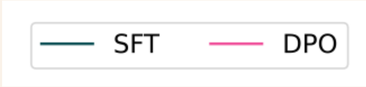
Digging in further

GSM Perf.



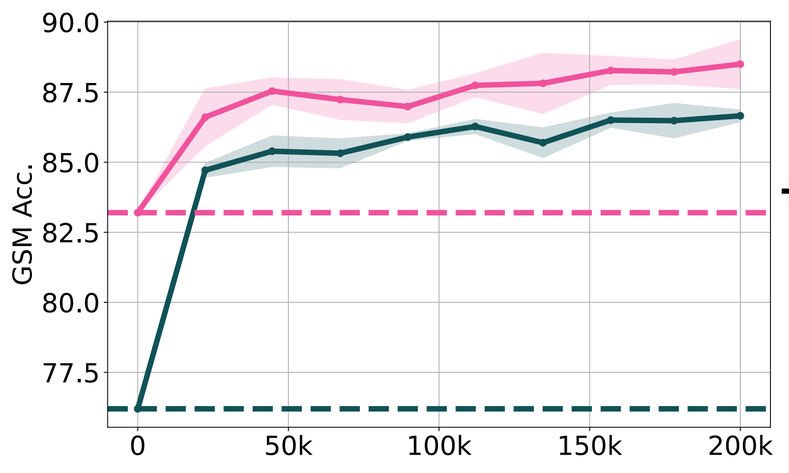
MATH Perf.





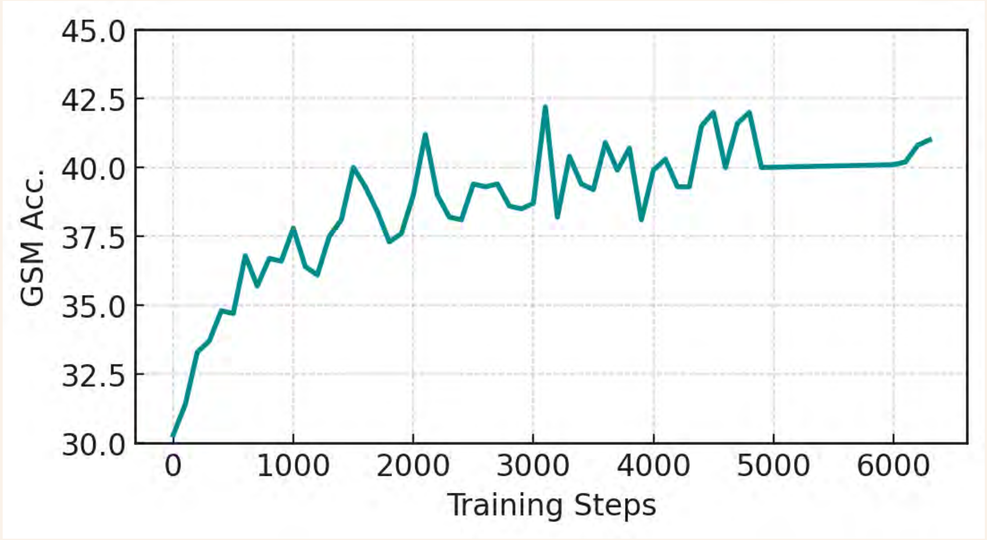
1. No sign of over-optimization for MATH and GSM8K

Digging in further



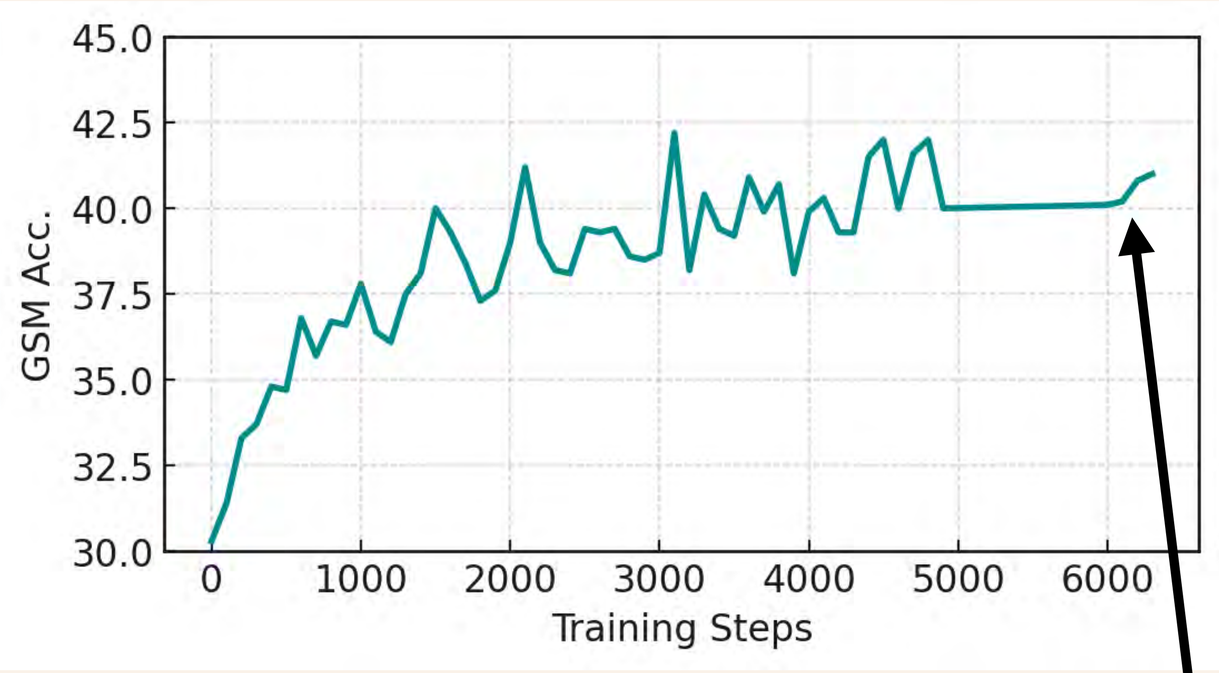
Tulu 3 SFT/DPO 8B

Llama 3.2 1B
+ SFT



1. No sign of over-optimization for MATH and GSM8K
2. Weaker / worse models can still benefit from RLVR.

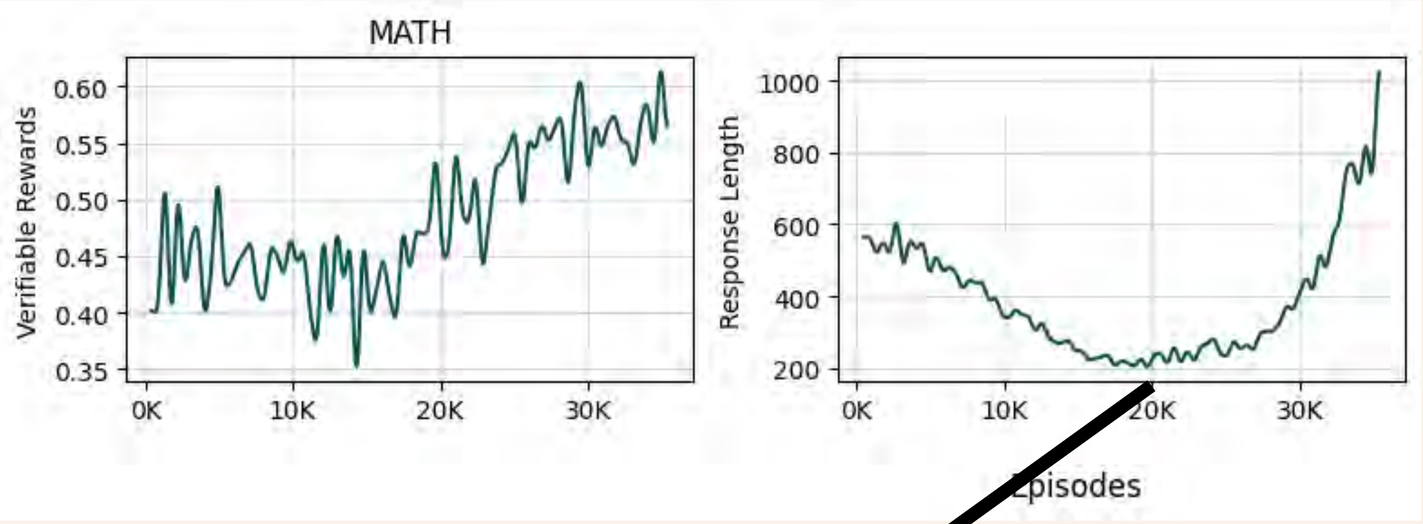
Digging in further



~20 epochs!

1. No sign of over-optimization for MATH and GSM8K
2. Weaker / worse models can still benefit from RLVR.
3. Data efficiency is extremely high - still improving over many steps.

“A-ha” moment?



Model Response: “…This means \\(x\\) must be between 4 and 3, which is impossible. Let's recheck:…This indicates a mistake in the initial setup. Let's correct it:….”

1. No sign of over-optimization for MATH and GSM8K
2. Lower / worse models can still benefit from RLVR.
3. Data efficiency is extremely high - still improving over many samples.
4. RL can lead to emerging behaviors!

RLVR was also used by DeepSeek R1

2.2.2. *Reward Modeling*

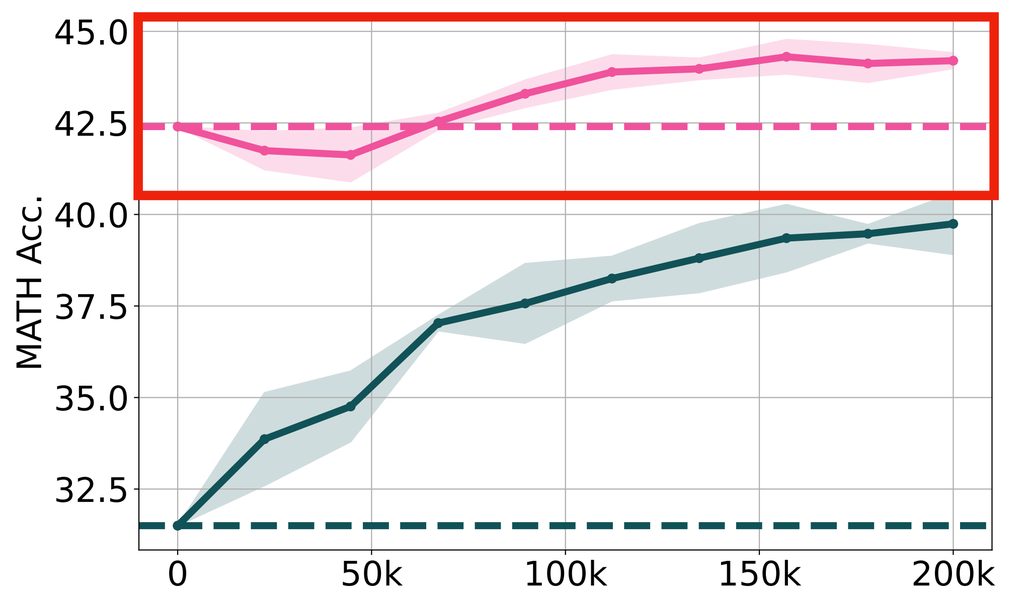
The reward is the source of the training signal, which decides the optimization direction of RL. To train DeepSeek-R1-Zero, we adopt a rule-based reward system that mainly consists of two types of rewards:

- **Accuracy rewards**: The accuracy reward model evaluates whether the response is correct. For example, in the case of math problems with deterministic results, the model is required to provide the final answer in a specified format (e.g., within a box), enabling reliable rule-based verification of correctness. Similarly, for LeetCode problems, a compiler can be used to generate feedback based on predefined test cases.
- **Format rewards**: In addition to the accuracy reward model, we employ a format reward model that enforces the model to put its thinking process between '<think>' and '</think>' tags.

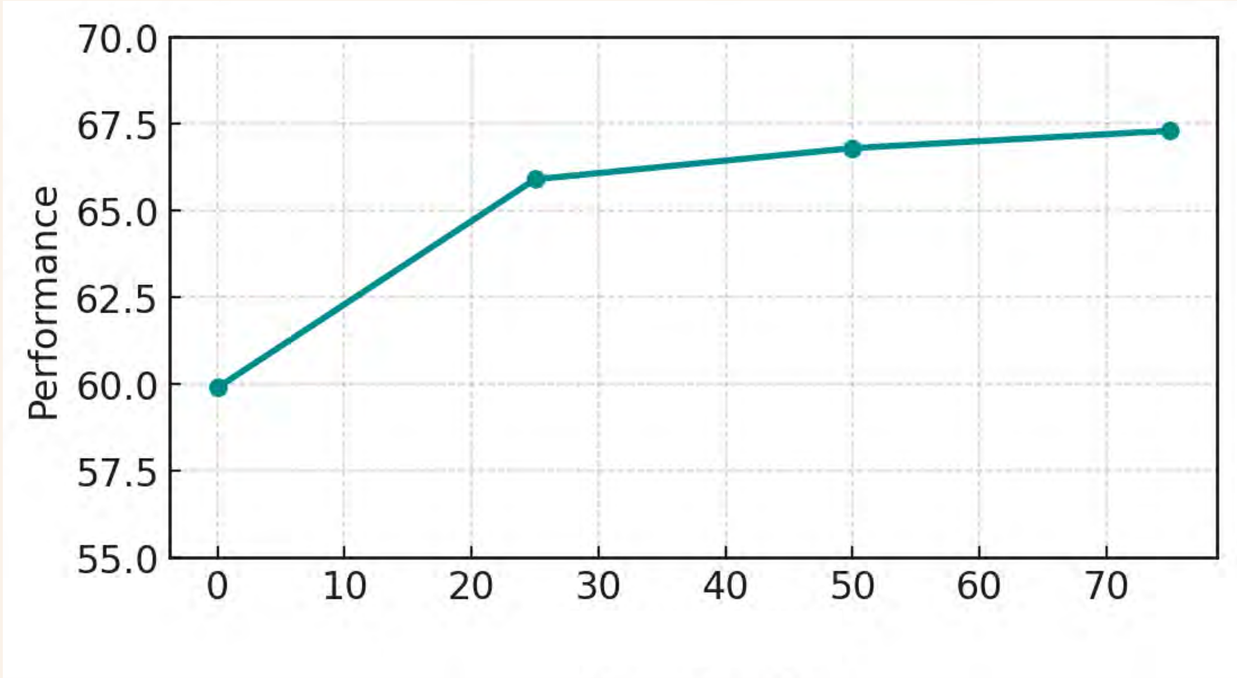
We do not apply the outcome or process neural reward model in developing DeepSeek because we find that the neural reward model may suffer from reward hacking in the large-scale reinforcement learning process, and retraining the reward model needs additional training resources and it complicates the whole training pipeline.

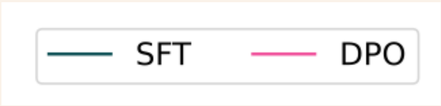
RLVR works better at scale

8B training



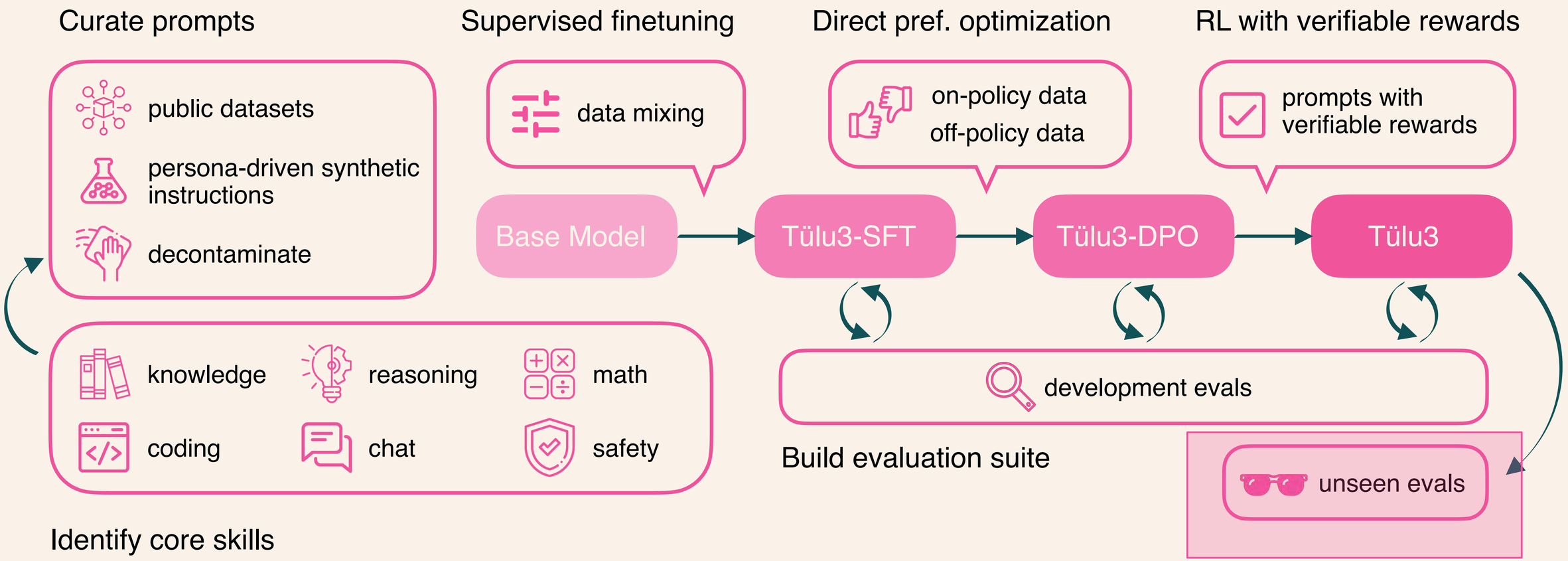
405B training





Ai2

Tülu 3 recipe



Ai2

Ingredients to start with—Reliable unseen evaluation

Core Skill	Development
Knowledge	$\text{MMLU}_{(\text{em})}$
	$\text{PopQA}_{(\text{EM})}$
	$\text{TruthfulQA}_{(\text{MC2 em})}$
Reasoning	$\text{BigBenchHard}_{(\text{em})}$
	$\text{DROP}_{(\text{F1})}$
Math	$\text{MATH}_{(\text{flex em})}$
	$\text{GSM8K}_{(\text{em})}$
Coding	$\text{HumanEval}_{(\text{Pass@10})}$
	$\text{HumanEval+}_{(\text{Pass@10})}$
Instruction Following (IF)	$\text{IFEval}_{(\text{em})}$
	$\text{AlpacaEval 2}_{(\text{winrate})}$
Safety	$\text{TÜLU 3 Safety}_{(\text{avg*})}$

During development: hill climb on reliable evaluations and compare against prior work.

But how to ensure we are not **overfitting** to those evaluations?

Ingredients to start with—Reliable unseen evaluation

Core Skill	Development	Unseen
Knowledge	$\text{MMLU}_{(\text{em})}$ $\text{PopQA}_{(\text{EM})}$ $\text{TruthfulQA}_{(\text{MC2 em})}$	$\text{MMLU-Pro}_{(\text{em})}$ $\text{GPQA}_{(\text{em})}$
Reasoning	$\text{BigBenchHard}_{(\text{em})}$ $\text{DROP}_{(\text{F1})}$	$\text{AGIEval English}_{(\text{em})}$
Math	$\text{MATH}_{(\text{flex em})}$ $\text{GSM8K}_{(\text{em})}$	$\text{Deepmind Mathematics}_{(\text{em})}$
Coding	$\text{HumanEval}_{(\text{Pass@10})}$ $\text{HumanEval+}_{(\text{Pass@10})}$	$\text{BigcodeBench}_{(\text{Pass@10})}$
Instruction Following (IF)	$\text{IFEval}_{(\text{em})}$ $\text{AlpacaEval 2}_{(\text{winrate})}$	$\text{IFEval-OOD}_{(\text{Pass@1})}$ $\text{HREF}_{(\text{winrate})}$
Safety	$\text{TÜLU 3 Safety}_{(\text{avg*})}$	

During development: hill climb on reliable evaluations and compare against prior work.

But how to ensure we are not **overfitting** to those evaluations?

Our solution: Separate set of unseen evaluations run only at the end of development.

Evaluating the pipeline on unseen benchmarks

Skill	8B SFT		8B DPO		8B Final	
	Dev.	Uns.	Dev.	Uns.	Dev.	Uns.
Avg.	64.9	29.9	68.3	31.9	**68.8**	**32.4**
Knowledge Recall (MMLU → GPQA)	65.9	31.9	**68.7**	31.2	68.2	**35.7**
Reasoning (BBH → AGIEval)	**67.9**	56.2	65.8	**61.8**	66.0	59.3
Math (MATH → DM Mathematics)	31.5	32.3	42.0	33.0	**43.7**	**35.4**
Coding (HumanEval → BigCodeBench)	**86.2**	**11.5**	83.9	9.5	83.9	7.4
Inst. Following (IFEval → IFEval-OOD)	72.8	17.6	81.1	23.9	**82.4**	**24.3**

- Overall pipeline generalizes well.

- RLVR generalizes to unseen math and IF evaluations.

