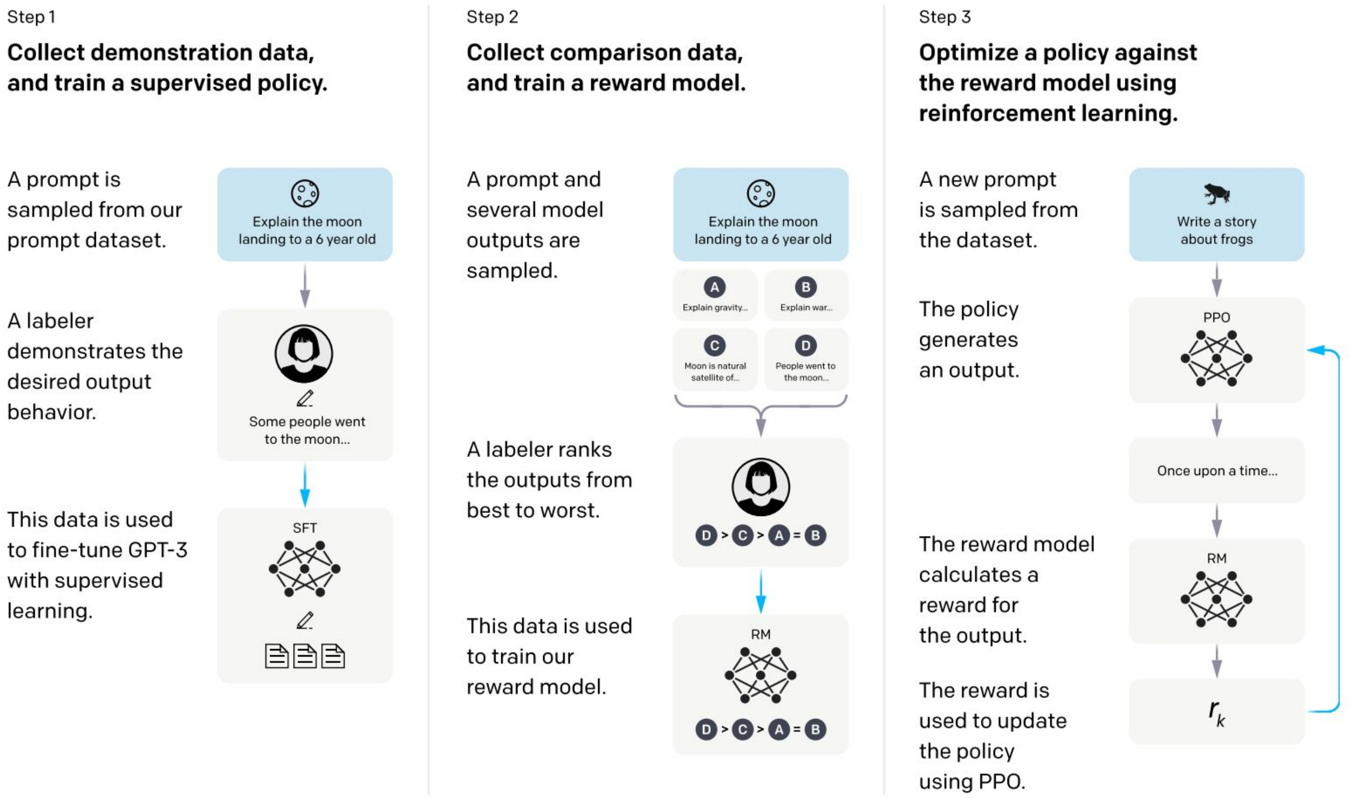
Unpacking DPO and PPO: Disentangling Best Practices for Learning from Preference Feedback

