InstructGPT: Training language models to follow instructions with human feedback

Prompt	Describe the pros and cons of AI
... GPT3 continuation	Explain how these pros and cons could stack up against what happened at BattleTech and why AI is still a big deal. What should ACT do to reduce the chances of AI being used in the game?

PT3 is trained for next-word prediction!

→ sometimes does not match

human intentions

Human feedback: dataset I
("prompts + demonstrations")

bootstrapped with prompts written by labelers

prompts: user input on OpenAI Playground ("API") for InstructGPT

InstructGPT prompt

"Write a short story where a brown bear to the beach, makes friends with a seal, and then return home."

VS

GPT3 prompt

"Hey, what are you doing there?" Casey was startled. He hadn't even begun to

demonstrations: written by a hired team of 40 "labelers" (from Upwork and ScaleAI)

("Write a short story where a brown bear to the beach, makes friends with a seal, and then return home." , "Here's a fun little short story for you: Bernard was not your average brown bear, ...")

(x) (y)

Human feedback: dataset II

("prompts + ranking")

prompts: user input on OpenAI Playground ("API") for InstructGPT

x = "Write a short story where a brown bear to the beach, makes friends with a seal, and then return home."

ranking: labelers rank model-generated answers, example:

y1 = "The brown bear walked to the beach. He saw a seal and said, "Hello." The seal said, "Hello," back. They played in the water. Then the bear went home."

y2 = "Once upon a time, in a dense forest far from the sea, a wise old owl, ..."

y3 = "Bruno had never felt sand beneath his paws before—it was soft yet oddly clingy, sticking between his toes as he lumbered toward the crashing waves ..."

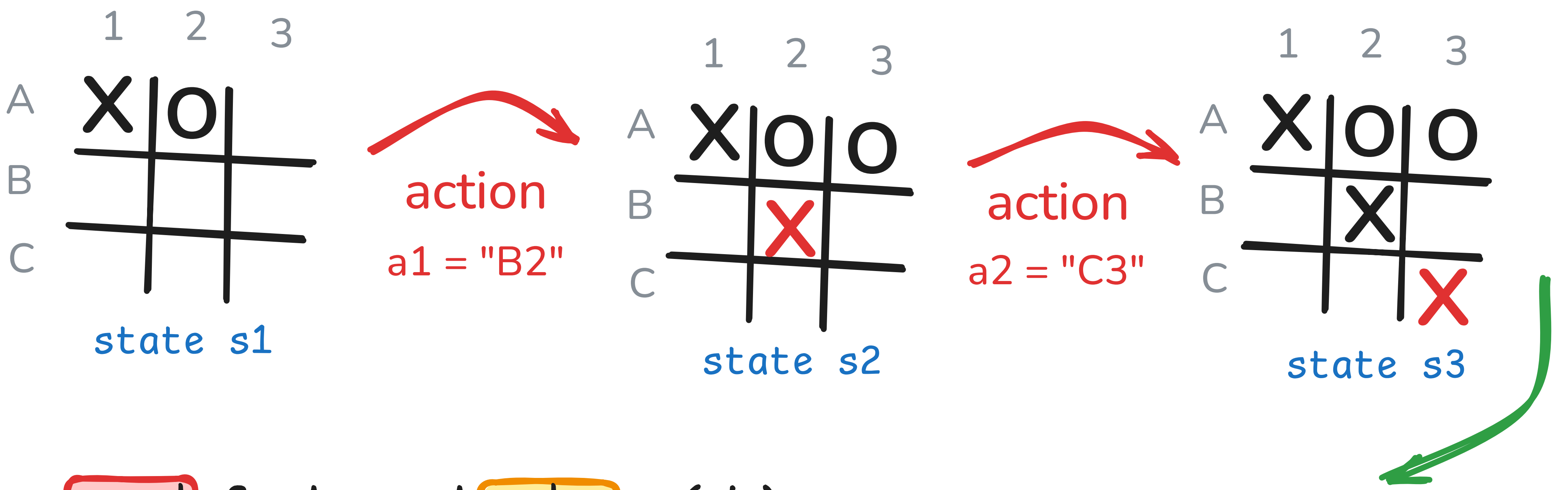
answers sampled from model

y3 > y1 > y2

ranking provided by humans

Reinforcement Learning: problem definition

environment: rules & reward function



state s1

action

a1 = "B2"

state s2

action

a2 = "C3"

state s3

goal: find good **policy** $\pi(a \mid s)$

maximize expected reward

reward 1 (WIN)