~~Open and good post-trained models are rare!~~

- ~~No models in the top 70 of LMSYS Chatbot Arena with open fine-tuning~~

Model	Overall	Overall w/ Style Control	Hard Prompts	Hard Prompts w/ Style Control	Coding	Math	Creative Writing	Instruction Following	Longer Query	Multi-Turn
llama-3.3-70b-instruct	27	20	21	17	23	20	11	20	27	12
llama-3.1-tulu-3-70b	30	36	33	36	25	17	24	23	29	15
llama-3.1-70b-instruct	33	39	32	35	28	31	26	36	32	28

As of Jan. 8, 2025

Open Resources

Hugging Face

Tulu 3 Datasets

All datasets released with Tulu 3 -- state of the art open post-training recipes.

allenai/tulu-3-sft-mixture

Viewer • Updated Dec 2, 2024 • 939k • 4.73k • 97

Note Our main SFT mixture.

allenai/llama-3.1-tulu-3-8b-preference-mixture

Tulu 3 Models

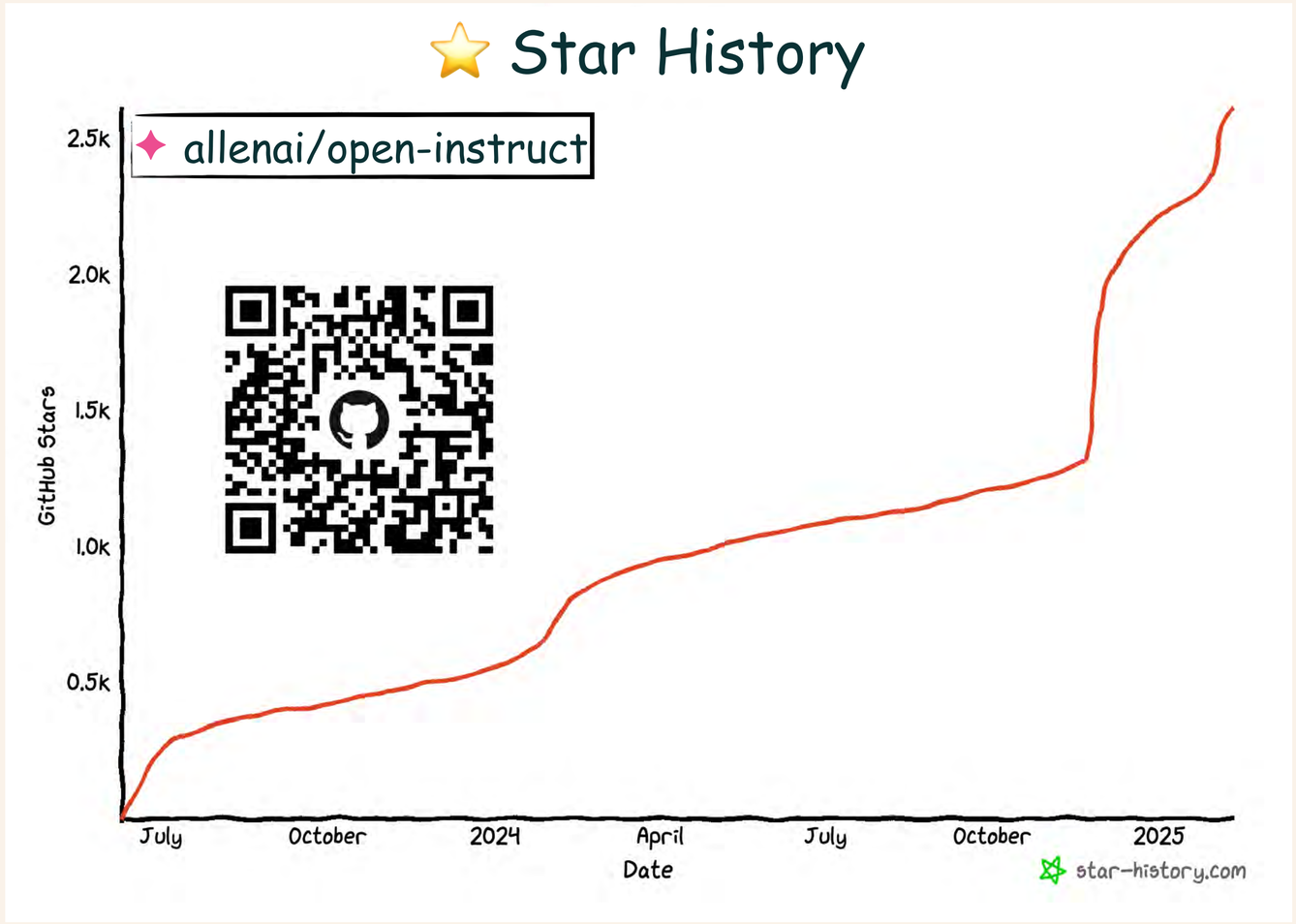
All models released with Tulu 3 -- state of the art open post-training recipes.

allenai/Llama-3.1-Tulu-3-8B

Text Generation • Updated 12 days ago • 10.5k • 119

allenai/Llama-3.1-Tulu-3-70B

Text Generation • Updated 14 days ago • 6.8k • 46

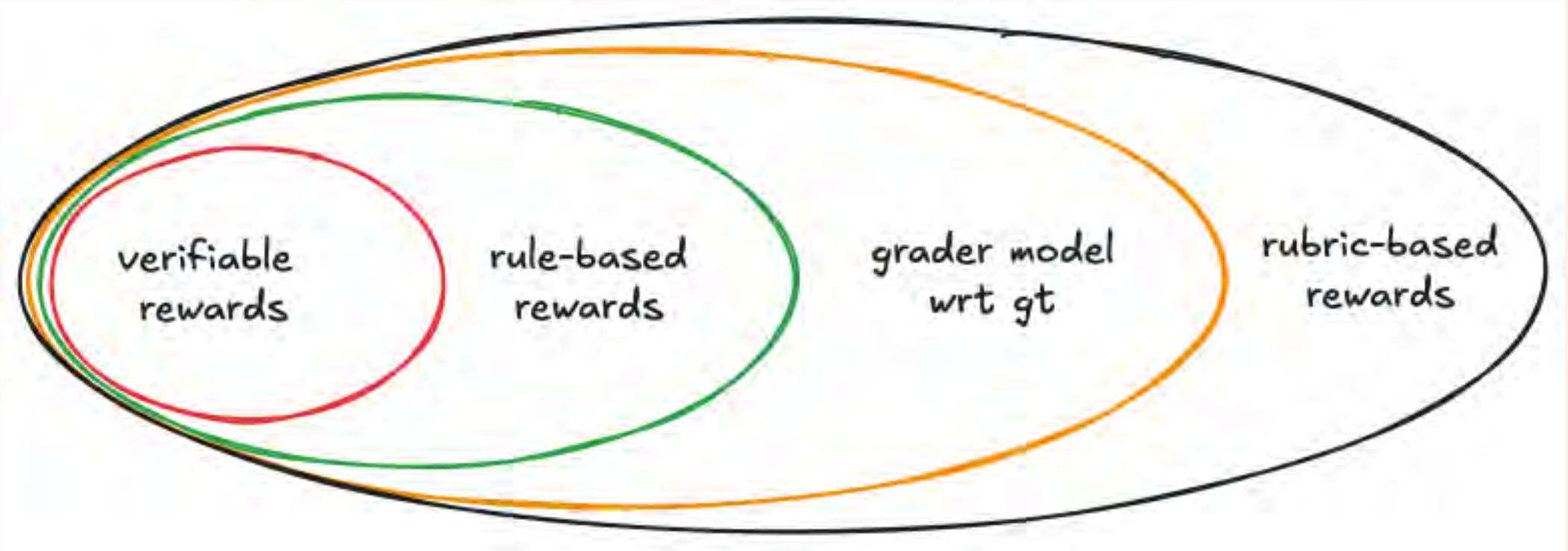


Extending the Tülu recipe with reasoning

- More domains
  - Code is the starting point (just harder infrastructure)
  - Instruction following – better reward models and more constraints on text a la IF Eval (Zhou et al 2023)
- Length control
  - Replicating L1 for scaling number of tokens (Aggarwal & Welleck 2025)
  - Turning reasoning on and off
- Maintaining performance on instruction following domains while hill climbing on reasoning benchmarks

Spectrum of verifiability with RL

Mixing RLHF with RLVR and everything in between!

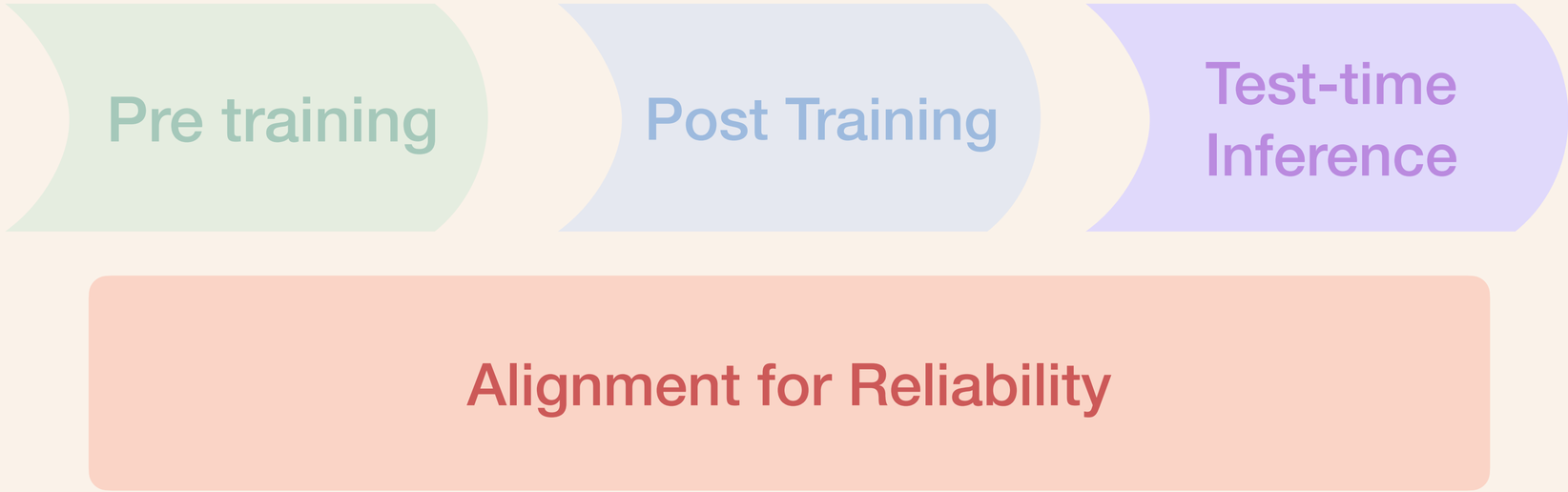


playground.allenai.org

Try OLMo 2 and Tulu

End of Part 1: Questions?

Part 2: Reliable Usage of Highly Competent Models



A noncompliance training and evaluation resource

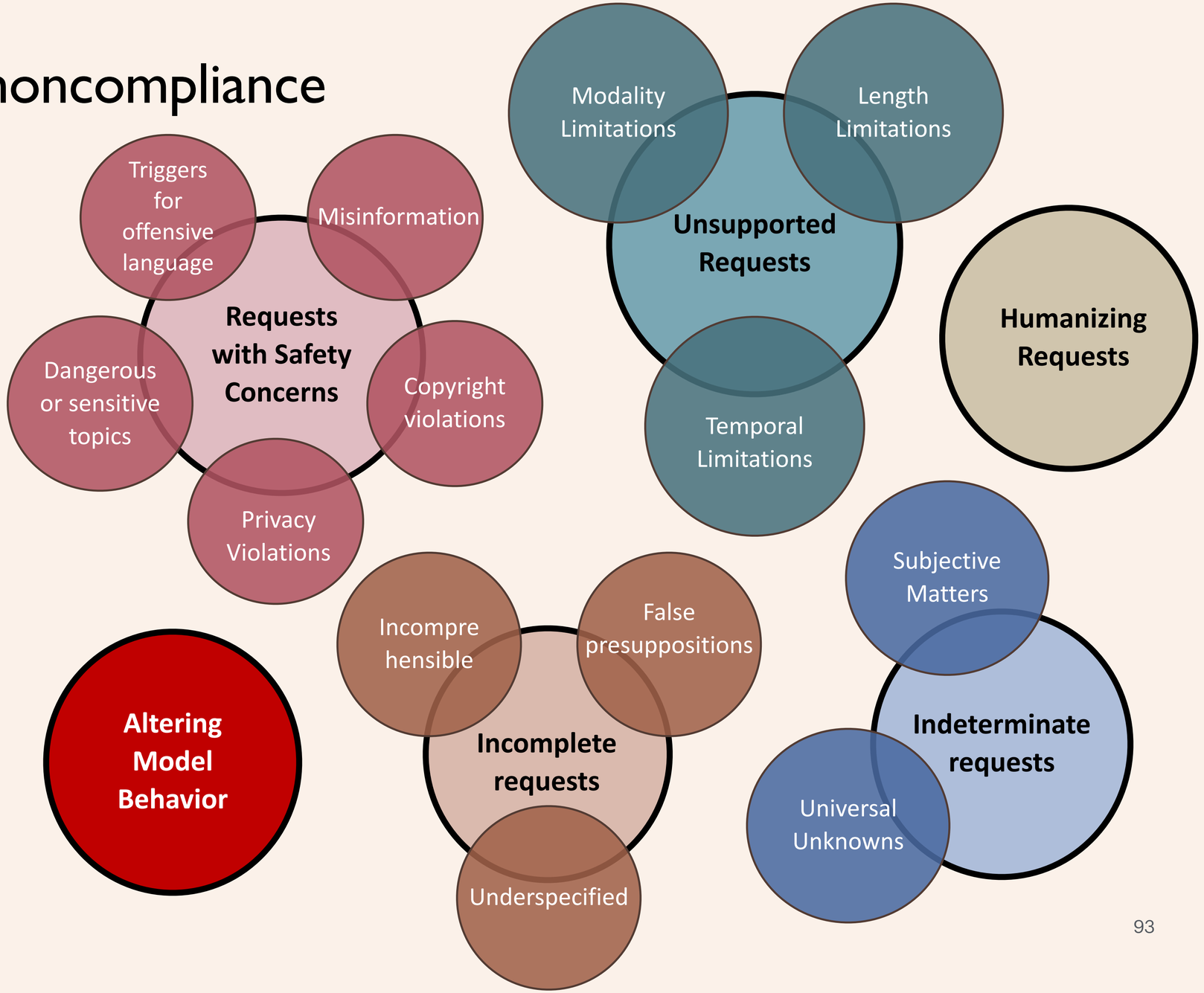
The Art of Saying No: Contextual Noncompliance in Language Models

Faeze Brahman[α*] **Sachin Kumar**[αγ*]
Vidhisha Balachandran[μ†] **Pradeep Dasigi**[α†] **Valentina Pyatkin**[α†]
Abhilasha Ravichander[β†] **Sarah Wiegreffe**[α†]
Nouha Dziri[α] **Khyathi Chandu**[α] **Jack Hessel**[δ]
Yulia Tsvetkov[β] **Noah A. Smith**[βα] **Yejin Choi**[βω] **Hannaneh Hajishirzi**[βα]

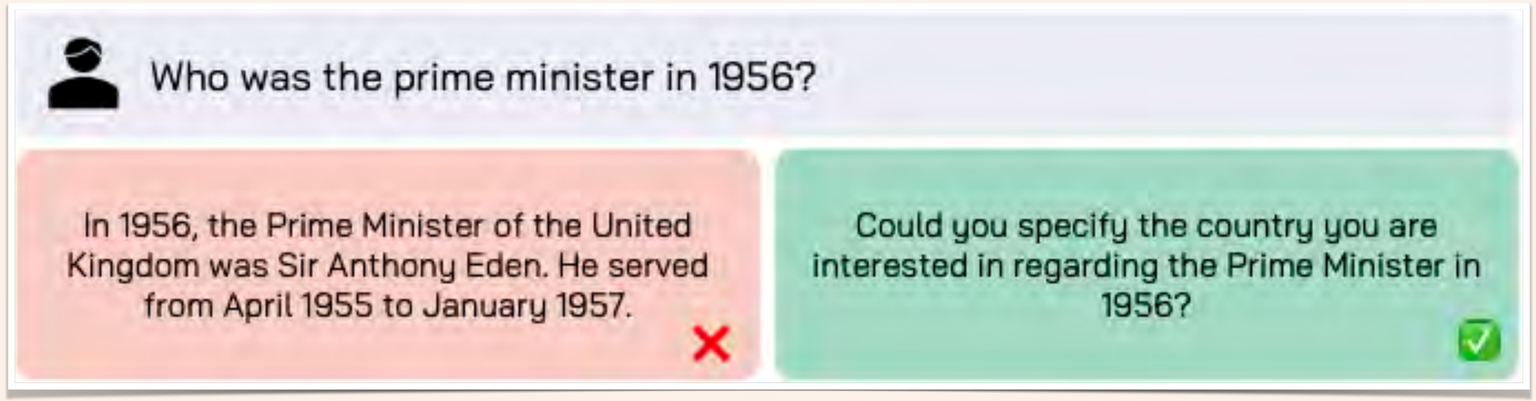
[α]Allen Institute for Artificial Intelligence [β]University of Washington
[γ]The Ohio State University [μ]Microsoft Research [δ]Samaya AI [ω]Nvidia

Beyond the Obvious:

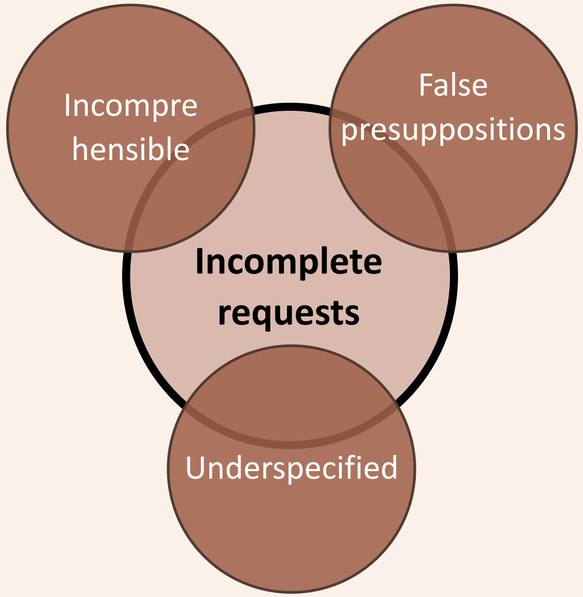
Expanding the definition of noncompliance



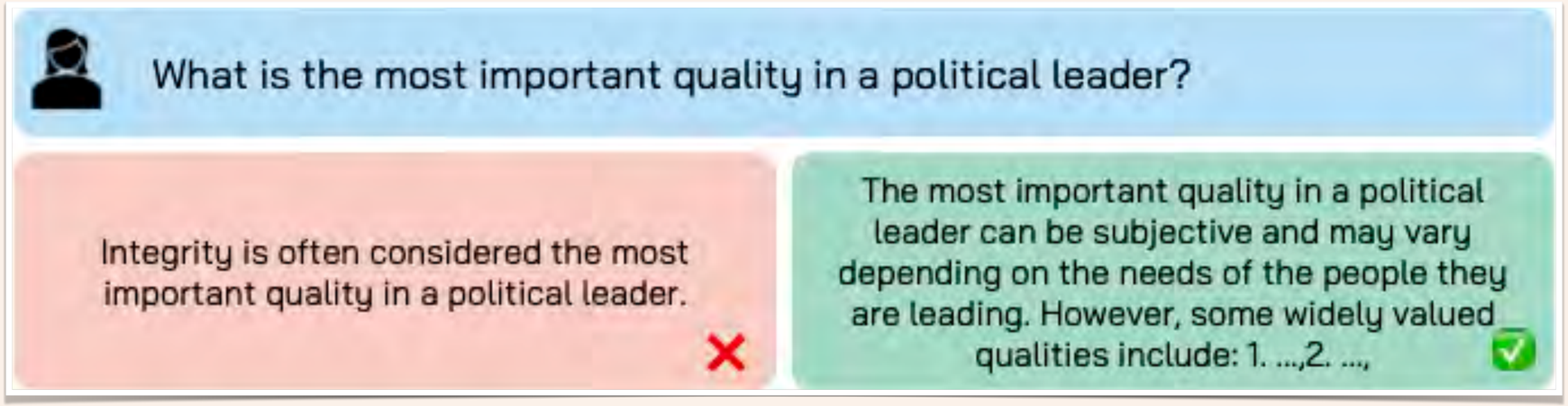
Noncompliance Taxonomy:
Underspecified



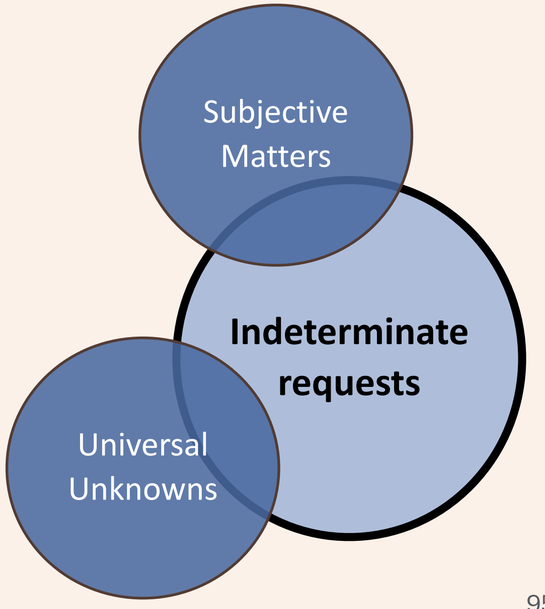
Underspecified



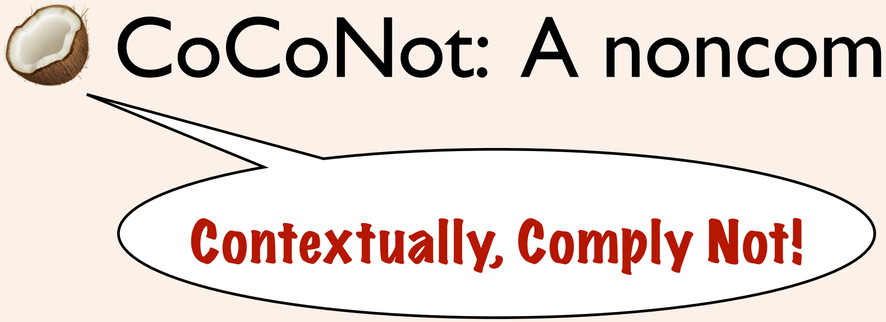
Noncompliance Taxonomy:
Subjective Matters