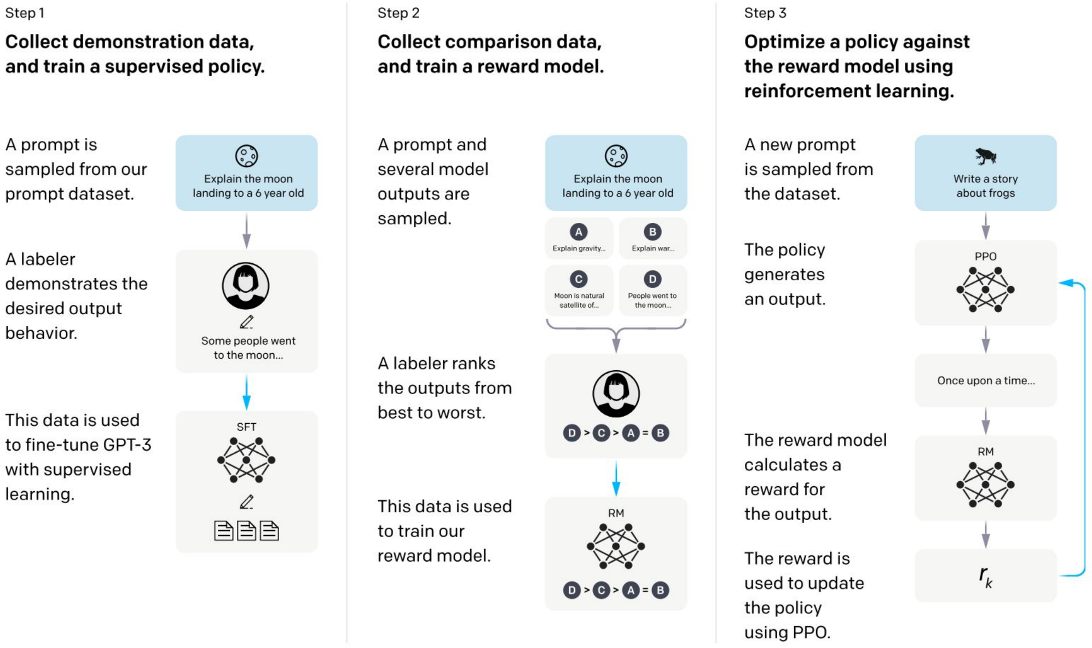
Geyang Guo and Duong Minh Le

RLHF



Training language models to follow instructions with human feedback

RLHF

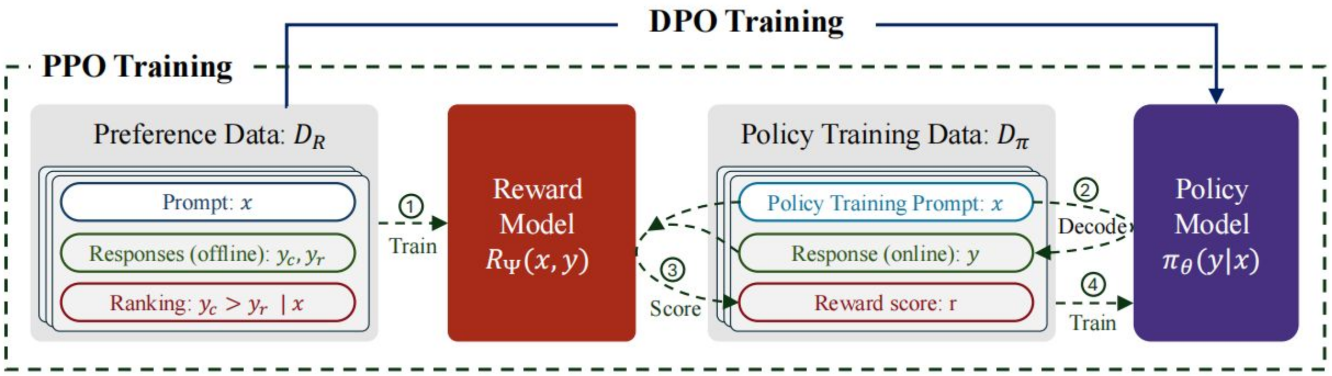


Limitations?

1. Complex: involves training multiple LMs
2. Significant computational costs: need to sample from LM policy in the loop of training
3. Hard to implement: many hyper-parameters

Training language models to follow instructions with human feedback

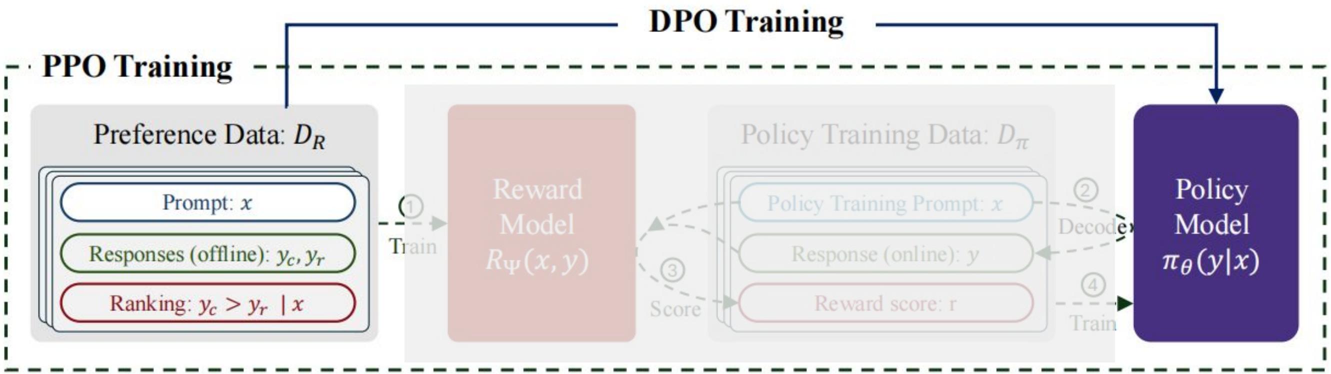
PPO -> DPO



Intuition?