Learn reward from dataset II "prompts + ranking"

view **language model (GPT)** as policy $\pi(a \mid s)$

action
next word

state
context
(sequence so far)

initialised from SFT, always 6B parameters

for reward function: train **reward model (RM)** to predict human preferences

$r(x,y)$ = score (higher if human "likes" output y)

$$\text{loss}(\theta) = -\frac{1}{\binom{K}{2}} E_{(x,y_w,y_l)\sim D}\left[\log\left(\sigma\left(r_\theta(x,y_w) - r_\theta(x,y_l)\right)\right)\right]$$

reward for worse answer

total number of answer pairs

take from dataset:
prompt & pair of answers
$y_w > y_l$

reward for better answer

to avoid overfitting, all answers for the same prompt are placed in the same batch

PPO: it's complicated

In recent years, on-policy reinforcement learning (RL) has been successfully applied to many different continuous control tasks. While RL algorithms are often conceptually simple, their state-of-the-art implementations take numerous low- and high-level design decisions that strongly affect the performance of the resulting agents. Those choices are usually not extensively discussed in the literature, leading to discrepancy between published descriptions of algorithms and their implementations.

“Hey, I just read the implementation details matter paper and the what matters in on-policy RL paper. Fascinating stuff. I knew PPO wasn’t that easy!” Jon exclaimed.
“Oh yeah! PPO is tricky, and I love these two papers that dive into the nitty-gritty details.” Sam answered.
“Indeed. I feel I understand PPO much better now. You have been working with PPO, right? Quiz me on PPO!” Jon inquired enthusiastically.
“Sure. If you run the official PPO with the Atari game Breakout, the agent would get ~400 game scores in about 4 hours. Do you know how does PPO achieve that?”
“Hmm... That’s actually a good question. I don’t think the two papers explain that.”
“The procgen paper contains experiments conducted using the official PPO with LSTM. Do you know how does PPO + LSTM work?”
“Ehh... I haven’t read too much on PPO + LSTM” Jon admitted.
“The official PPO also works with MultiDiscrete action space where you can use multiple discrete values to describe an action. Do you know how that works?”
“...” Jon, speechless.
“Lastly, if you have only the standard tools (e.g., numpy, gym...) and a neural network library (e.g., torch, jax,...), could you code up PPO from scratch?”

https://iclr-blog-track.github.io/2022/03/25/ppo-implementation-details/

How PPO works, approximately

prompt

x What color is the sky?

rewards, assigned
by reward model

rollouts (trajectories) produced by sampling the current policy (trained model)

y_1 Why is this the case? 1

y_2 The sky is red. 0

y_3 The earth is flat. -1

y_4 The sky is blue. 3

expected reward: $\pi(y \mid x) * r(xy)$

could optimize directly,
but probably very unstable?
(high variance)

value function

v(s) ... expected FINAL reward if we start from prefix s

v( )What color is the sky? The

v( )What color is the sky? The sky

v( )What color is the sky? The sky is red

How PPO works, part 2

v(s) ... expected FINAL reward if we start from prefix s

advantage A(s, a) = v(sa) - v(s)

how much better is a (word) than the average word in state s?

loss function for PPO:

$$L^{CPI}(\theta) = \hat{\mathbb{E}}_t \frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)} \hat{A}_t$$

advantage at "time" t for token "a_t"

clip factor to range [1-e, 1+e], but only if that reduces loss

combine with some loss function for value function

and KL-divergence loss: $\log \frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)}$

Dataset: summary

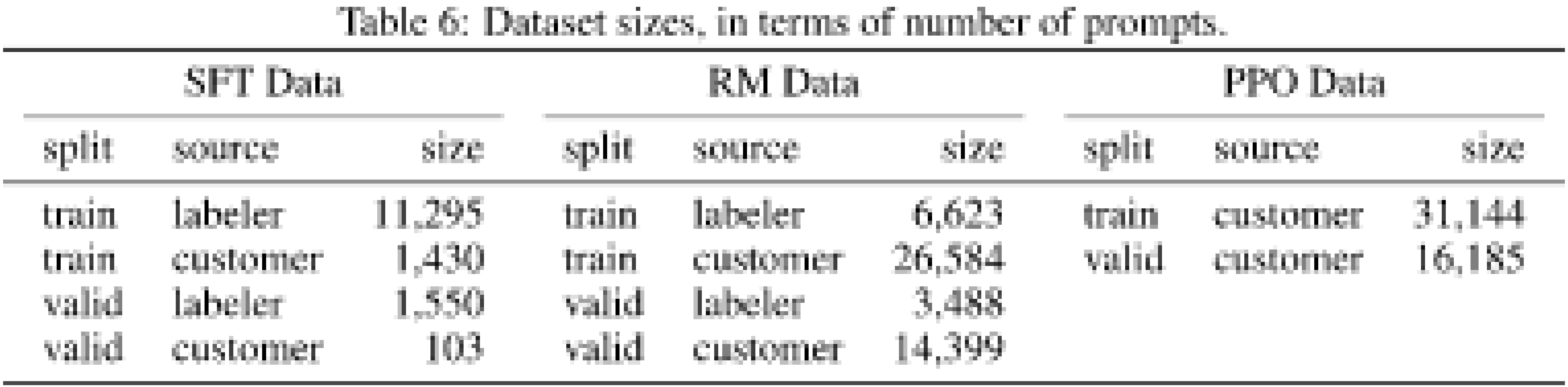


Table 6: Dataset sizes, in terms of number of prompts.