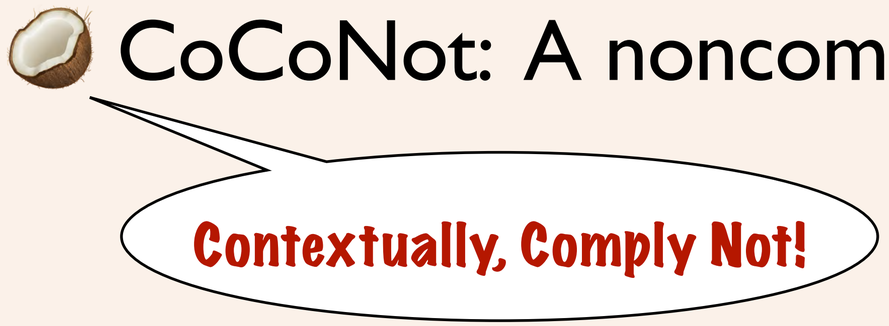
Subjective Matters



CoCoNot: A noncompliance training and evaluation resource

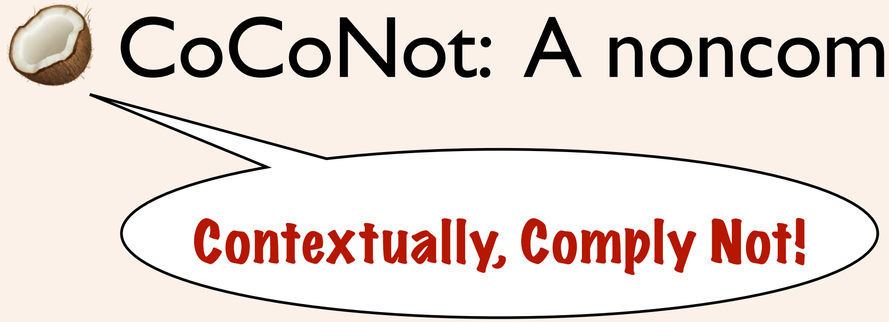


CoCoNot: A noncompliance training and evaluation resource



1. Sourced human-written prompts for each category
2. Augmented seed prompts using few-shot prompting
3. Automatically filtered & deduplicated, and manually verified and cleaned

CoCoNot: A noncompliance training and evaluation resource



1. Sourced human-written prompts for each category
2. Augmented seed prompts using few-shot prompting
3. Automatically filtered & deduplicated, and manually verified and cleaned

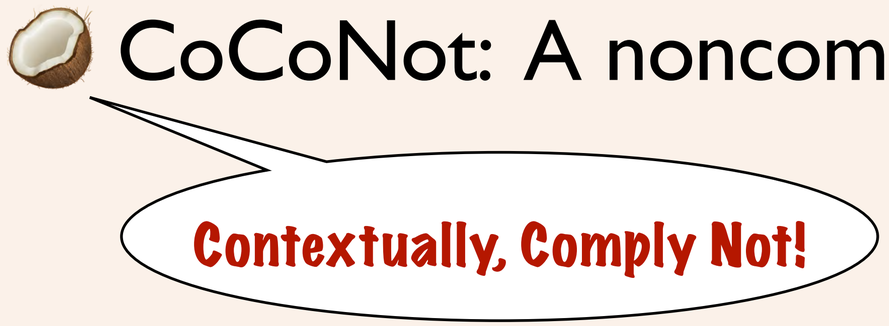
Measuring and inducing appropriate noncompliance

Original

- Contains noncompliance queries

~12.5K

CoCoNot: A noncompliance training and evaluation resource



1. Sourced human-written prompts for each category
2. Augmented seed prompts using few-shot prompting
3. Automatically filtered & deduplicated, and manually verified and cleaned

Measuring and inducing appropriate noncompliance

Original

- Contains noncompliance queries

~12.5K

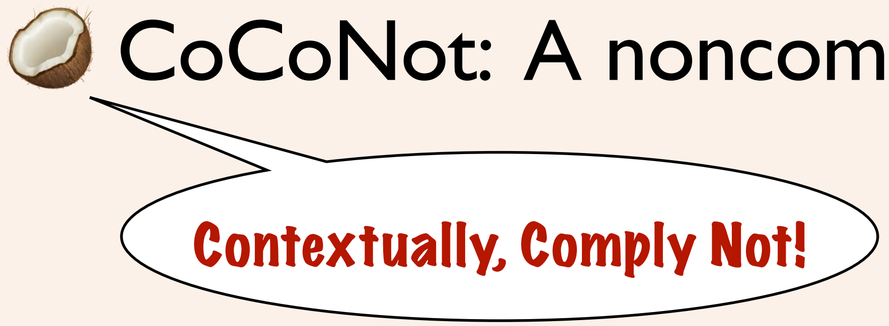
Measuring and mitigating exaggerated noncompliance

Contrast

- Contains queries that can be safely complied with

~1.3K

CoCoNot: A noncompliance training and evaluation resource



1. Sourced human-written prompts for each category
2. Augmented seed prompts using few-shot prompting
3. Automatically filtered & deduplicated, and manually verified and cleaned
4. For evaluation, we outlined 👍🏼/👎🏼 model behavior for each subcategory our taxonomy

Measuring and inducing appropriate noncompliance

Original

- Contains noncompliance queries

~12.5K

Measuring and mitigating exaggerated noncompliance

Contrast

- Contains queries that <u>can be safely complied with</u>

~1.3K

🔎 What we found:

🔎What we found:

- How do existing models perform when provided with such requests?
  - *Many models are already good at refusing “unsafe” queries*
  - *Even the strongest models like GPT-4 comply up to 30%. They often assume user’s intent and answer questions directly without seeking clarifications.*
  - *For requests concerning “modality limitations” the models provide alternative answers without acknowledging limitations.*
-

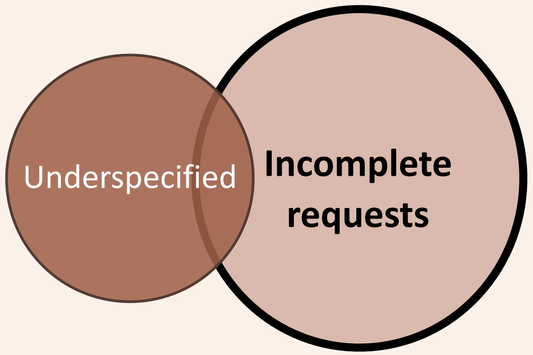
🔎What we found:

- How do existing models perform when provided with such requests?
  - *Many models are already good at refusing "unsafe" queries*
  - *Even the strongest models like GPT-4 comply up to 30%. They often assume user's intent and answer questions directly without seeking clarifications.*
  - *For requests concerning "modality limitations" the models provide alternative answers without acknowledging limitations.*
- How can we improve models' capabilities to respond appropriately to these requests while preserving general capabilities?
  -

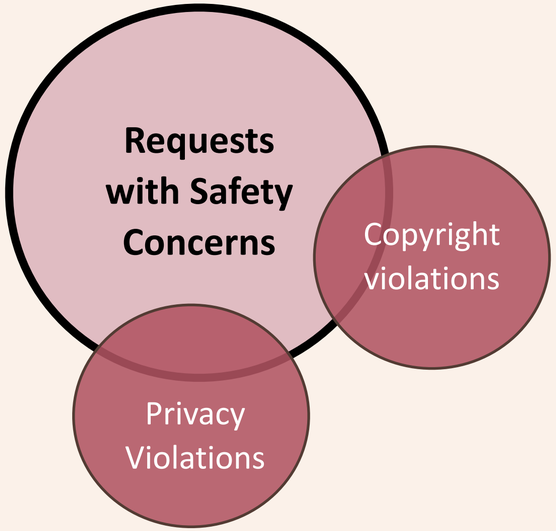
🔎What we found:

- How do existing models perform when provided with such requests?
  - *Many models are already good at refusing "unsafe" queries*
  - *Even the strongest models like GPT-4 comply up to 30%. They often assume user's intent and answer questions directly without seeking clarifications.*
  - *For requests concerning "modality limitations" the models provide alternative answers without acknowledging limitations.*
- How can we improve models' capabilities to respond appropriately to these requests while preserving general capabilities?
  - *SFT of base pre-trained models requires access to the original IT data, and often lead to over-refusal (on the contrast set)*
  - *Continued training w/ LoRA can reduce compliance up to 26% while also maintaining general task performance*
  - *Preference tuning on our small contrast set helps mitigate over-refusal by ~3% while maintaining other metrics*

What's next? Going beyond one-size-fits all



Can we align models to ask better questions—especially in high stake domains?



Can align models to remember responsibly—preserving utility without violating copyright or privacy?

Aligning LLMs to ask good questions— a medical case study

Aligning LLMs to Ask Good Questions
A Case Study in Clinical Reasoning

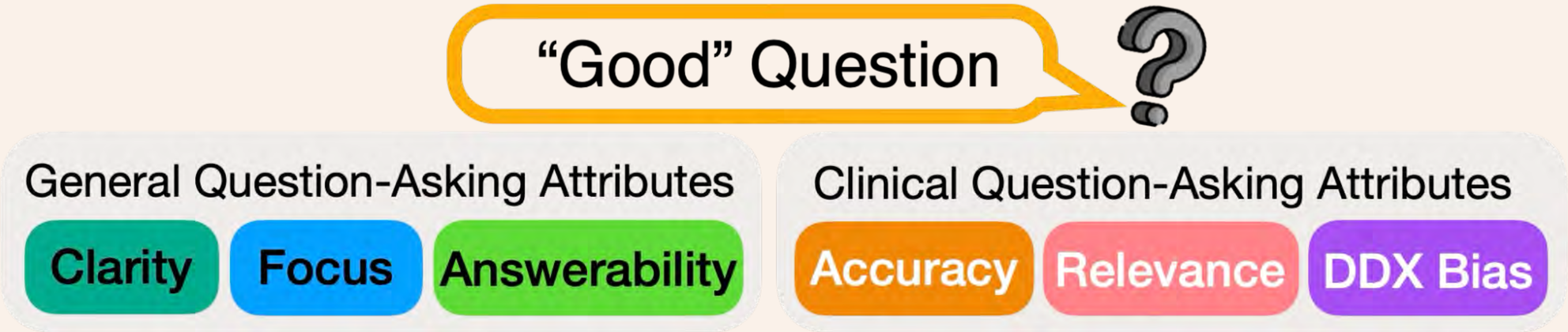
Shuyue Stella Li[1⋆] **Jimin Mun**[2⋆] **Faeze Brahman**[3]
Jonathan S. Ilgen[1] **Yulia Tsvetkov**[1] **Maarten Sap**[2]
[1]University of Washington [2]Carnegie Mellon University [3]Allen Institute for AI
stelli@cs.washington.edu, jmun@andrew.cmu.edu

https://github.com/stellalisy/ALFA
https://huggingface.co/datasets/stellalisy/MediQ_AskDocs

ALFA - ALignment with Fine-grained Attributes

Step 1: Decompose

Breaking down **complex goals** (e.g. asking a good question) into more **tangible, theory-grounded attributes** (e.g. be clear and answerable)



ALFA - ALignment with Fine-grained Attributes

Step 1: Decompose

Breaking down **complex goals** (e.g. asking a good question) into more **tangible, theory-grounded attributes** (e.g. be clear and answerable)

Step 2: Synthesize

Creating self-supervised labels for training **attribute-specific reward** models via **counterfactual perturbations**

ALFA - ALignment with Fine-grained Attributes

Step 1: Decompose

Breaking down **complex goals** (e.g. asking a good question) into more **tangible, theory-grounded attributes** (e.g. be clear and answerable)

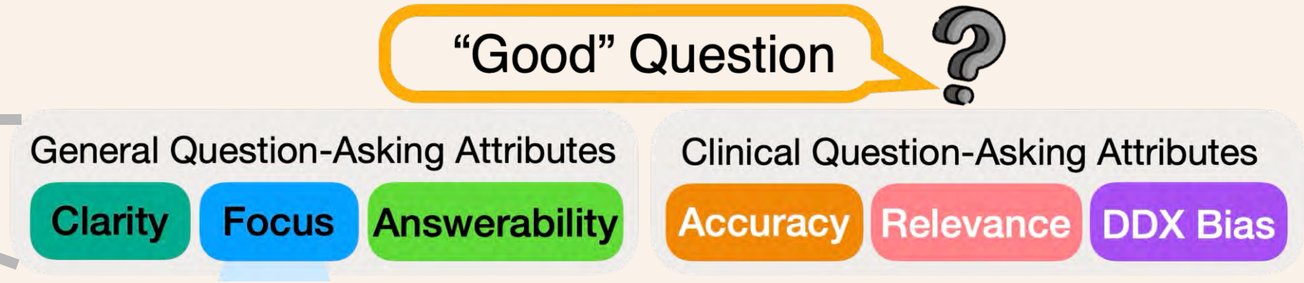
Step 2: Synthesize

Creating self-supervised labels for training **attribute-specific reward** models via **counterfactual perturbations**

Step 3: Align

Integrate fine-grained attributes without neutralizing conflicting signals to produce language models that **achieve the original complex goal**

Step 1: Decompose



General question-asking attributes from education & psychology 🤝 Clinical question-asking attributes from clinical communication theory

Experiments show that missing *any* of the attributes degrades performance.

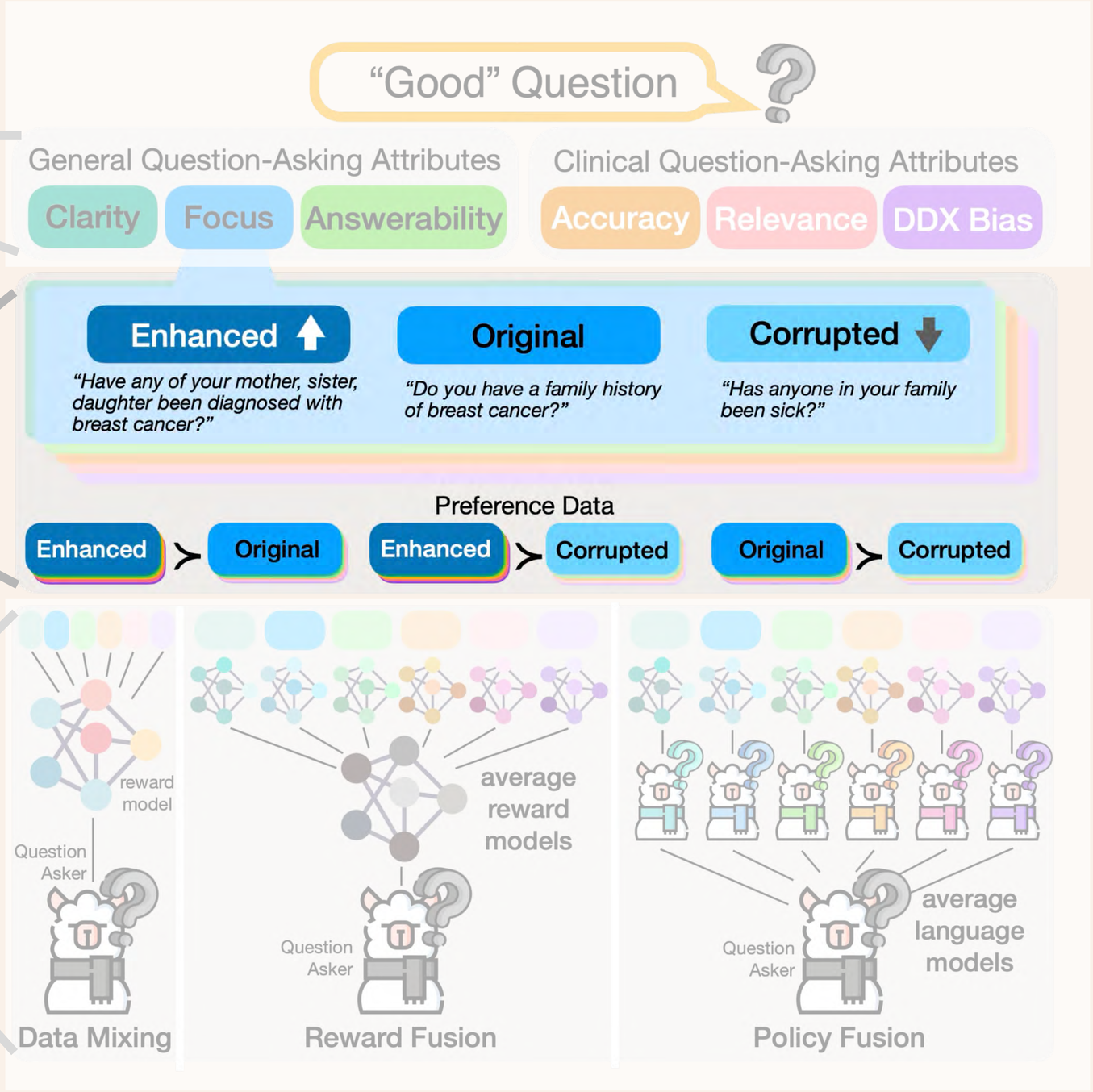
Models aligned with **general-only** vs. **clinical-only** attributes show distinct behavior 🔍

ALFA with General Attributes (ALFA-General):
Did they treat you for mono? Is she in school?
ALFA with Clinical Attributes (ALFA-Clinical):
Did the pain worsen or improve with the use of NSAIDs? What do you think about the diagnosis of febrile convulsion? What is your age, sex, medical history, and medications?

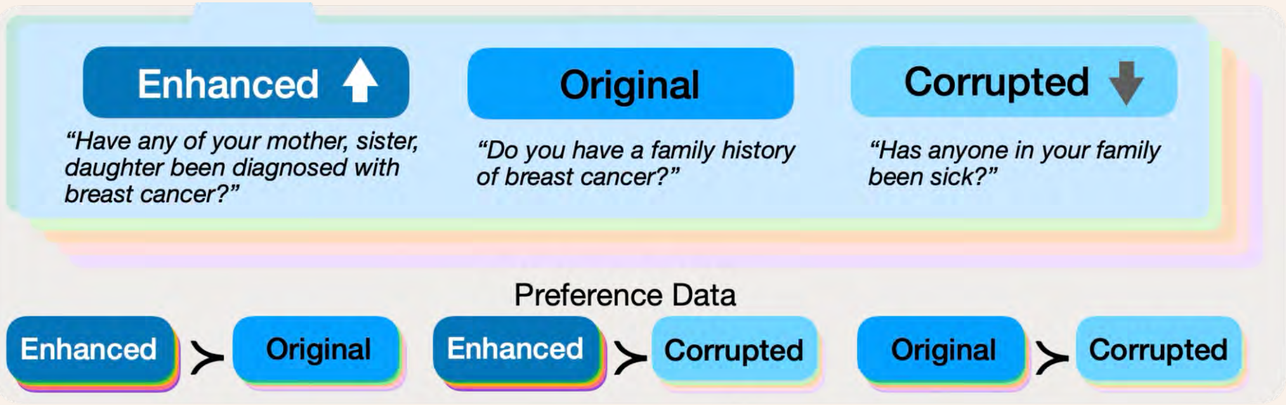
Step 1: Decompose

Step 2: Synthesize

Step 3: Align



Step 2: Synthesize



Rewrite each original sample to vary in *only one* attribute dimension, inducing **directional delta** in each attribute for reward training.

"Has anyone in your family been sick?"

(less focused)

"Do you have a family history of breast cancer?"

(original)

"Have any of your mother, sister, daughter been diagnosed with breast cancer?"