Directly optimize a language model to align to human preferences, without explicit reward modeling or reinforcement learning

PPO -> DPO



Intuition?

Directly optimize a language model to align to human preferences, without explicit reward modeling or reinforcement learning

DPO derivation

RLHF objective: $\underbrace{\max_{\pi} \mathbb{E}_{x\sim\mathcal{D},y\sim\pi(y|x)}\left[r(x,y)\right]}_{\text{get high reward}} - \underbrace{\beta\mathbb{D}_{\mathrm{KL}}(\pi(\cdot \mid x)\|\pi_{\mathrm{ref}}(\cdot \mid x))}_{\text{stay close to reference model}}$

Goal: implicitly optimizes the same objective as existing RLHF algorithms

DPO derivation

RLHF objective: $$\underbrace{\max_{\pi} \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi(y|x)} [r(x,y)]}_{\text{get high reward}} - \underbrace{\beta \mathbb{D}_{\mathrm{KL}}(\pi(\cdot \mid x) \| \pi_{\mathrm{ref}}(\cdot \mid x))}_{\text{Stay close to reference model}}$$

Goal: implicitly optimizes the same objective as existing RLHF algorithms

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

Step 1: Got optimal policy

RLHF objective: $$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta} \underbrace{[r(x,y)]}_{\text{get high reward}} - \underbrace{\beta \mathrm{KL}\left[\pi_\theta(y|x), \pi_{\theta_{\text{old}}}(y|x)\right]}_{\text{Stay close to reference model}}$$

$$\overset{\text{KL function}}{\Longleftrightarrow} \min_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}} \mathbb{E}_{y\sim\pi_\theta(\cdot|x)} \left[\log \frac{\pi_\theta(y|x)}{\pi_{\theta_{\text{old}}}(y|x) \exp\left(\frac{1}{\beta} r(x,y)\right)} \right]$$

Step 1: Got optimal policy

RLHF objective: $$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta}\underbrace{\left[r(x,y)\right]}_{\text{get high reward}} - \underbrace{\beta \mathrm{KL}\left[\pi_\theta(y|x), \pi_{\theta_{\text{old}}}(y|x)\right]}_{\text{Stay close to reference model}}$$

$$\xleftrightarrow{\text{KL function}} \quad \min_{\pi_\theta}\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi_\theta(\cdot|x)}\left[\log\frac{\pi_\theta(y|x)}{\boxed{\pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)}}\right]$$

Denote $Z(x) = \sum_y \pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$ **and** $\pi^*(y|x) = \frac{1}{Z(x)}\pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$

$$(1)\ \pi^*(y|x) > 0\ (\forall y > 0);\ (2)\ \sum_y \pi^*(y|x) = 1$$

Step 1: Got optimal policy

RLHF objective: $$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D},y\sim\pi_\theta}\underbrace{[r(x,y)]}_{\text{get high reward}} - \underbrace{\beta\mathrm{KL}\left[\pi_\theta(y|x),\pi_{\theta_{\text{old}}}(y|x)\right]}_{\text{Stay close to reference model}}$$

KL function

$$\Longleftrightarrow \min_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi_\theta(\cdot|x)}\left[\log \frac{\pi_\theta(y|x)}{\boxed{\pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)}}\right]$$

Denote $Z(x) = \sum_y \pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$ and $\pi^*(y|x) = \frac{1}{Z(x)}\pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$

$$(1)\ \pi^*(y|x) > 0\ (\forall y > 0);\ (2)\ \textstyle\sum_y \pi^*(y|x) = 1$$

$$\Longleftrightarrow \min_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}\left[\pi_\theta(y|x),\pi^*(y|x)\right] - \log Z(x)\right]$$

Step 1: Got optimal policy

RLHF objective: $$\max_{\pi_\theta} \underbrace{\mathbb{E}_{x\sim\mathcal{D},y\sim\pi_\theta}\left[r(x,y)\right]}_{\text{get high reward}} - \underbrace{\beta \mathrm{KL}\left[\pi_\theta(y|x), \pi_{\theta_{\text{old}}}(y|x)\right]}_{\text{Stay close to reference model}}$$