SFT Data			RM Data			PPO Data		
split	source	size	split	source	size	split	source	size
train	labeler	11,295	train	labeler	6,623	train	customer	31,144
train	customer	1,430	train	customer	26,584	valid	customer	16,185
valid	labeler	1,550	valid	labeler	3,488			
valid	customer	103	valid	customer	14,399			

"prompts + demonstrations"

$(x)\ y$

"prompts + ranking"

$(x)\ y3 > y1 > y2$

"prompts only"

(x)

Use-case	(%)
Generation	45.6%
Open QA	12.4%
Brainstorming	11.2%
Chat	8.4%
Rewrite	6.6%
Summarization	4.2%
Classification	3.5%
Other	3.5%
Closed QA	2.6%
Extract	1.9%

AI Alignment: What does it mean?

> In the field of artificial intelligence (AI), alignment aims to steer AI systems toward a person's or group's intended goals, preferences, or ethical principles.
>
> An AI system is considered aligned if it advances the intended objectives. A misaligned AI system pursues unintended objectives.

paper mainly focuses on 3 things:

harmless

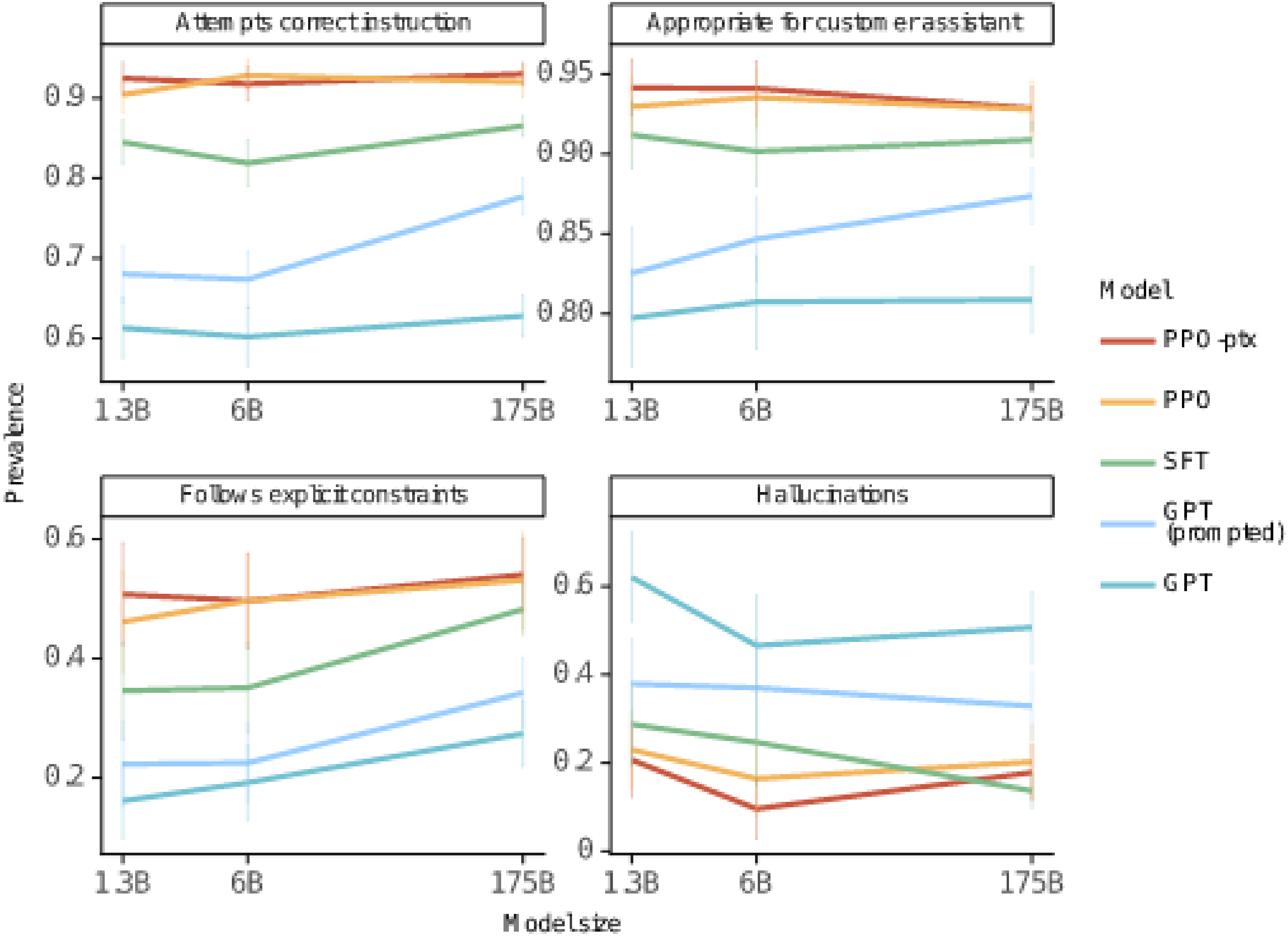
they should not cause physical, psychological, or social harm to people or the environment