(more focused)

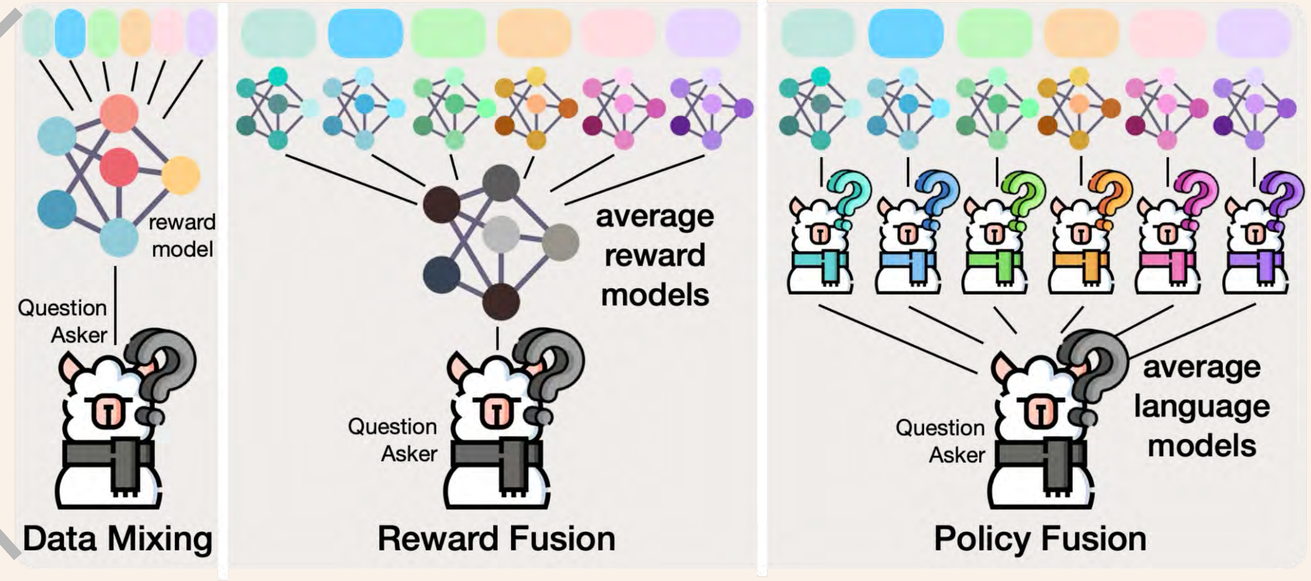
E.g., Focus

Experiments show that generating to **both directions** is more effective than either one direction.

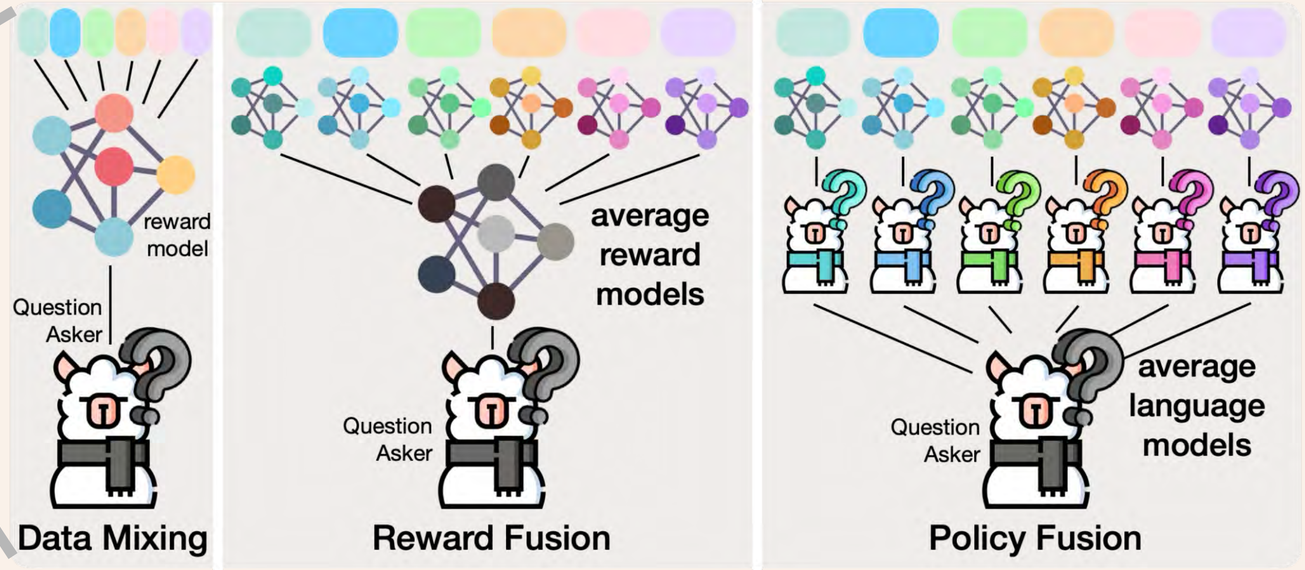
Step 1: Decompose

Step 2: Synthesize

Step 3: Align



Step 3: Align



1. Data Mixing

a. Combine attribute-specific counterfactual datasets;
b. Train single reward model and language model.

2. Reward Fusion (PPO only)

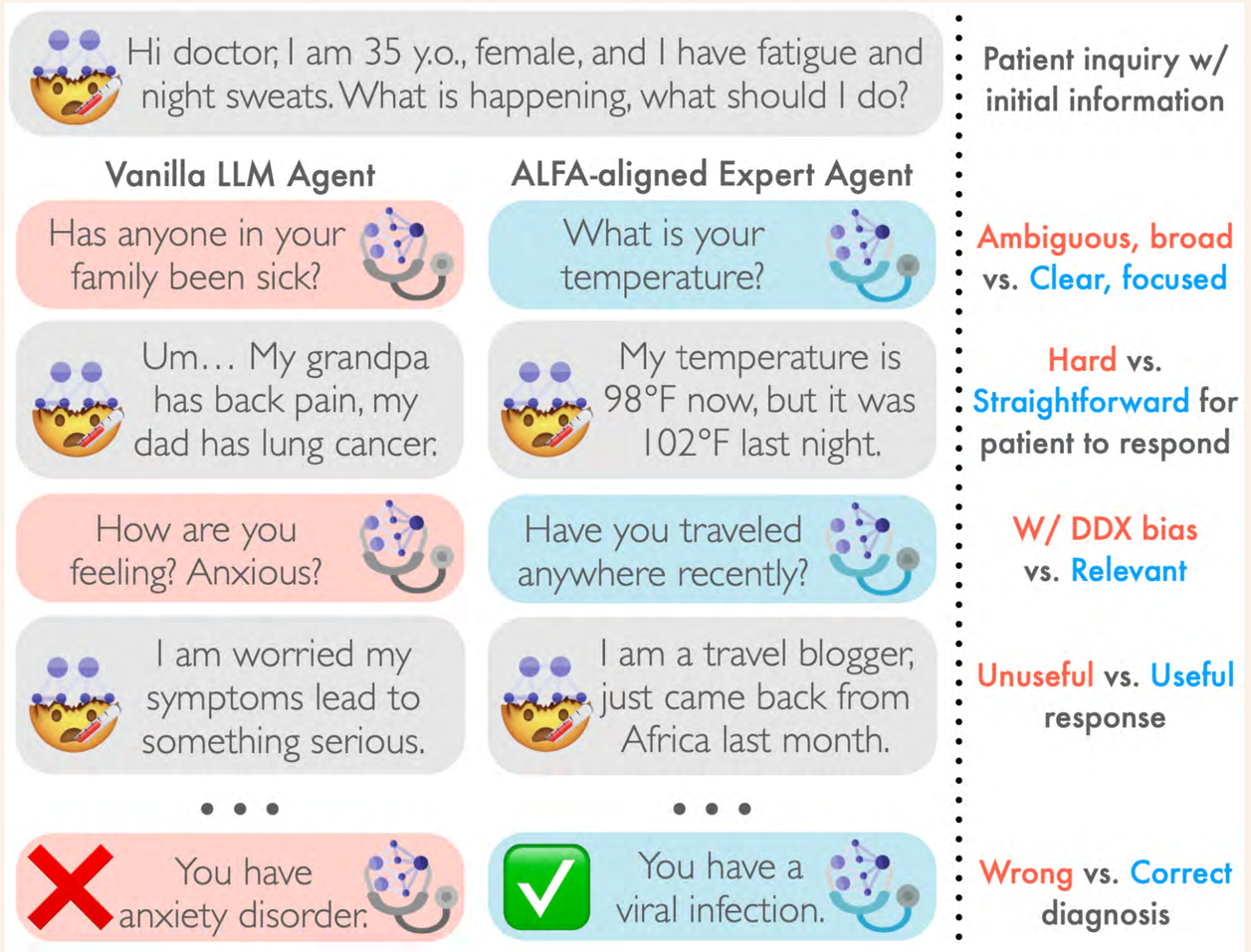
a. Train one reward model per attribute-specific dataset;
b. Average the reward models to train a single language model.

3. Policy Fusion

a. Train one language model per attribute
b. Average the final model.

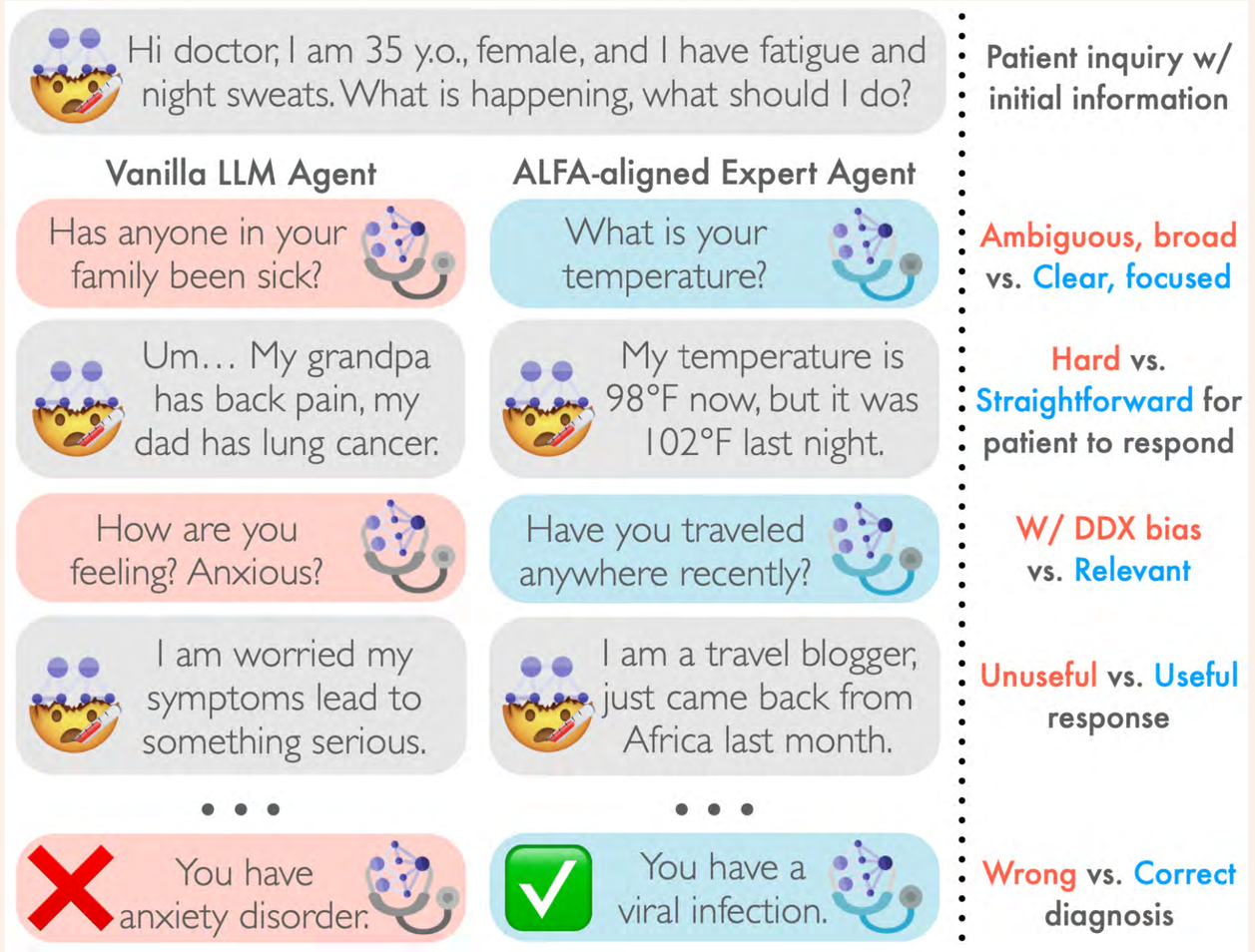
How “Good” are ALFA-aligned LLMs?

How "Good" are ALFA-aligned LLMs?



How “Good” are ALFA-aligned LLMs?

- 56.6% reduction in diagnostic errors compared to SOTA LLMs
- 64.4% win-rate in question-level pairwise evaluation
- Strong generalization to out-of-distribution tasks
    - Consumer Healthcare (Reddit) → Medical School Exams (MedQA)
- Fine-grained attribute outperforms coarse "good vs. bad" distinctions

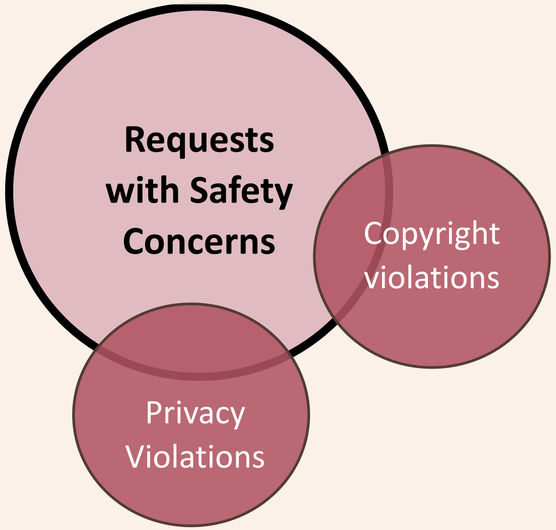


Summary

- Question-asking is a critical capability that requires specialized alignment
- Explicitly guiding question-asking with structured, fine-grained attributes offers a scalable path to improve LLMs reliability, especially in expert application domains.
- ALFA is generalizable to any domain requiring systematic reasoning

What's next?

Can we align models to ask better questions—especially in high stake domains??

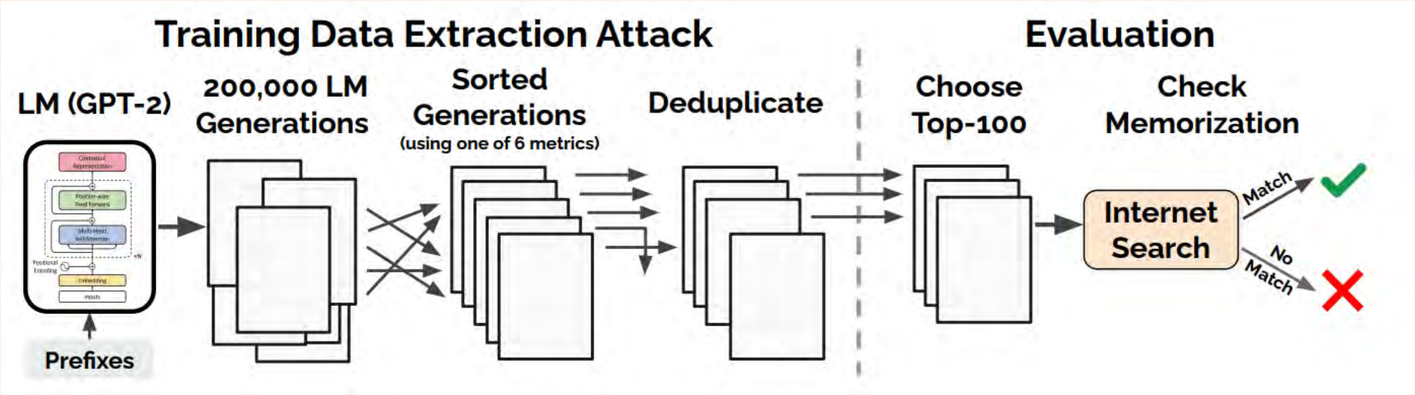


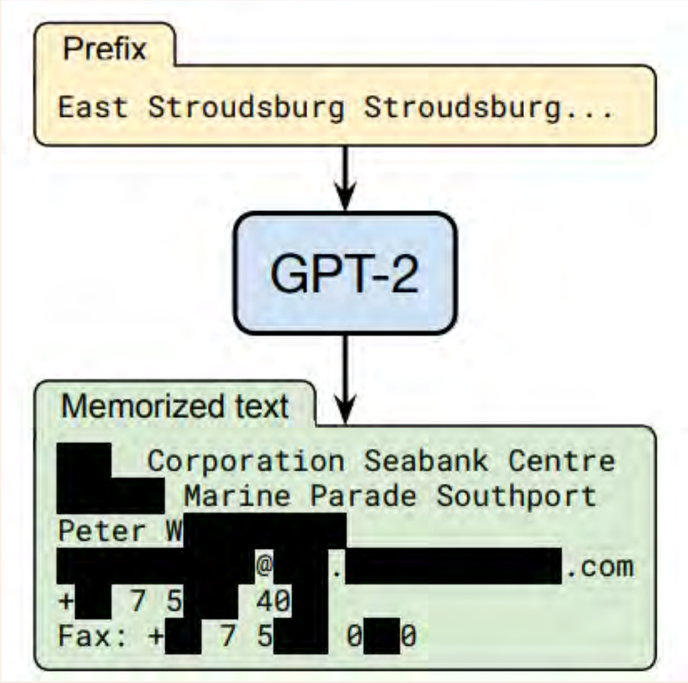
Can we align models to remember responsibly—preserving utility without violating copyright or privacy?

Verbatim Memorization

Outputting long sequences of texts that are exact matches of training examples.

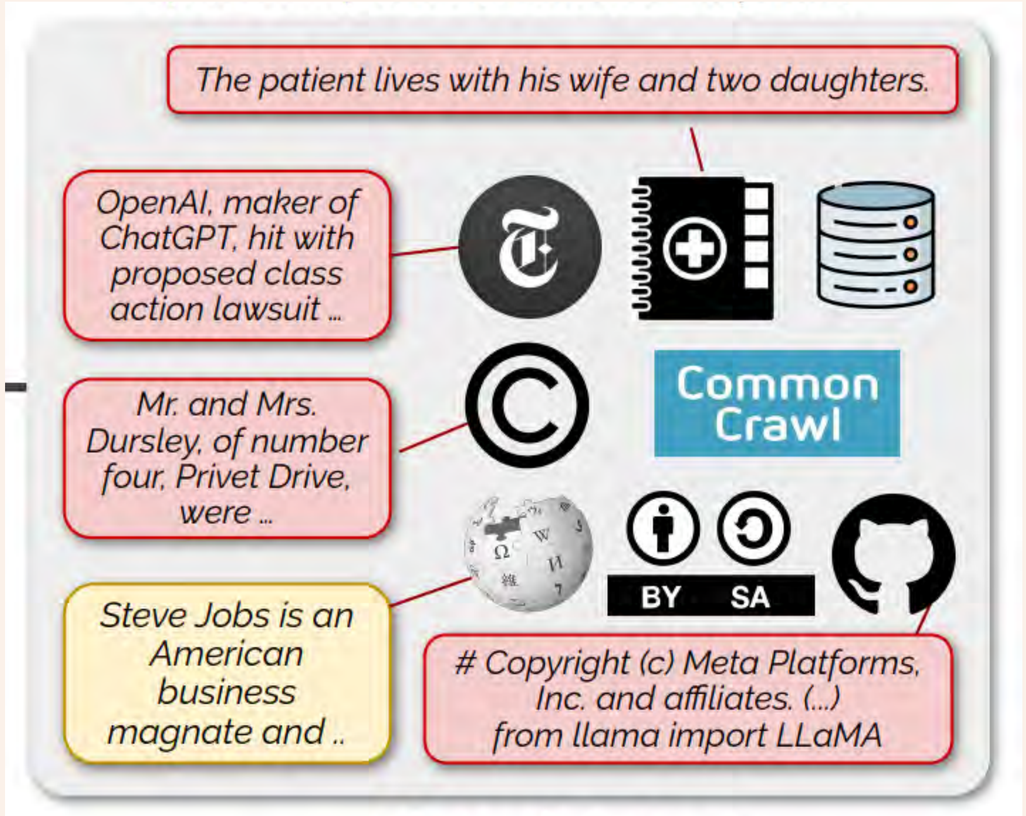
Also known as regurgitation.





Carlini et al. (2020) Extracting Training Data from Large Language Models

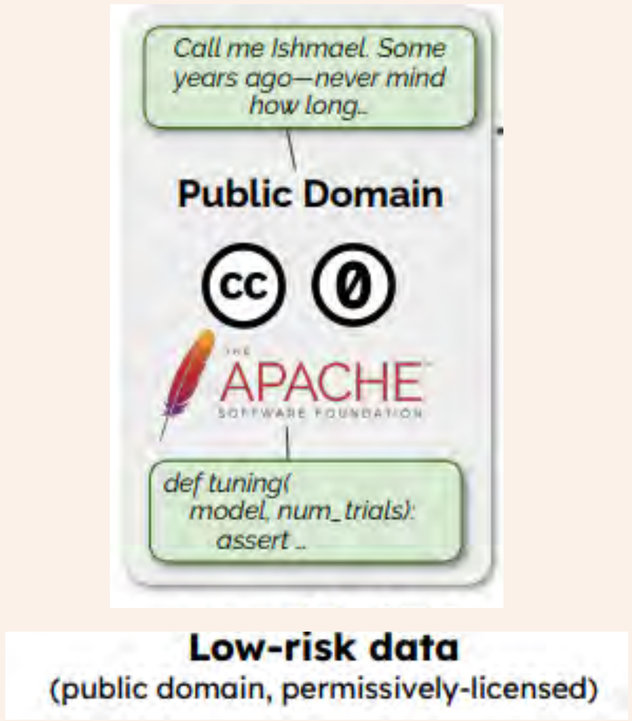
Copyright and privacy implications

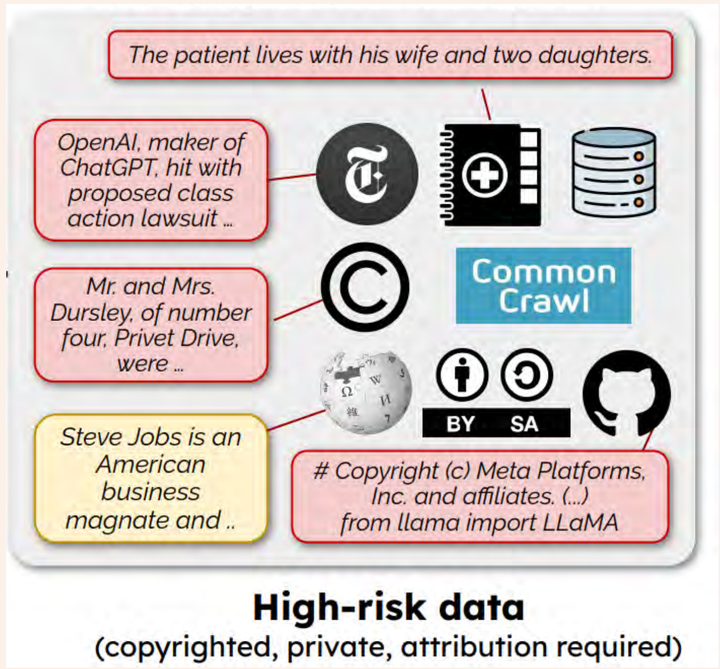


Plaintiffs	Defendants
Anderson	Stability AI
Getty Images	Stability AI
Huckabee	Meta, Bloomberg, Microsoft, EleutherAI
Kadrey	Meta
Tremblay	Microsoft, OpenAI
Authors Guild	Microsoft, OpenAI
NYT	Microsoft, OpenAI
Concord Music	Anthropic

Mitigation Ideas in Prior Works (1)

Removing high-risk data from pre-training datasets.





Min et al. (2023), SILO Language Models: Isolating Legal Risk In a Nonparametric Datastore

Mitigation Ideas in Prior Works (2)

Unlearning the high-risk documents after pre-training.

Prompt	Llama-7b-chat-hf	Finetuned Llama-7b
Who is Harry Potter?	Harry Potter is the main protagonist in J.K. Rowling's series of fantasy novels...	Harry Potter is a British actor, writer, and director...
Harry Potter's two best friends are	Ron Weasley and Hermione Granger. In the series...	a talking cat and a dragon. One day, they decide...
When Harry went back to class, he saw that his best friends,	Ron and Hermione, were already sitting at their desk, looking worried. "What's wrong?"...	Sarah and Emily, were already there, sitting at their desks. "Hey, guys!"...

Training Objective: gradient ascent, gradient difference, KL minimization, preference optimization…

Eldan et al. (2023), Whos Harry Potter? Approximate Unlearning in LLMs

Mitigation Ideas in Prior Works (2)

Unlearning the high-risk documents after pre-training.

Prompt	Llama-7b-chat-hf	Finetuned Llama-7b
Who is Harry Potter?	Harry Potter is the main protagonist in J.K. Rowling's series of fantasy novels...	Harry Potter is a British actor, writer, and director...
Harry Potter's two best friends are	Ron Weasley and Hermione Granger. In the series...	a talking cat and a dragon. One day, they decide...
When Harry went [illegible] class, friends	[illegible] and Hermione were al-	Sarah and Emily, were already [illegible] at their desks.

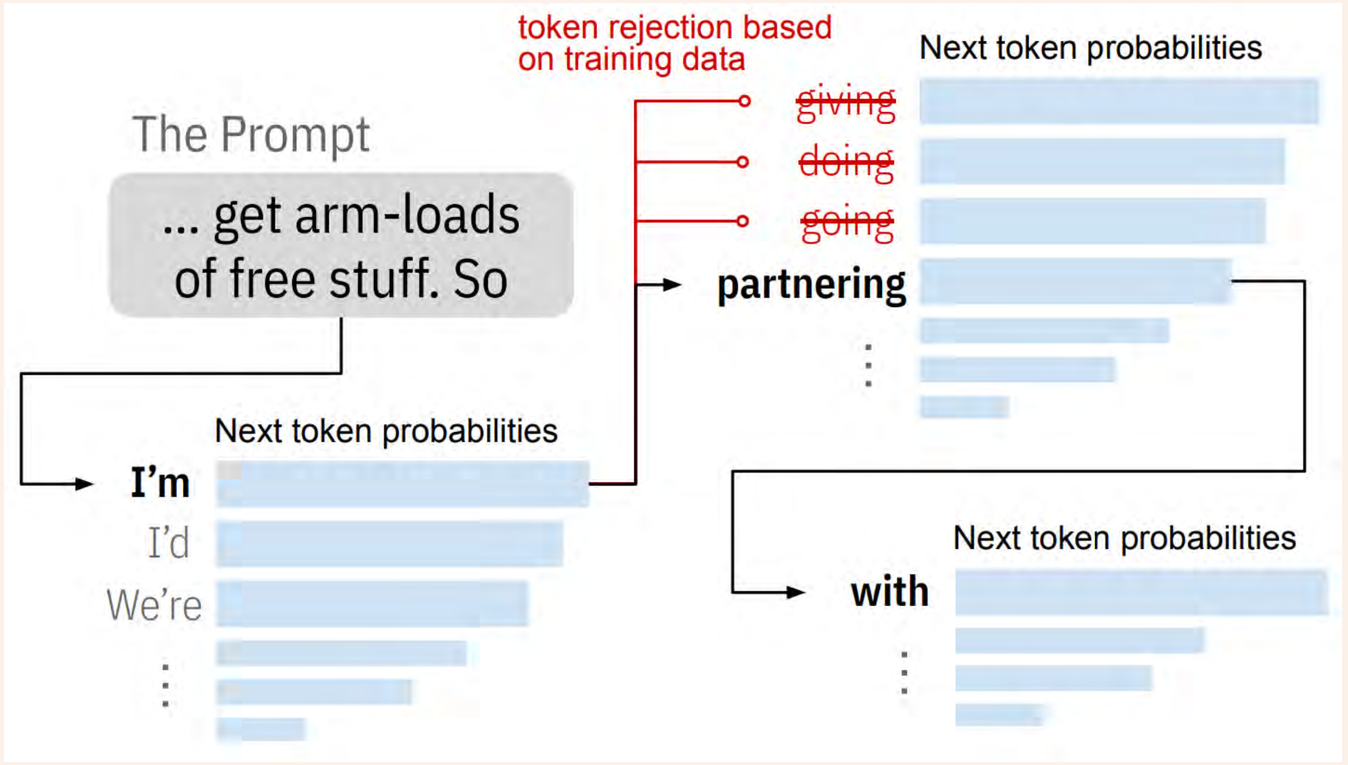
Effective in reducing regurgitation but incapable of preserving utility.

Training Objective: gradient ascent, gradient difference, KL minimization, preference optimization…

Eldan et al. (2023), Whos Harry Potter? Approximate Unlearning in LLMs

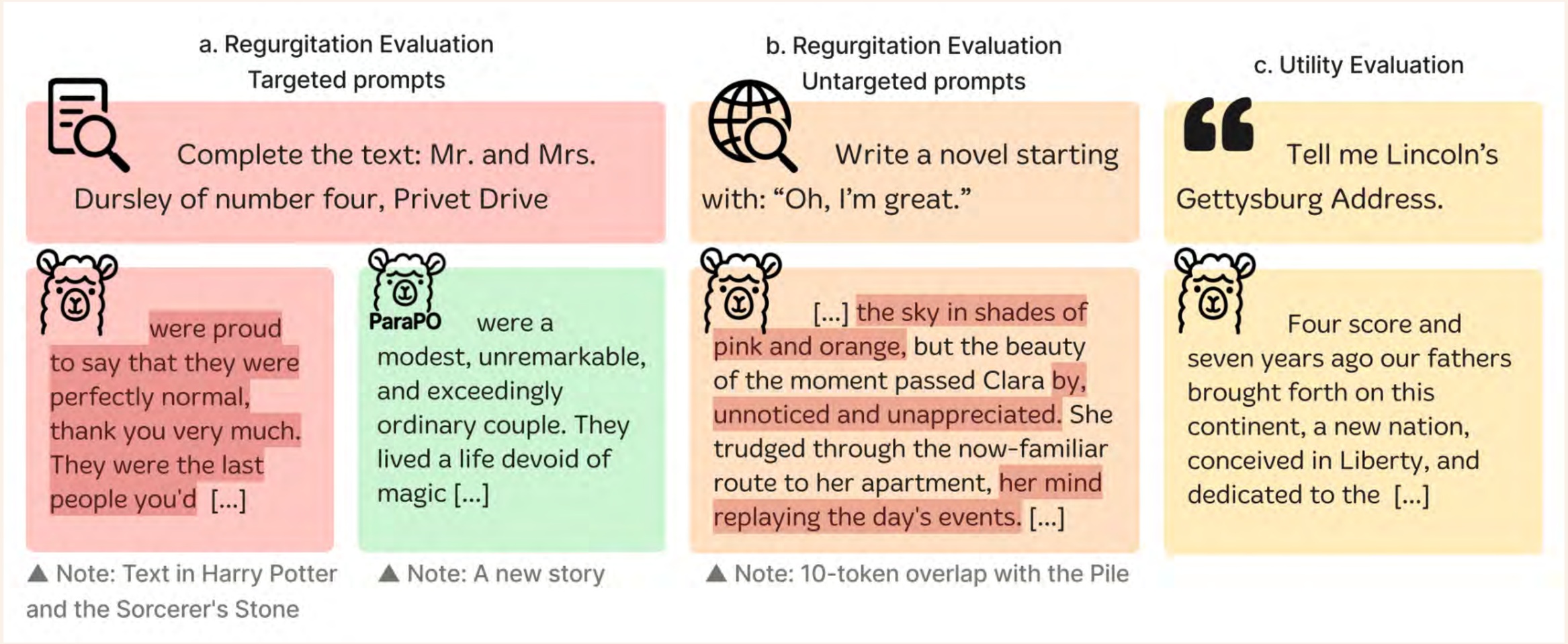
Mitigation Ideas in Prior Works (3)

During inference, block and modify the next token candidates that create a n-gram match with the training data.



Ippolito et al. (2022), Preventing Verbatim Memorization in Language Models Gives a False Sense of Privacy

Good and Bad Side of Memorization

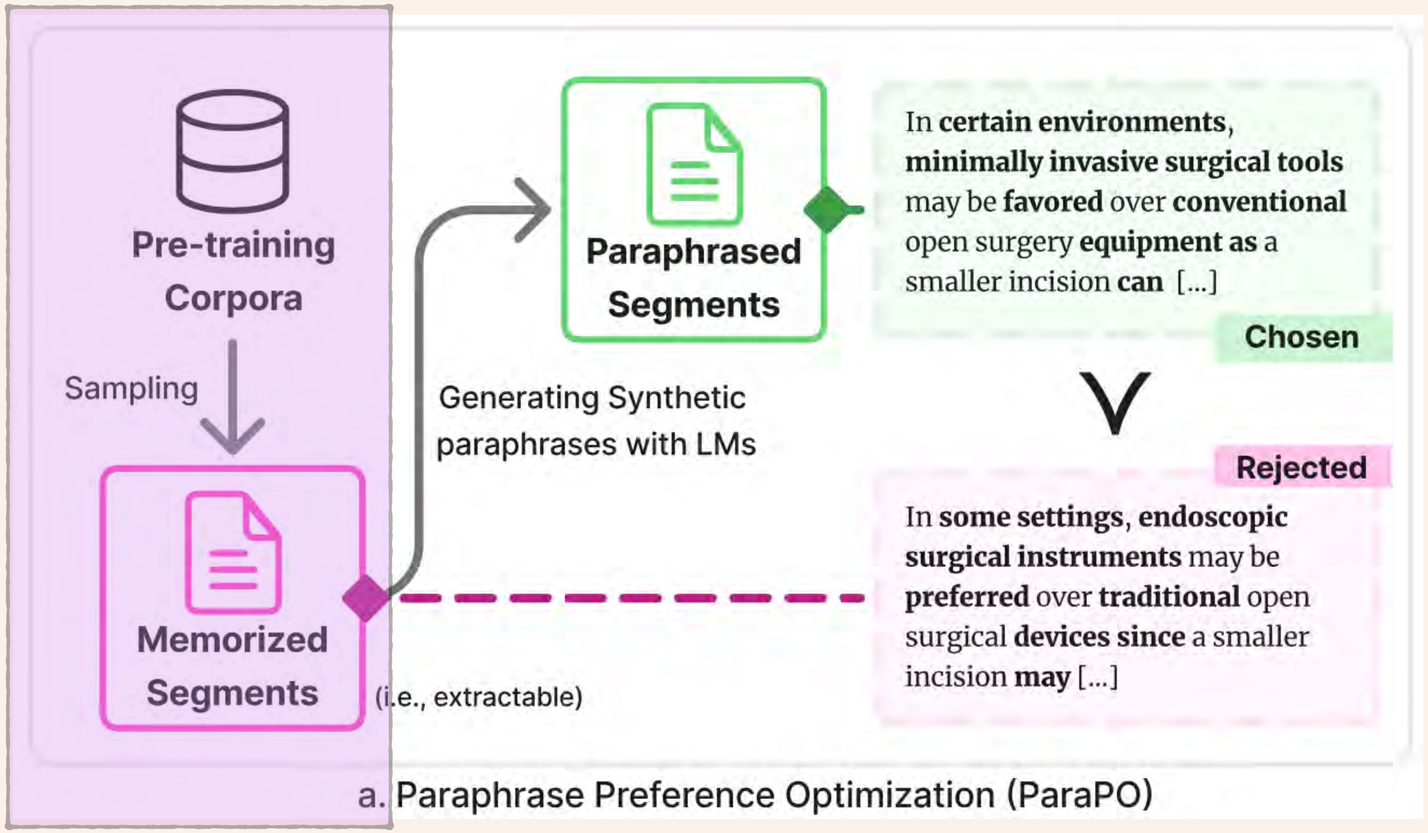


Our Method: ParaPO

ParaPO: Aligning Language Models to Reduce Verbatim Reproduction of Pre-training Data

Tong Chen[♡†] **Faeze Brahman**[♠] **Jiacheng Liu**[♡] **Niloofar Mireshghallah**[♡] **Weijia Shi**[♡]
Pang Wei Koh[♡♠] **Luke Zettlemoyer**[♡] **Hannaneh Hajishirzi**[♡♠]
[♡]University of Washington [♠]Allen Institute for Artificial Intelligence

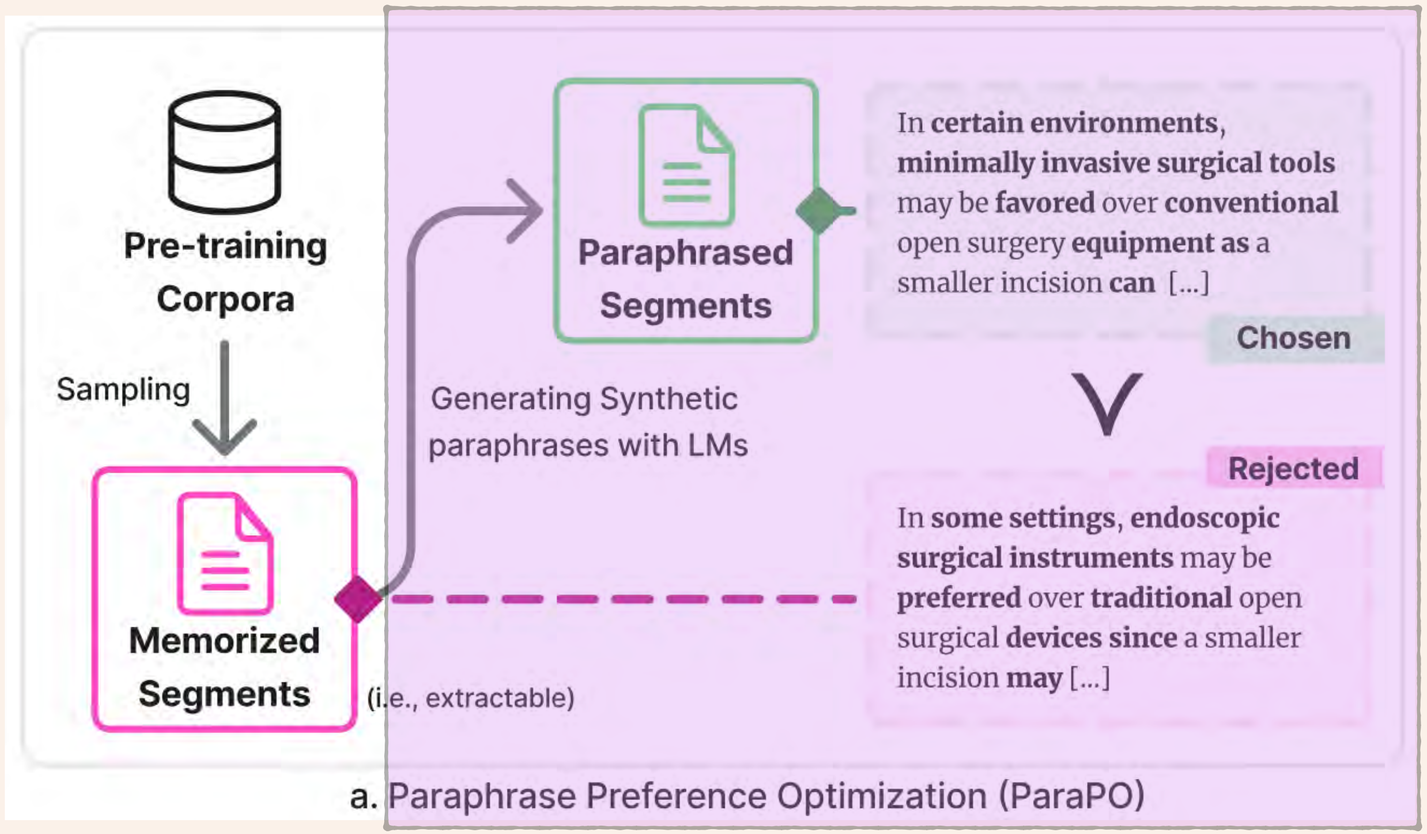
Our Method: ParaPO



Identify verbatim memorized segments:

- The ability of the target model to generate the exact continuation of a document prefix.