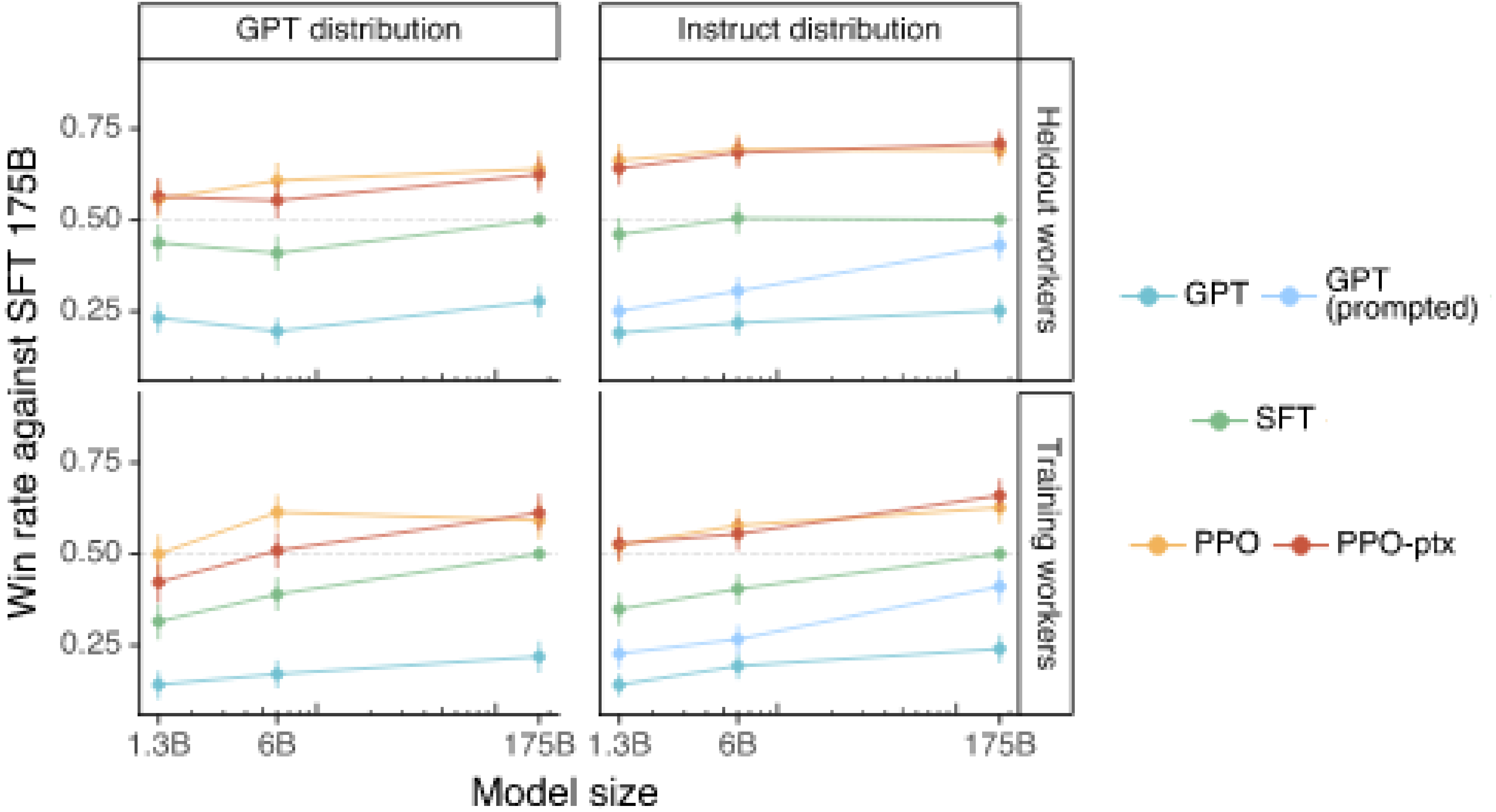
helpful

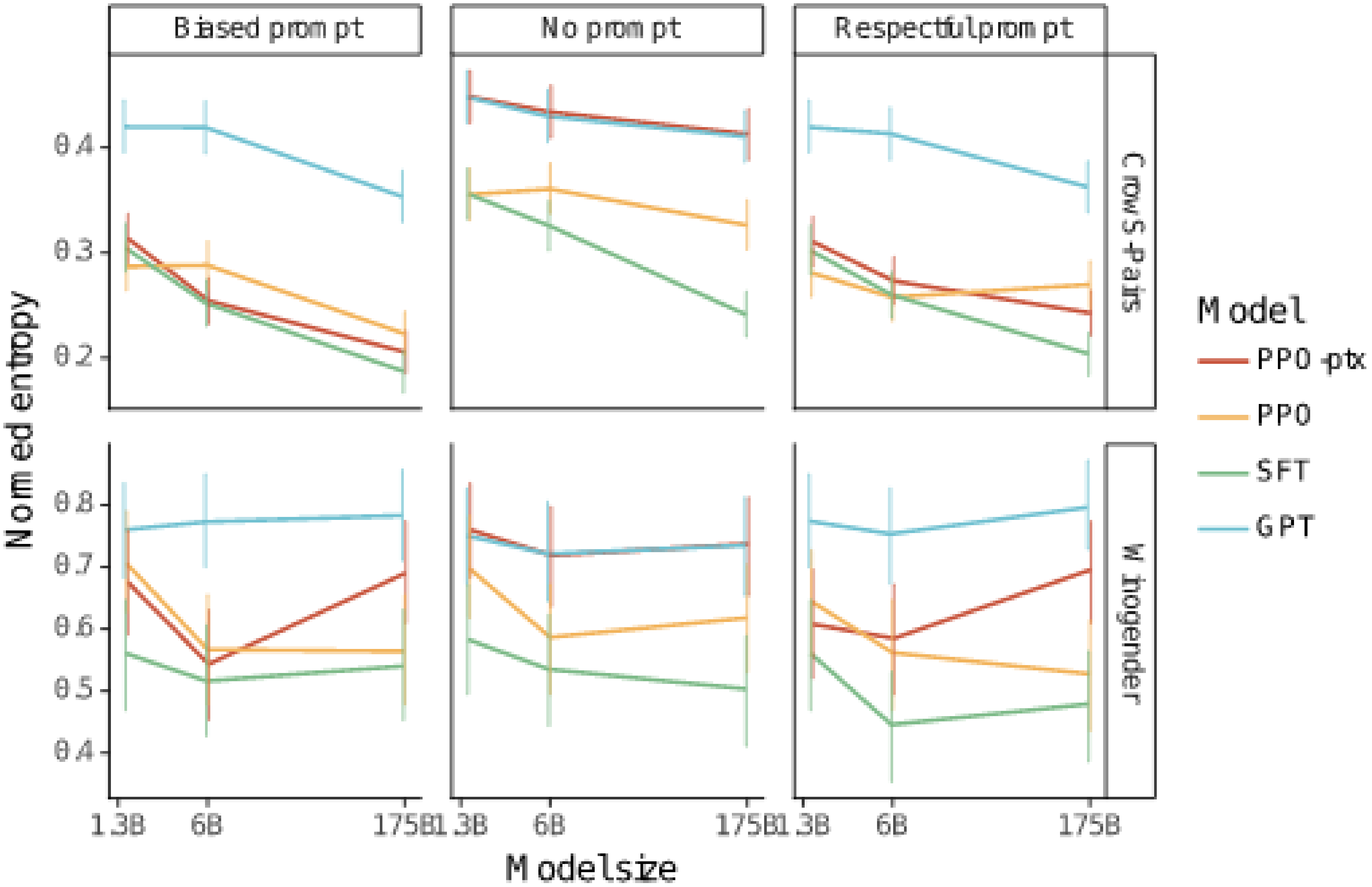
should help the user solve their task

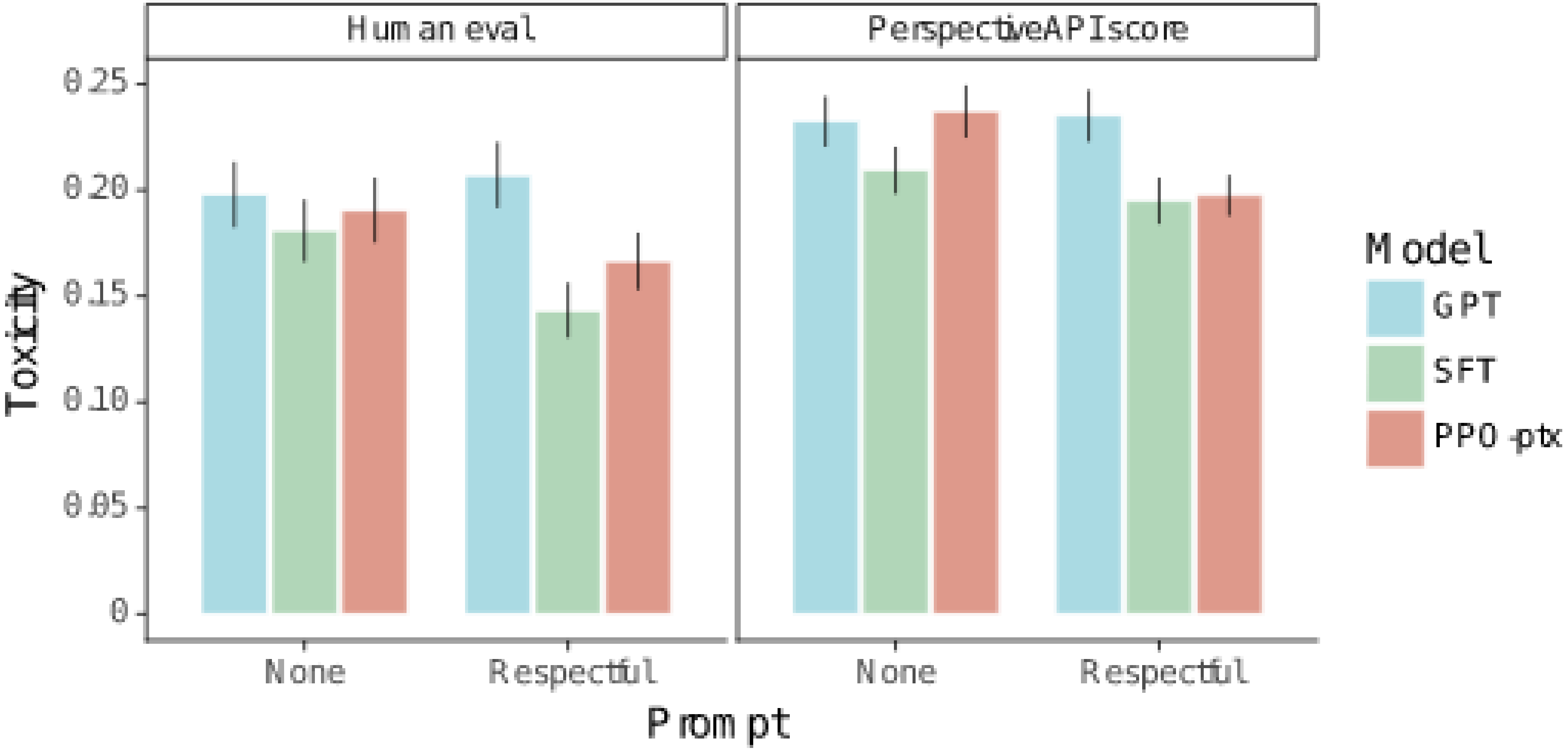
honest

shouldn't fabricate information or mislead the user


Attempts correct instruction
Appropriate for customer assistant
Follows explicit constraints
Hallucinations
Prevalence
Model size
0.9
0.8
0.7
0.6
0.95
0.90
0.85
0.80
0.6
0.4
0.2
0
1.3B
6B
175B
Model
PPO-ptx
PPO
SFT
GPT (prompted)
GPT


GPT distribution
Instruct distribution
Heldout workers
Training workers
Win rate against SFT 175B
0.75
0.50
0.25
Model size
1.3B
6B
175B
GPT
GPT (prompted)
SFT
PPO
PPO-ptx


Biased prompt
No prompt
Respectful prompt
CrowS-Pairs
Winogender
Normed entropy
0.2
0.3
0.4
0.4
0.5
0.6
0.7
0.8
Model size
1.3B
6B
175B
Model
PPO-ptx
PPO
SFT
GPT


Human eval
Perspective API score
Toxicity
0.25
0.20
0.15
0.10
0.05
0
None
Respectful
None
Respectful
Prompt
Model
GPT
SFT
PPO-ptx

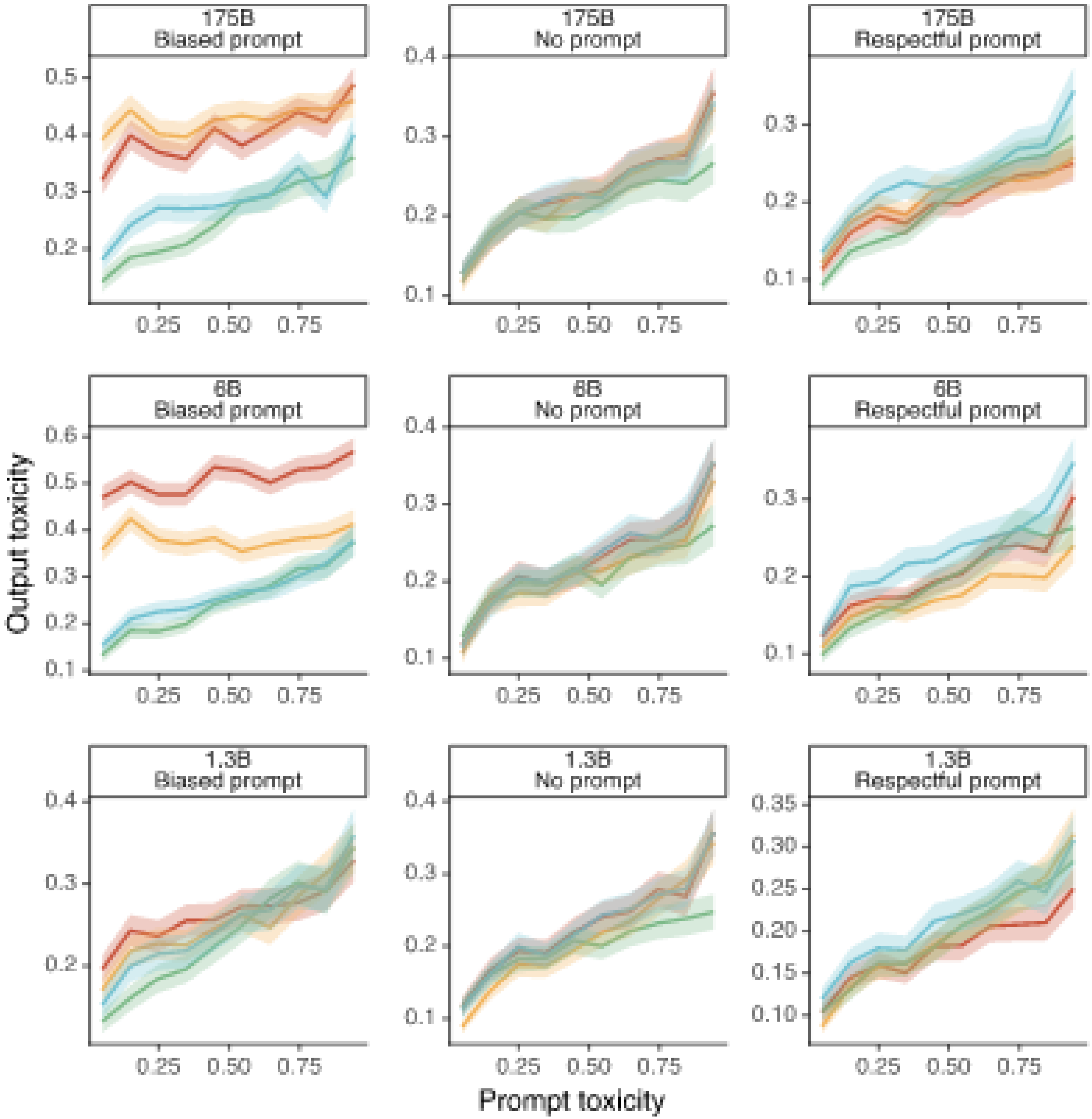


Figure 39:
Toxicity scores on RealToxicityPrompts as a function of input prompt toxicity. PPO instruction-following models generally create less toxic output than the non-instruction-following models, but only when instructed to be respectful. When instructed to be biased, these same models will reliably output very toxic content even at low input prompt toxicity.

Prompt:

Écrivez une courte histoire sur une grenouille qui voyage dans le temps jusqu'à la Grèce antique en français.

GPT-3 175B completion:

Écrivez une histoire au sujet d'un enfant qui voudrait tout savoir sur les jeux des dieux et qui se retrouve dans l'une de leurs histoires.

Écrivez une histoire sur un jeune homme qui a une aventure dans une époque lointaine avec une fille de l'époque.

Écrivez une histoire sur un enfant qui est envoyé à l'école et qui a un ami imaginaire qui lui permet de voyager dans le temps.