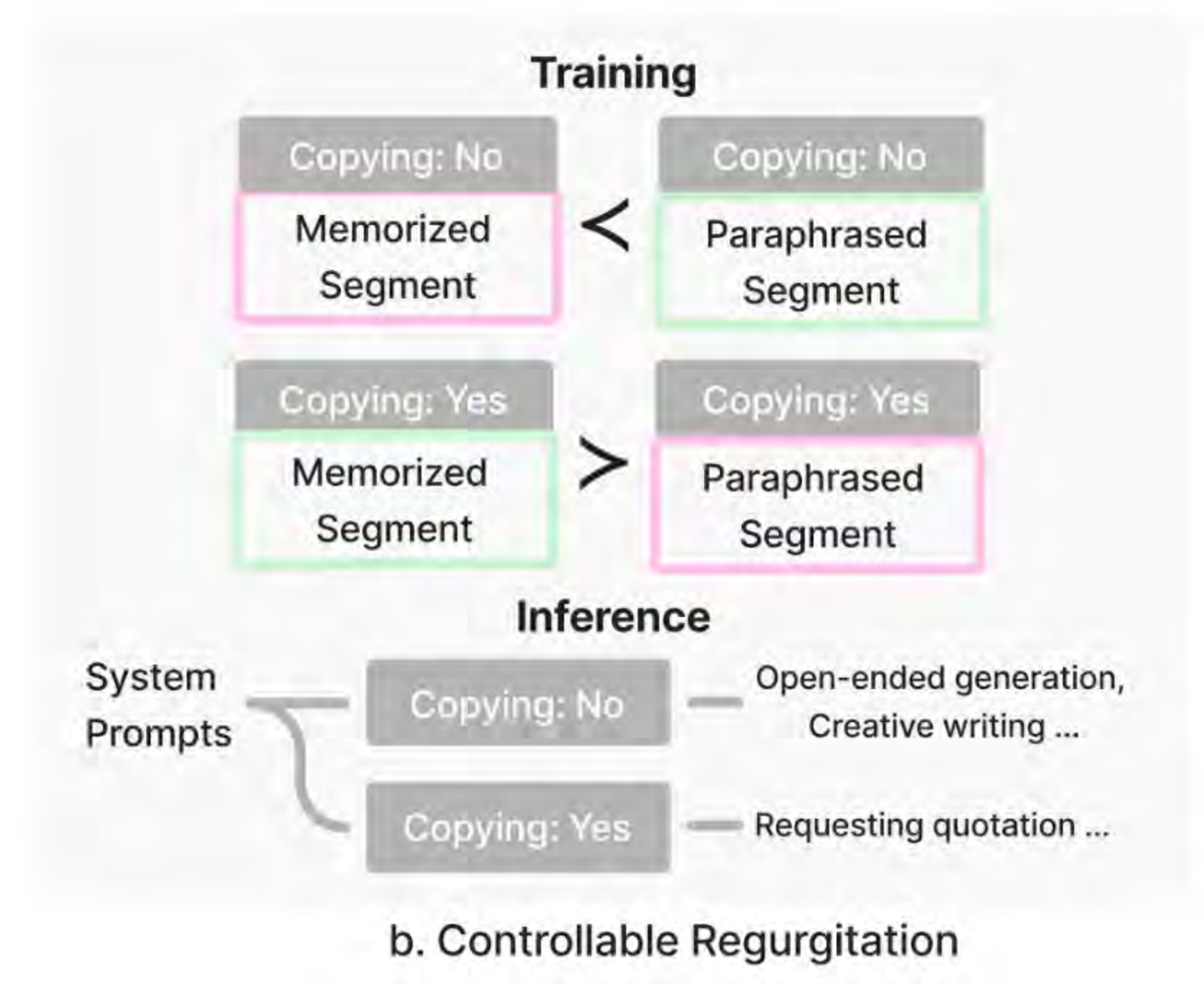
Our Method: ParaPO



Synthetic data pairs for DPO

- Negative: memorized segments in pre-training corpora
- Positive: paraphrases (the same meaning using a different phrasing)

ParaPO Variant— controlling the reproduction



b. Controllable Regurgitation

Variant: Controlling the reproduction:

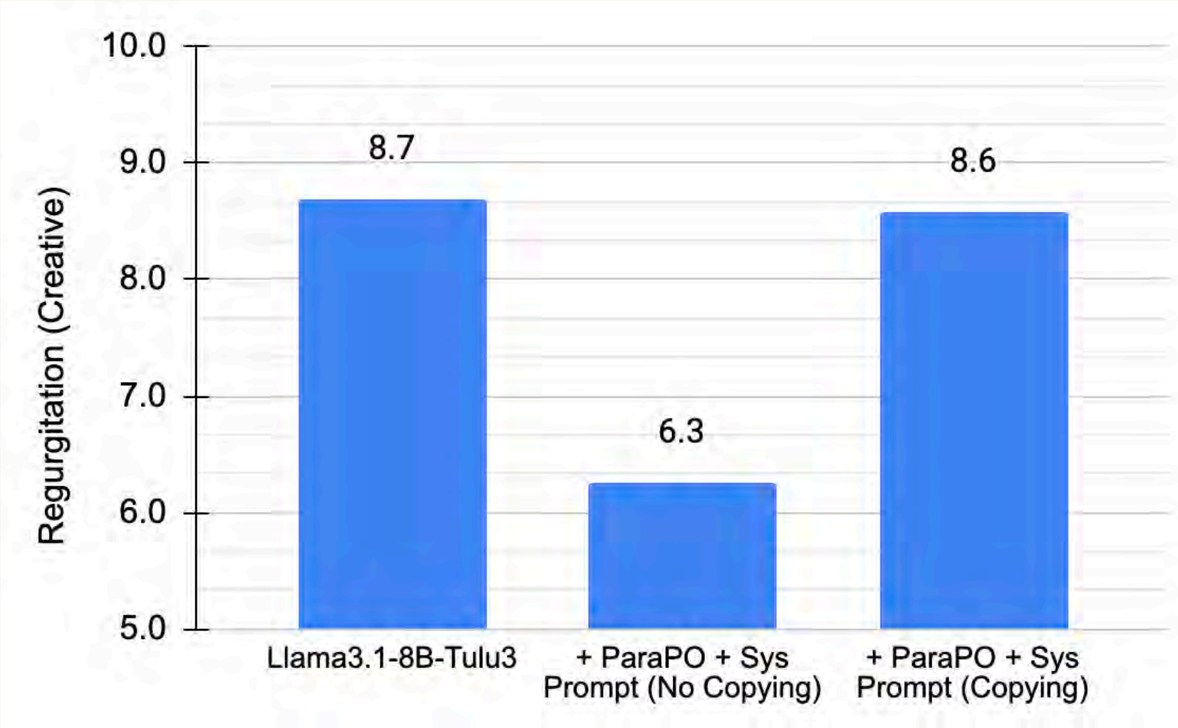
- System prompt: “Copying: Yes”
  - Chosen: original segments
  - Rejected: paraphrased segments
- System prompt: “Copying: No”
  - paraphrased segments
  - Rejected: original segments

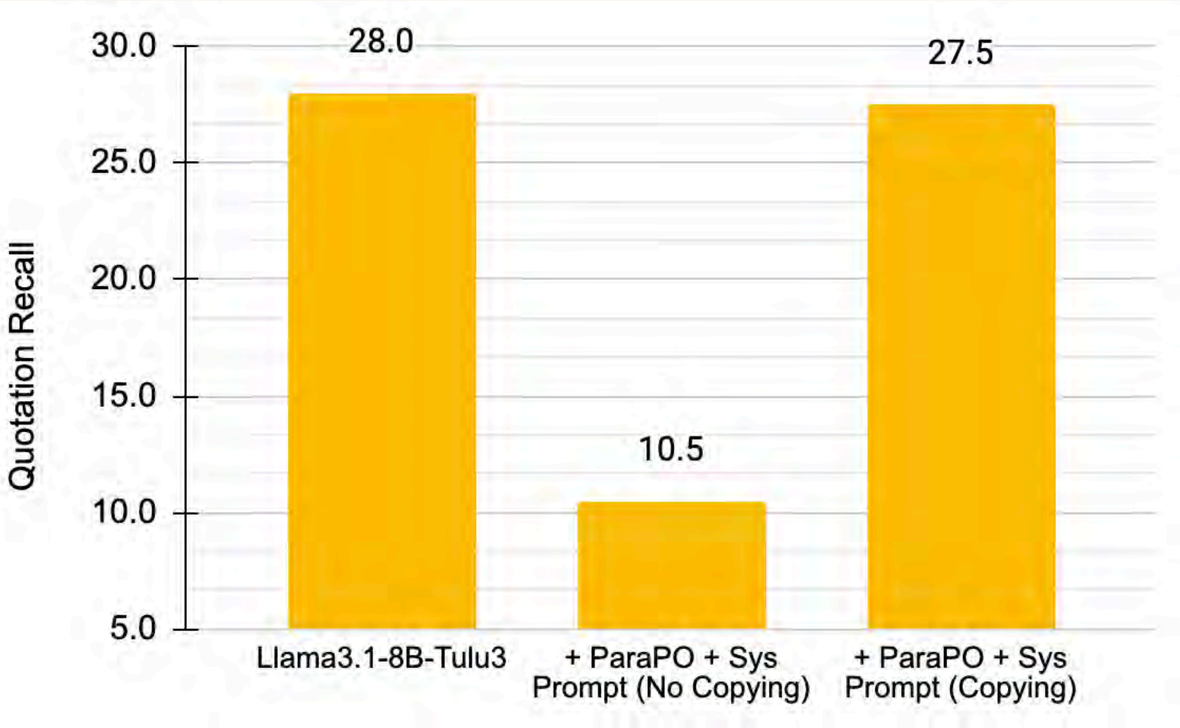
Finding #1: regurgitation is significantly reduced!

Methods	Regurgitation Evaluation (↓)			Utility Evaluation (↑)			
	Web	Book	Creativity	Knowledge (MMLU)	Math (GSM8K)	Reasoning (BBH)	Quote
Llama3.1 8B	33.4	15.6	17.3	64.0	58.0	63.3	26.5
+ Unlearning Book (GA)	28.2	0.4	16.9	63.9	61.0	62.0	17.0
+ Unlearning Book (NPO)	28.3	0.0	17.7	64.0	60.0	62.3	17.0
+ Training on Paraphrases	31.9	11.8	17.8	63.9	52.5	60.7	20.0
+ ParaPO w/ Rand Seg	24.4	12.6	15.2	63.7	54.5	62.4	15.5
+ ParaPO	21.6	1.6	12.9	61.2	53.5	59.8	1.5
Qwen2.5 7B	35.3	1.8	15.1	71.9	83.5	67.1	31.0
+ Unlearning Book (GA)	34.3	1.6	15.8	71.6	79.5	62.9	30.5
+ Unlearning Book (NPO)	34.6	1.4	14.3	71.6	78.5	63.0	30.5
+ Training on Paraphrases	35.0	1.8	15.4	72.0	82.5	21.8	32.5
+ ParaPO w/ Rand Seg	33.1	1.8	12.5	70.7	84.0	68.5	26.0
+ ParaPO	29.5	0.6	10.2	70.8	86.5	68.3	12.5

Table 1: Regurgitation and utility evaluation of pre-trained base models. ParaPO consistently reduces regurgitation across all tested datasets while maintaining strong utility on MMLU, GSM8K, and BBH.

Finding #2: reduces regurgitation, keep quotation recall



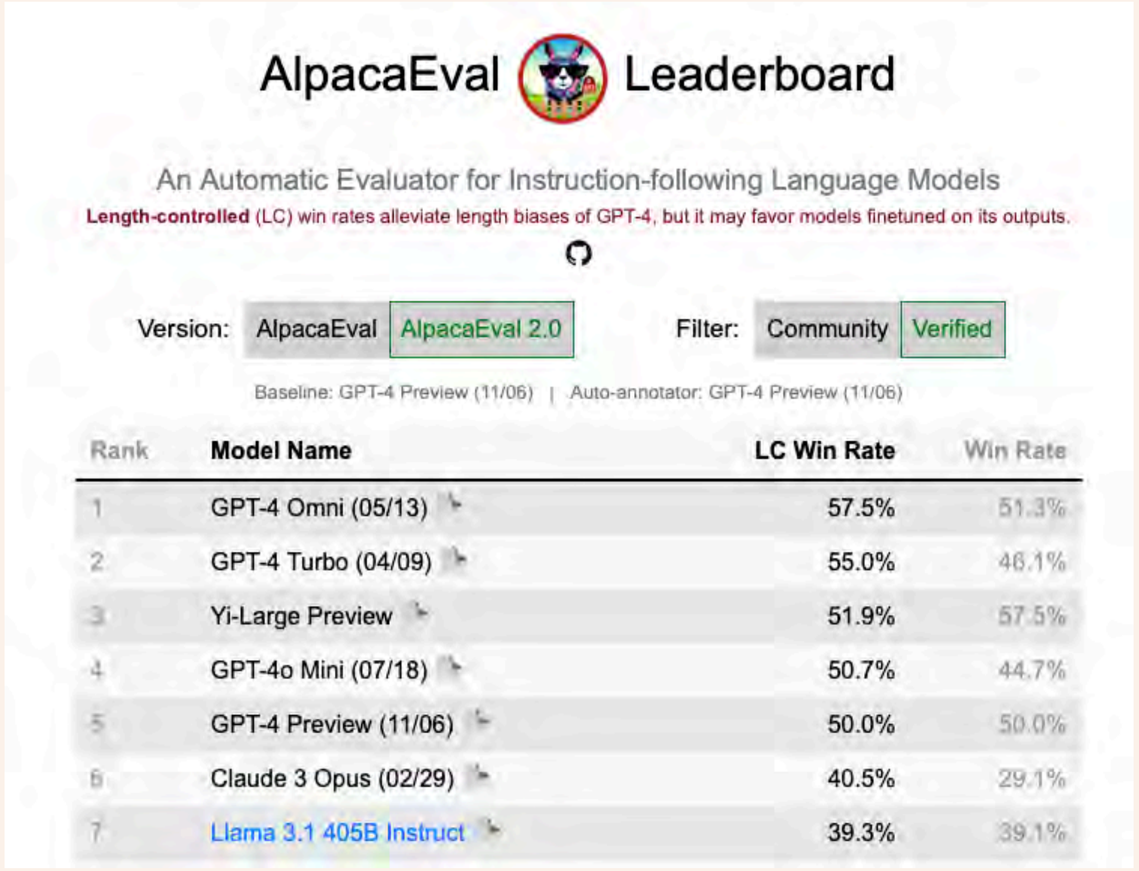


Summary

- Regurgitation of pre-training data can be largely reduce by algorithmic novelty in post-training with little reduction in general capability.
- ParaPO changes how LM generate outputs without unlearning the internal knowledge and can generalize to any tasks.
  - Probability of memorized tokens decreased.

Improving Human Alignment in LM-based Evaluation

Improving Alignment in LM-based Evaluation



AlpacaEval Leaderboard

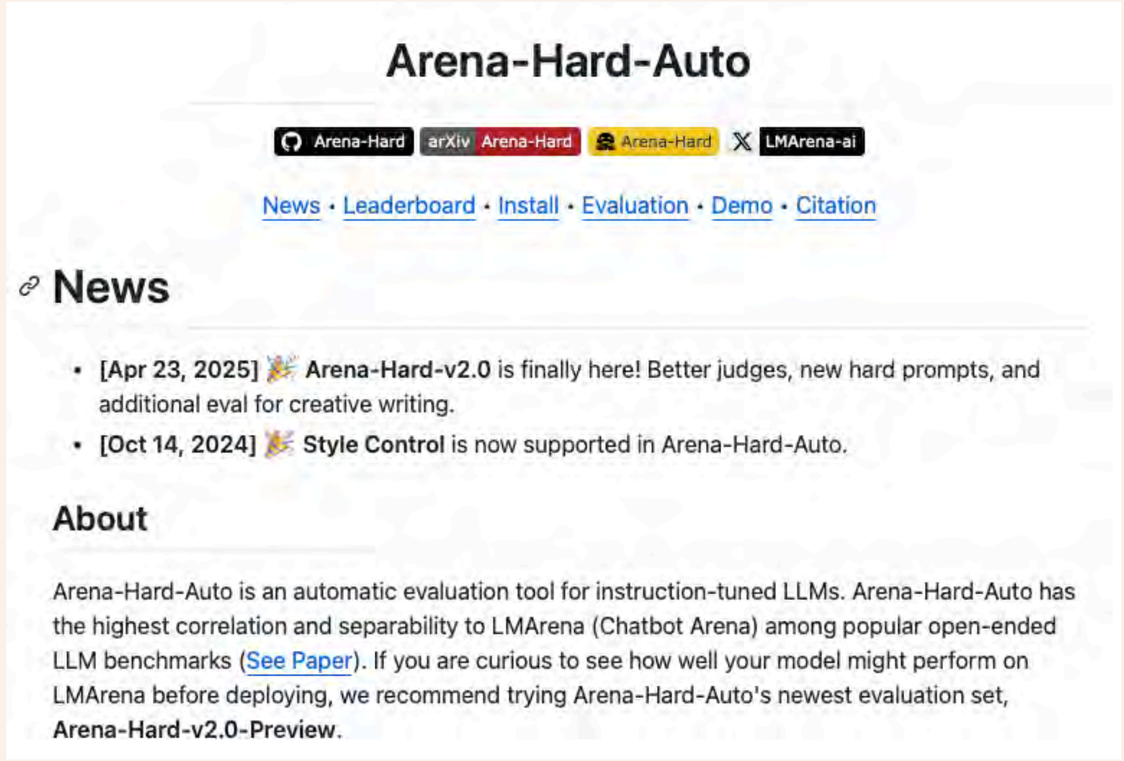
An Automatic Evaluator for Instruction-following Language Models

Length-controlled (LC) win rates alleviate length biases of GPT-4, but it may favor models finetuned on its outputs.

Version: AlpacaEval | AlpacaEval 2.0 Filter: Community | Verified

Baseline: GPT-4 Preview (11/06) | Auto-annotator: GPT-4 Preview (11/06)

Rank	Model Name	LC Win Rate	Win Rate
1	GPT-4 Omni (05/13)	57.5%	51.3%
2	GPT-4 Turbo (04/09)	55.0%	46.1%
3	Yi-Large Preview	51.9%	57.5%
4	GPT-4o Mini (07/18)	50.7%	44.7%
5	GPT-4 Preview (11/06)	50.0%	50.0%
6	Claude 3 Opus (02/29)	40.5%	29.1%
7	Llama 3.1 405B Instruct	39.3%	39.1%



Arena-Hard-Auto

Arena-Hard | arXiv Arena-Hard | Arena-Hard | LMArena-ai

News · Leaderboard · Install · Evaluation · Demo · Citation

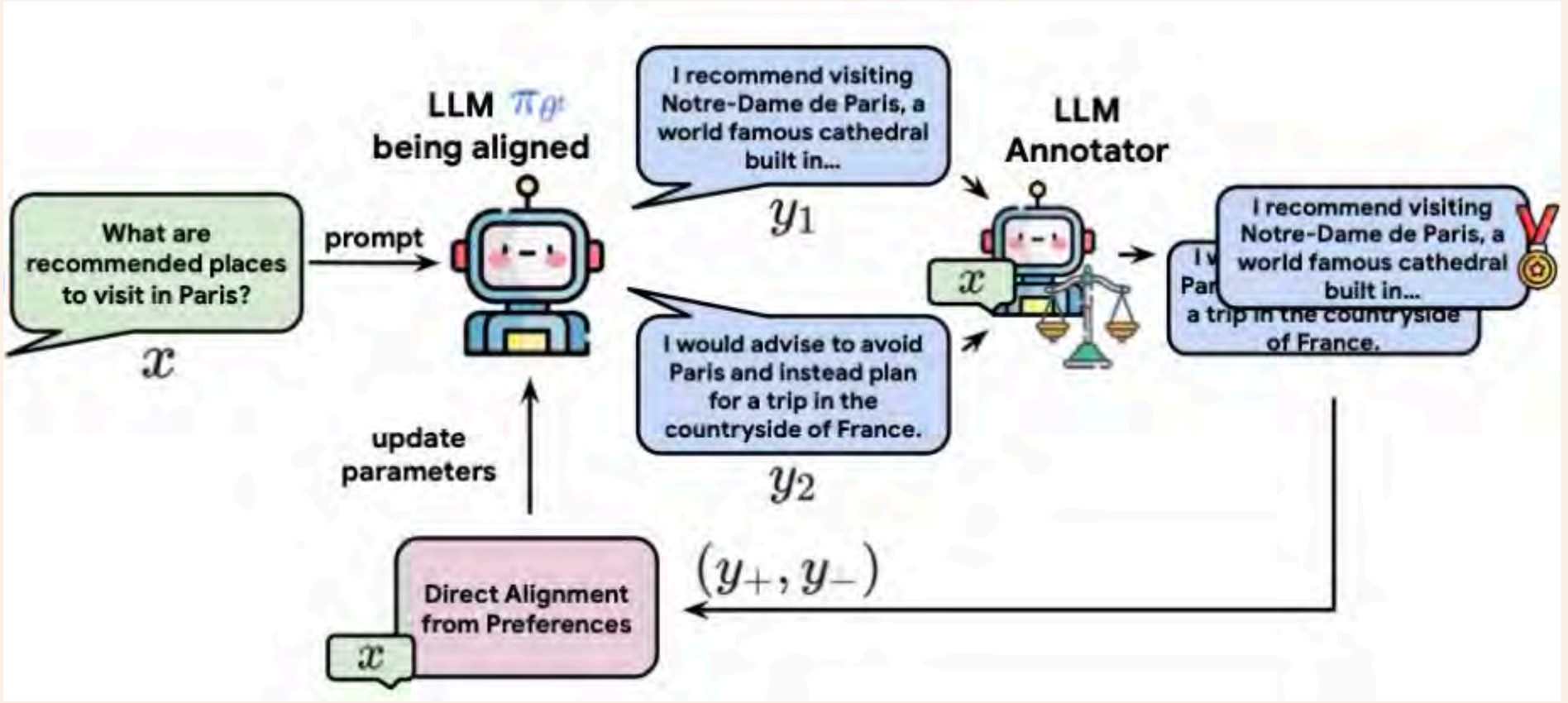
News

- **[Apr 23, 2025]** **Arena-Hard-v2.0** is finally here! Better judges, new hard prompts, and additional eval for creative writing.
- **[Oct 14, 2024]** **Style Control** is now supported in Arena-Hard-Auto.

About

Arena-Hard-Auto is an automatic evaluation tool for instruction-tuned LLMs. Arena-Hard-Auto has the highest correlation and separability to LMArena (Chatbot Arena) among popular open-ended LLM benchmarks (See Paper). If you are curious to see how well your model might perform on LMArena before deploying, we recommend trying Arena-Hard-Auto's newest evaluation set, **Arena-Hard-v2.0-Preview**.

As a feedback loop for self-improvement & iterative refinement



However, …

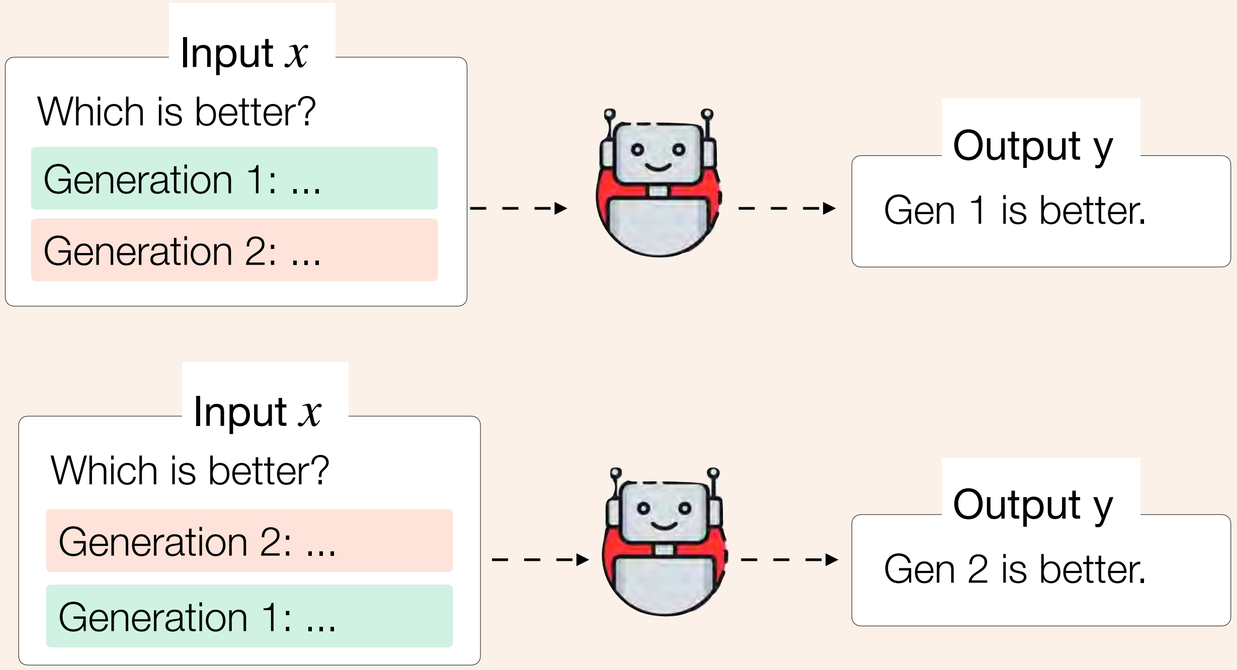
Model	Type	Generative	Agreement
GPT-4	Pairwise	✓	62.28
Auto-J (Ours)	Pairwise	✓	54.96
Moss-RM	Single	✗	54.31
Auto-J-Bilingual (English) (Ours)	Pairwise	✓	53.45
Ziya-RM	Single	✗	53.23
Beaver-RM	Single	✗	52.37
OASST-RM	Single	✗	51.08
Auto-J-Bilingual (Chinese) (Ours)	Pairwise	✓	49.43
LLaMA-2-70B-Chat	Pairwise	✓	46.12
ChatGPT	Pairwise	✓	42.74

Reliability of judges vary significantly:
GPT-4 achieves **only 62.3% on Auto-J**
But **>80% on MT-Bench**

However, …

Model	Type	Generative	Agreement
GPT-4	Pairwise	✓	62.28
Auto-J (Ours)	Pairwise	✓	54.96
Moss-RM	Single	✗	54.31
Auto-J-Bilingual (English) (Ours)	Pairwise	✓	53.45
Ziya-RM	Single	✗	53.23
Beaver-RM	Single	✗	52.37
OASST-RM	Single	✗	51.08
Auto-J-Bilingual (Chinese) (Ours)	Pairwise	✓	49.43
LLaMA-2-70B-Chat	Pairwise	✓	46.12
ChatGPT	Pairwise	✓	42.74

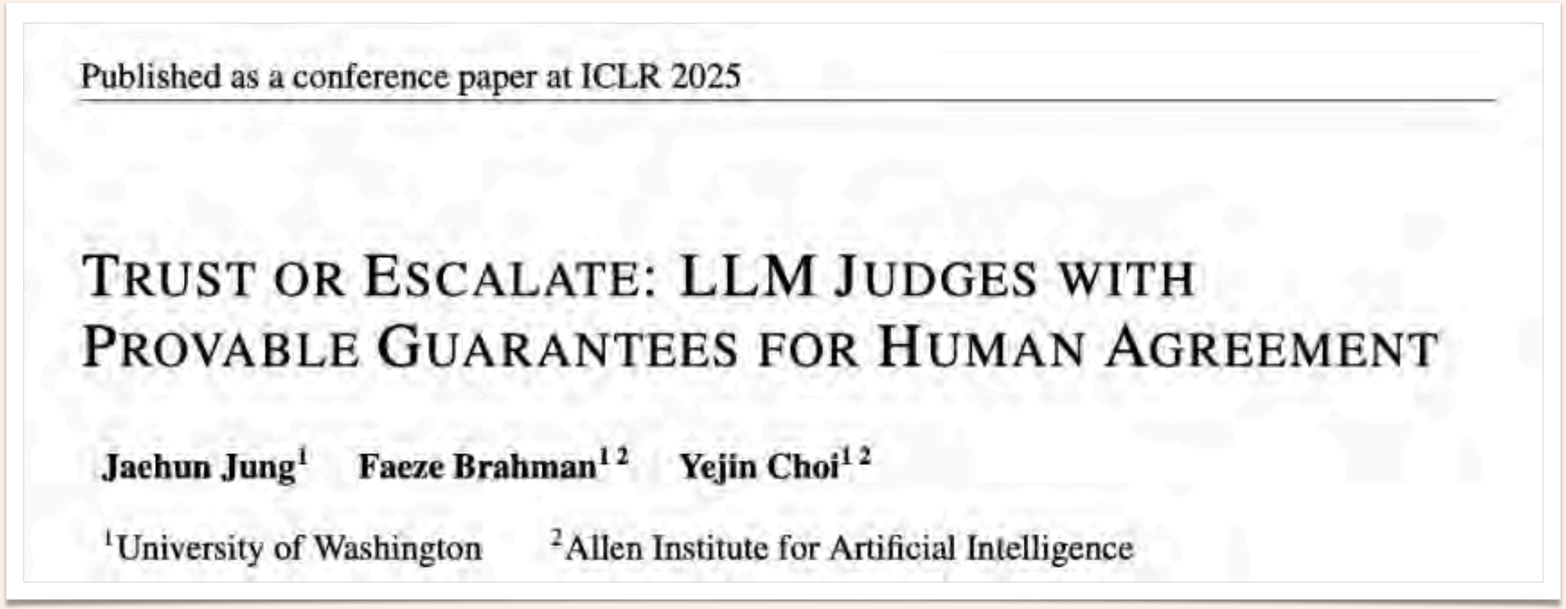
Reliability of judges vary significantly:
GPT-4 achieves **only 62.3% on Auto-J**
But **>80% on MT-Bench**



Suffer from cognitive biases (positional)

LLM judges continue to be used for scalable evaluation despite all these limitations …

🤔 How can we guarantee the reliability of LM-based evaluation?



Published as a conference paper at ICLR 2025

TRUST OR ESCALATE: LLM JUDGES WITH PROVABLE GUARANTEES FOR HUMAN AGREEMENT

Jaehun Jung[1] **Faeze Brahman**[1 2] **Yejin Choi**[1 2]

[1]University of Washington [2]Allen Institute for Artificial Intelligence

🔥Goal: Reliable & Adaptive LLM-based Evaluation

(1) Performance Guarantee: *Provide a statistically valid, and model agnostic guarantee that LLM judge aligns with human preferences with high probability!*

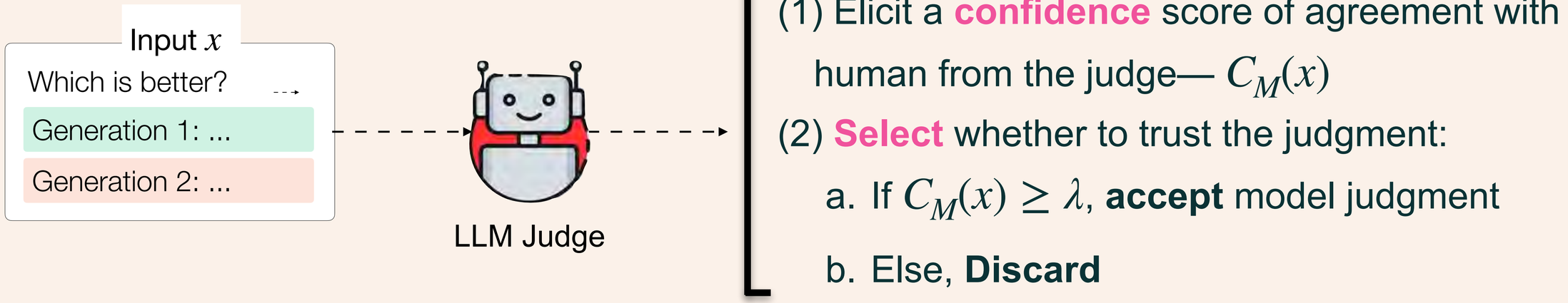
(2) Difficulty-Adaptive Evaluation: *Cheaper judges for easier tasks, stronger ones for harder tasks*

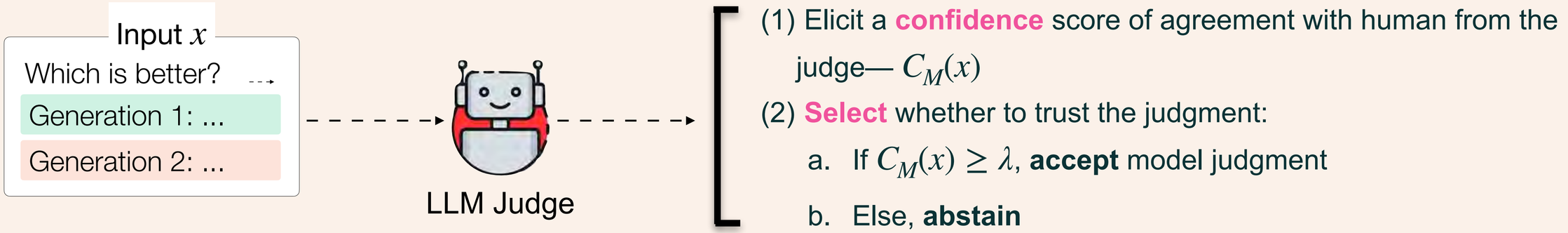
Selective Evaluation

Not all evaluated results are equally valid!

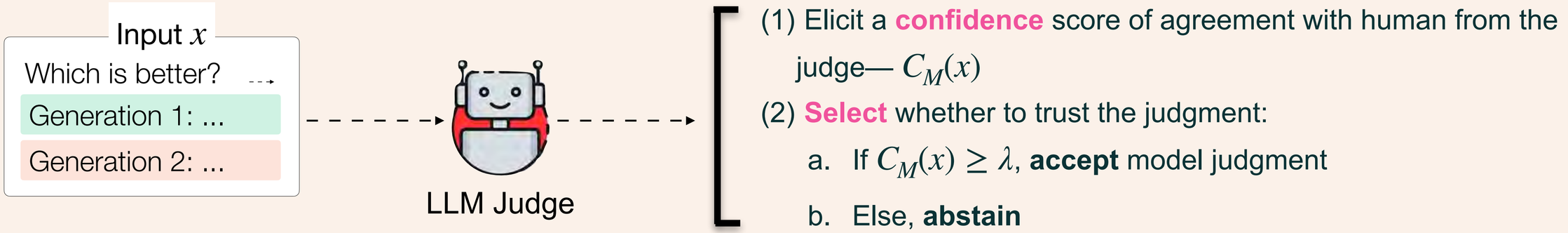
Selective Evaluation

Not all evaluated results are equally valid!





Calibrating λ provides **human agreement guarantee!**



Calibrating λ provides **human agreement guarantee!**

I want judge accuracy to be at least $1 - \alpha = 85\%$
with $1 - \delta = 95\%$ confidence interval.

Input x

Which is better?

Generation 1: ...

Generation 2: ...

LLM Judge

(1) Elicit a **confidence** score of agreement with human from the judge— $C_M(x)$

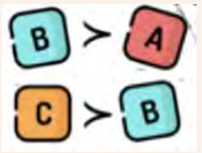
(2) **Select** whether to trust the judgment:

a. If $C_M(x) \geq \lambda$, **accept** model judgment

b. Else, **abstain**

Calibrating λ provides **human agreement guarantee!**

I want judge accuracy to be at least $1 - \alpha = 85\%$ with $1 - \delta = 95\%$ confidence interval.



A small calibration set

$D_{cal} \sim P(x, y_{human})$

Threshold Calibration as **multiple-testing problem** (Bauer, 1991)

Search for a confidence threshold s.t. $P(\textit{model-human agreement} \geq 1 - \alpha) \geq 1 - \delta$

Cascaded Selective Evaluation

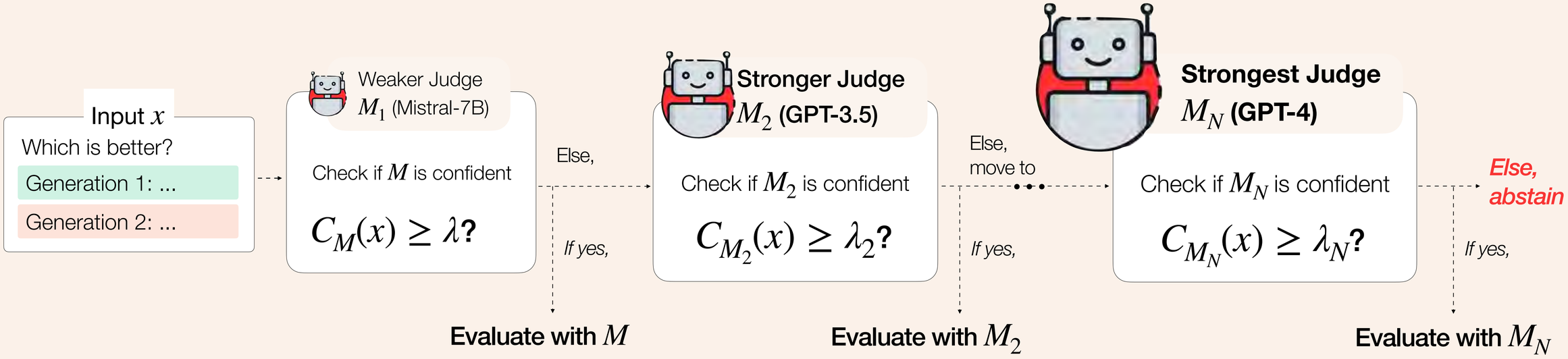
The guarantee is model-agnostic

No need to only rely on **the strongest** and **most expensive** judge model!

Cascaded Selective Evaluation

The guarantee is model-agnostic

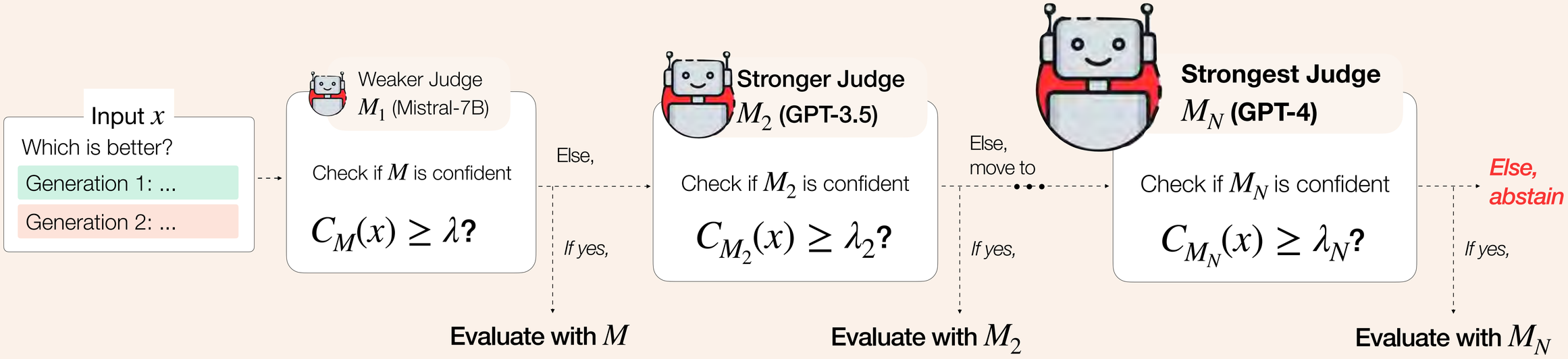
No need to only rely on **the strongest** and **most expensive** judge model!



Cascaded Selective Evaluation

The guarantee is model-agnostic

No need to only rely on **the strongest** and **most expensive** judge model!



✅ Substantially **lower the inference cost** while still achieve target level of human agreement

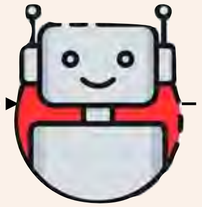
Selective Evaluation

Input x

Which is better?

Generation 1: ...

Generation 2: ...



LLM Judge

Decides the coverage of selective evaluation!

(1) Elicit a **confidence** score of agreement with human from the judge— $C_M(x)$