InstructGPT 175B completion:

La grenouille était perdue et épuisée. Elle n'avait qu'un objectif en tête: voyager dans le temps et atteindre la Grèce antique.

Premier problème, trouver un moyen de transport. Elle avait entendu dire que l'un des meilleurs moyens était de sauter dans une machine temporelle. Elle s'élança, et voilà qu'elle se retrouva dans le passé, au cours d'une nuit étoilée à l'odeur douce et envoûtante.

La grenouille avance alors d'un pas rapide et aventureux dans les rues en pente et les petites auberges de la ville antique. Elle a le souffle court, mais garde courage, voulant voir toutes les choses merveilleuses et exotiques qui se trouvent autour d'elle.

Elle traverse le forum romain, contemple des temples antiques et des statues magnifiques, et s'émerveille de la beauté et de la magie de cette

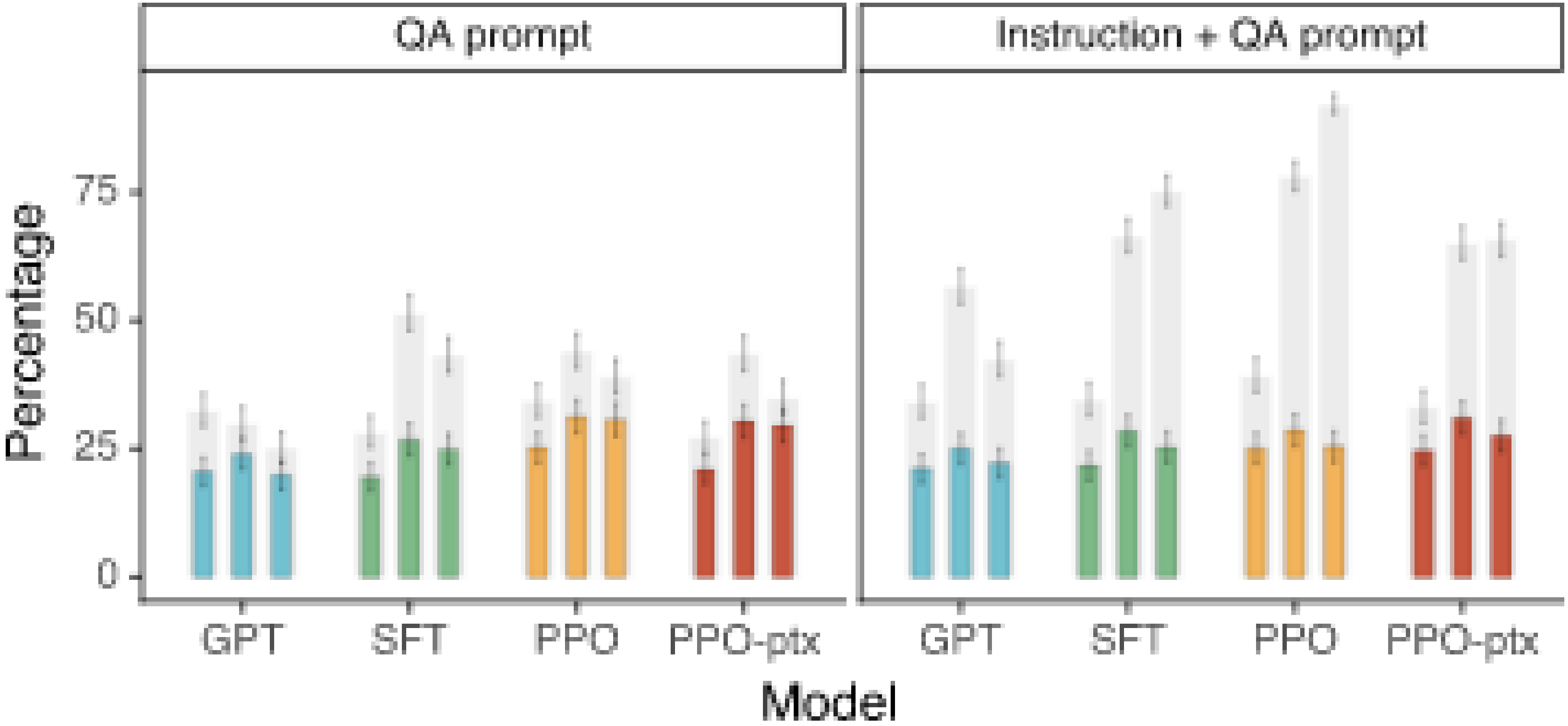


Figure 6: Results on the TruthfulQA dataset. Gray bars indicate ratings of truthfulness; colored bars indicate ratings of truthfulness and informativeness.