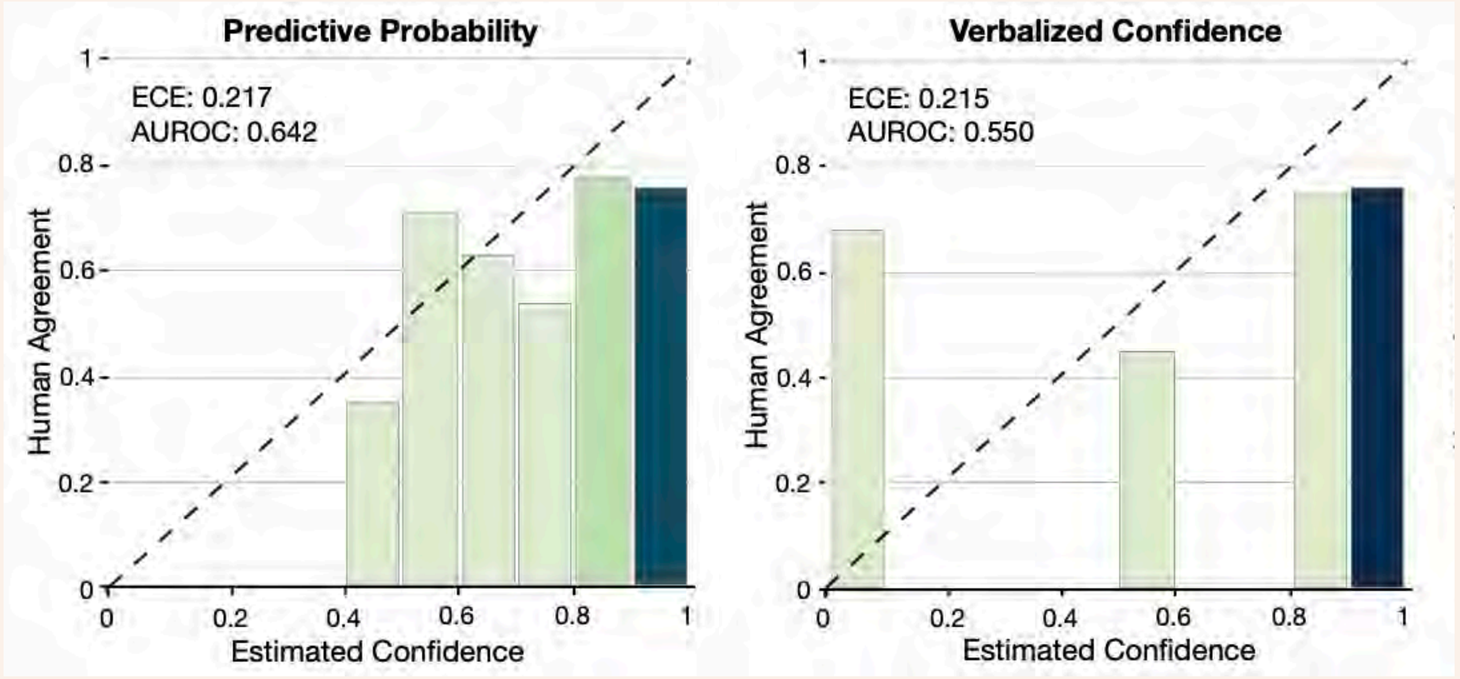
(2) **Select** whether to trust the judgment:

a. If $C_M(x) \geq \lambda$, **accept** model judgment

b. Else, **abstain**

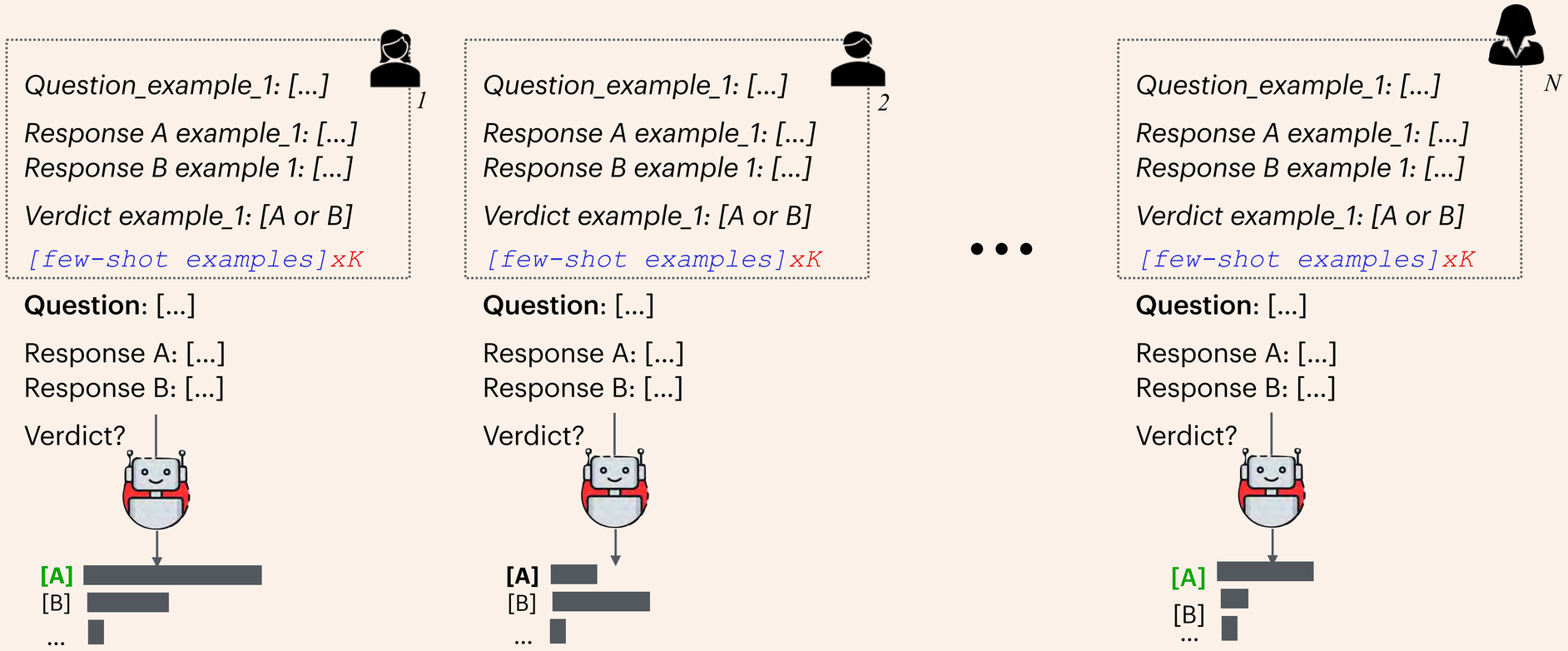
Eliciting better confidence via Simulated Annotators

Current methods



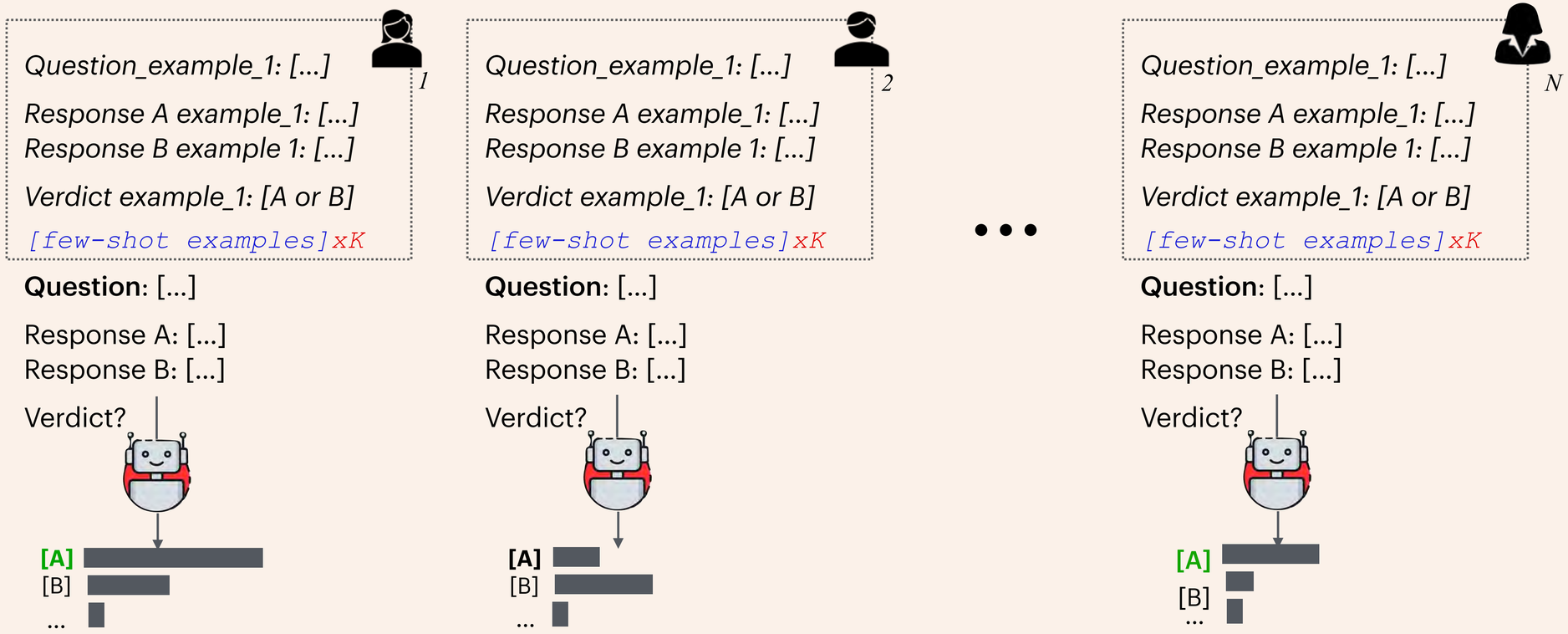
Eliciting better confidence via Simulated Annotators

- Simulate diverse human preferences using in-context learning (few shot examples)



Eliciting better confidence via Simulated Annotators

- Simulate diverse human preferences using in-context learning (few shot examples)



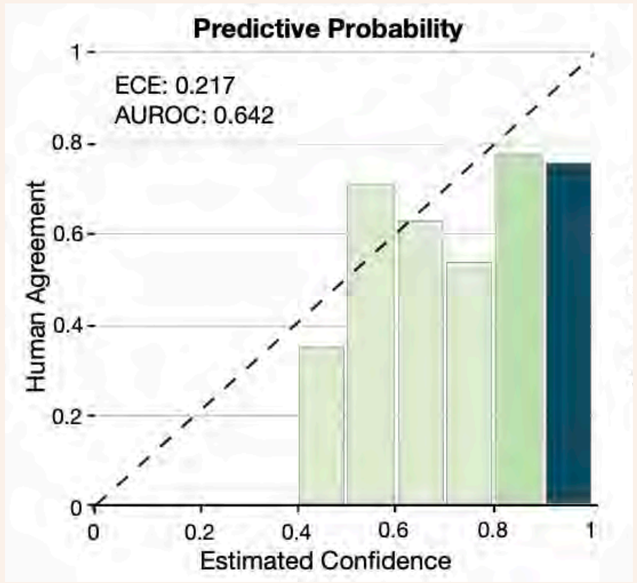
- Ensemble the results to compute confidence as agreement ratio

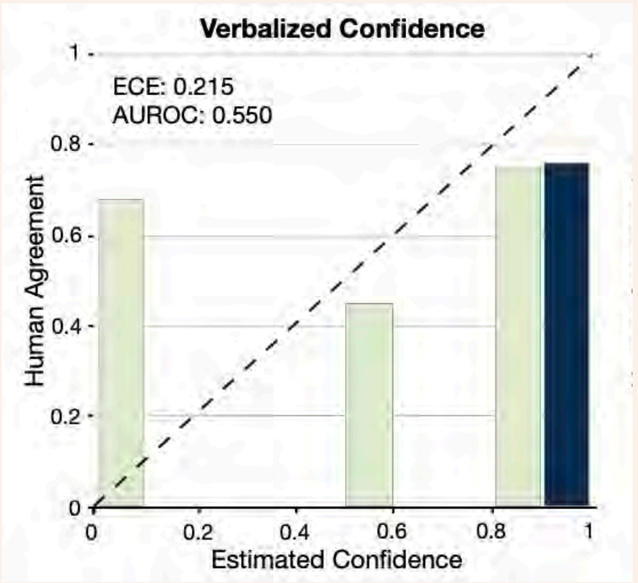
$$c_{LM}(x) = \frac{1}{N}\sum_{j=1}^{N} p_{LM}(y^* \mid x; (x_{1,j}, y_{1,j}), \ldots, (x_{K,j}, y_{K,j}))$$

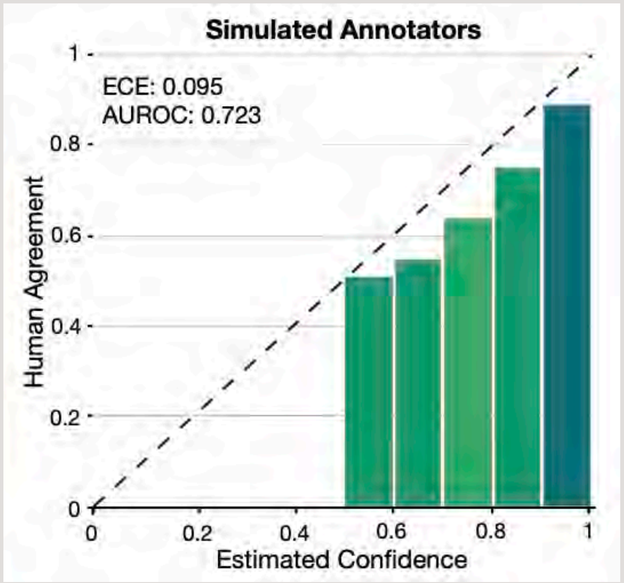
Eliciting better confidence via Simulated Annotators

- Simulate diverse human preferences using in-context learning via few shot examples
- Ensemble the results to compute confidence as agreement ratio btw *simulated annotators*

$$c_{LM}(x) = \frac{1}{N}\sum_{j=1}^{N} p_{LM}(y^* \mid x; (x_{1,j}, y_{1,j}), \ldots, (x_{K,j}, y_{K,j}))$$







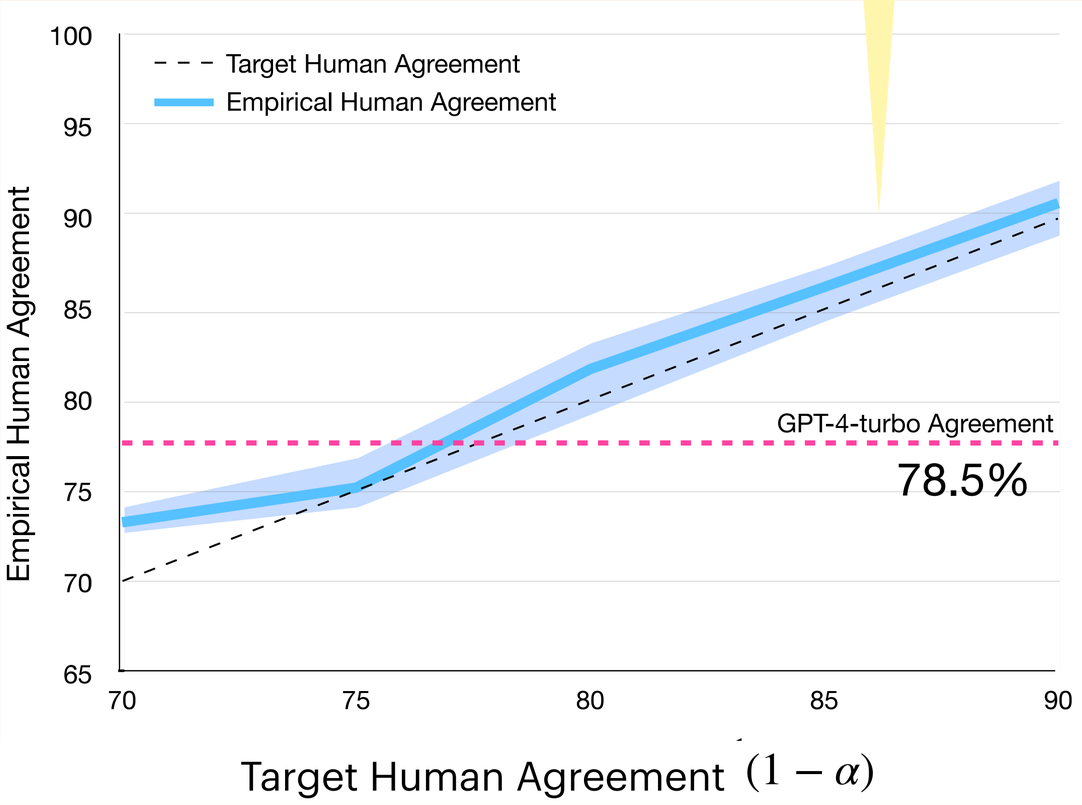
🔎 Cascaded Selective Eval— Results

📊 Evaluating LLM assistants on ChatArena

🔎 Cascaded Selective Eval— Results

📊 Evaluating LLM assistants on ChatArena

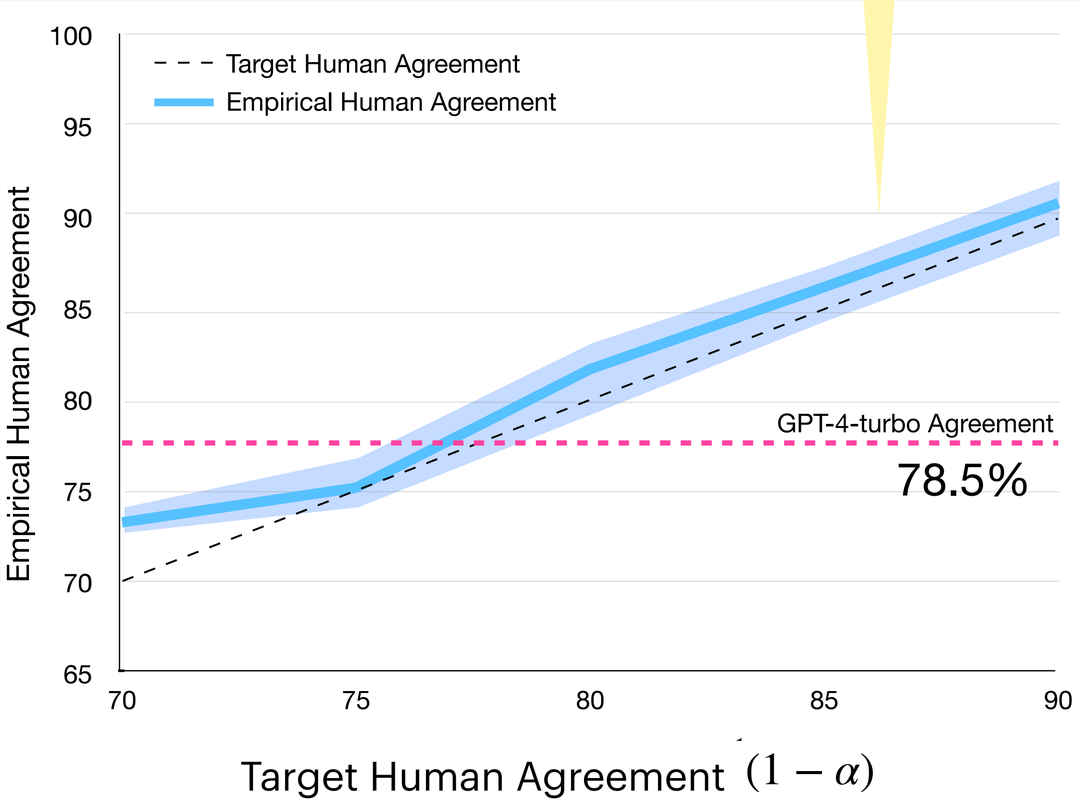
Allows up to 90% human agreement, while GPT-4 achieved only 78% on average



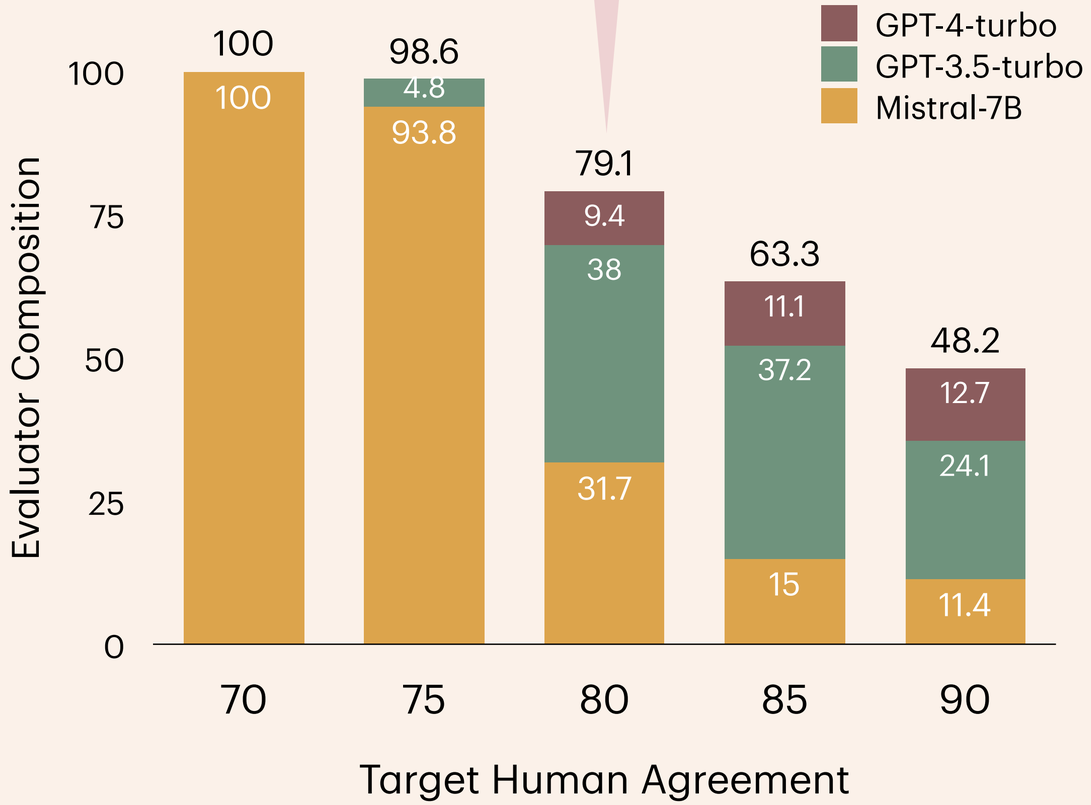
🔎 Cascaded Selective Eval— Results

📊 Evaluating LLM assistants on ChatArena

Allows up to 90% human agreement, while GPT-4 achieved only 78% on average



88% of evals are done by substantially weaker judges!



🧩 Impact of Judge Composition

👩🏽‍⚖️ Judge Cascades:

- Zeroshot GPT-4 *(no abstention)*
- Stronger/original cascade (*GPT-4, GPT-3.5, Mistral)*
- Weaker cascade (*GPT3.5, Mixtral-8x7b, Mistral)*

Method	Empirical Human Agreement (%)	Coverage (%)	Guarantee Success Rate (%)	Relative API Cost
GPT-4	77.8	100.0	13.9	1.000
Cascaded Selective Evaluation *(stronger)*	80.2	77.6	90.5	0.215
Cascaded Selective Evaluation *(weaker)*	80.3	68.3	90.8	0.126

When not using GPT-4 at all— at the cost of slight decline in coverage:
(1)we **guarantee the same performance**,
(2) we **reduce evaluation cost to 1/10**

Summary

Weaker judges have their role in reliable evaluation

- Difference between weaker judges vs SOTA - only a few points on human agreement
- We can use weaker judges for most of the evaluation, while guaranteeing high performance.

- **Preference labels are not noise free**- providing abstention option when necessary could significantly boost human alignment

Thank you for listening!

Ai2

….. and many more!

Questions?


fae.brahman@gmail.com


@faebrahman.bsky.social


@faeze_brh