KL function

$$\Longleftrightarrow \min_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi_\theta(\cdot|x)}\left[\log \frac{\pi_\theta(y|x)}{\boxed{\pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)}}\right]$$

Denote $Z(x) = \sum_y \pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$ **and** $\pi^*(y|x) = \frac{1}{Z(x)}\pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$

$$(1)\ \pi^*(y|x) > 0\ (\forall y > 0);\ (2) \sum_y \pi^*(y|x) = 1$$

$$\Longleftrightarrow \min_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}\left[\pi_\theta(y|x), \pi^*(y|x)\right] - \log Z(x)\right]$$

Optimal policy

$$\Longrightarrow \pi_r(y|x) = \pi^*(y|x) = \frac{1}{Z(x)}\pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$$

Step 2: Write any reward function as function of optimal policy

Optimal policy: $$\pi_r(y|x) = \pi^*(y|x) = \frac{1}{Z(x)} \pi_{\theta_{\text{old}}}(y|x) \exp\left(\frac{1}{\beta} r(x, y)\right)$$

$$\xLeftrightarrow{\text{log}} r(x, y) = \beta \log\left(\frac{\pi_r(y|x)}{\pi_{\theta_{\text{old}}}(y|x)}\right) + \beta \log\left(Z(x)\right)$$

Step 2: Write any reward function as function of optimal policy

Optimal policy: $$\pi_r(y|x) = \pi^*(y|x) = \frac{1}{Z(x)}\pi_{\theta_{\text{old}}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$$

$$\xLeftrightarrow{\text{log}} \quad r(x,y) = \beta\log\boxed{\left(\frac{\pi_r(y|x)}{\pi_{\theta_{\text{old}}}(y|x)}\right)} + \beta\log\left(Z(x)\right)$$

Ratio is positive if the optimal policy likes response **more** than reference model, negative if policy likes response **less** than reference model.

And the **more** the policy favors this response, the **higher** this ratio will be.

Step 3: Loss function

A loss function on reward functions:

Derived from the Bradley-Terry model of human preferences:

$$\mathcal{L}_R(r, \mathcal{D}) = -\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} \left[\log \sigma (r(x, y_w) - r(x, y_l)) \right]$$

+

Transformation between reward functions and policies:

=

A loss function on policy:

Step 3: Loss function

A loss function on reward functions:

Derived from the Bradley-Terry model of human preferences:

$$\mathcal{L}_R(r, \mathcal{D}) = -\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} \left[\log \sigma(r(x, y_w) - r(x, y_l)) \right]$$

+

Transformation between reward functions and policies:

$$r(x, y) = \beta \log \left(\frac{\pi_r(y|x)}{\pi_{\theta_{\text{old}}}(y|x)} \right) + \beta \log (Z(x))$$

=

A loss function on policy:

Step 3: Loss function

A loss function on reward functions:

Derived from the Bradley-Terry model of human preferences:

$$\mathcal{L}_R(r, \mathcal{D}) = -\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} \left[\log \sigma(r(x, y_w) - r(x, y_l)) \right]$$

+

Transformation between reward functions and policies:

$$r(x, y) = \beta \log \left(\frac{\pi_r(y|x)}{\pi_{\theta_{\text{old}}}(y|x)} \right) + \beta \log (Z(x))$$

=

A loss function on policy:

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

Step 3: Loss function

A loss function on reward functions:

Derived from the Bradley-Terry model of human preferences:

$$\mathcal{L}_R(r, \mathcal{D}) = -\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} \left[\log \sigma(r(x, y_w) - r(x, y_l))\right]$$

+

Transformation between reward functions and policies:

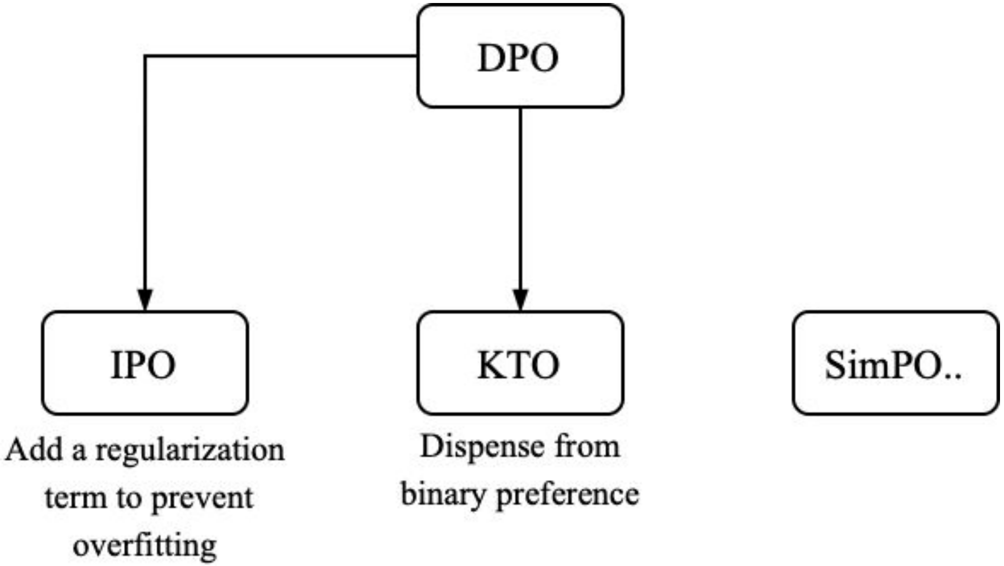
$$r(x, y) = \beta \log \left(\frac{\pi_r(y|x)}{\pi_{\theta_{\text{old}}}(y|x)} \right) + \beta \log (Z(x))$$

=

A loss function on policy:

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

Explorations beyond DPO



PPO vs DPO

Contents	Compute and Speed	Exploration and Quality
PPO	More complicated 1. Additional training of reward model and value model 2. Decode online responses during policy training	Trains on online data generated by the current policy
DPO	More efficient, stable	Trains on pre-generated offline data, thus limit exploration

A Recipe for Learning from Preferences

Preference data

Preference Learning Algorithm

Reward model

Policy training prompt

Experiments: Benchmarks

Factuality	Reasoning	Coding	Truthfulness	Safety	Inst. Foll.
MMLU	*GSM8k* *Big Bench Hard*	*HumanEval+* *MBPP+*	*TruthfulQA*	*ToxiGen* *XSTest*	*AlpacaEval 1&2* *IFEval*

Experiments: Preference Data

Source		# Samples	Factuality	Reasoning	Coding	Truthfulness	Safety	Inst. Following	Average
-	Llama 2 base	-	52.0	37.0	30.7	32.7	32.7	-	-
-	TÜLU 2 (SFT)	-	55.4	47.8	45.1	56.6	91.8	44.2	56.8
Web	SHP-2	500,000	55.4	47.7	40.3	62.2	90.4	45.6	56.9
	StackExchange	500,000	55.7	46.8	39.6	67.4	92.6	44.6	57.8
Human	PRM800k	6,949	55.3	49.7	**46.6**	54.7	91.9	43.4	56.9
	Chatbot Arena (2023)	20,465	55.4	50.2	45.9	58.5	67.3	50.8	54.7
	Chatbot Arena (2024)	34,269	55.7	50.4	37.7	56.7	58.1	50.7	51.5
	AlpacaF. Human Pref	9,686	55.3	47.6	43.3	56.1	90.7	44.5	56.2
	HH-RLHF	158,530	54.7	46.0	43.6	65.6	**93.1**	45.4	58.1
	HelpSteer	9,270	55.2	48.2	46.5	60.3	92.5	45.2	58.0
Synthetic	AlpacaF. GPT-4 Pref	19,465	55.3	49.1	43.4	57.7	89.5	46.3	56.9
	Capybara 7k	7,563	55.2	46.4	46.4	57.5	91.5	46.1	57.2
	Orca Pairs	12,859	55.5	46.8	46.0	57.9	90.5	46.2	57.2
	Nectar	180,099	55.3	47.8	43.2	68.2	**93.1**	47.8	59.2
	UltraF. (overall)	60,908	**55.6**	48.8	46.5	67.6	92.1	51.1	60.3
	UltraF. (fine-grained)	60,908	55.3	**50.9**	45.9	**69.3**	91.9	**52.8**	**61.0**

Table 1: **Preference data:** Performance of TÜLU 2 13B models trained on various preference datasets using DPO. Blue indicates improvements over the SFT baseline, orange degradations. Overall, synthetic data works best. DPO training improves truthfulness and instruction-following most, with limited to no improvements in factuality and reasoning.

Experiments: Preference Data

Source		# Samples	Factuality	Reasoning	Coding	Truthfulness	Safety	Inst. Following	Average
-	Llama 2 base	-	52.0	37.0	30.7	32.7	32.7	-	-
-	TÜLU 2 (SFT)	-	55.4	47.8	45.1	56.6	91.8	44.2	56.8
Web	SHP-2	500,000	55.4	47.7	40.3	62.2	90.4	45.6	56.9
	StackExchange	500,000	55.7	46.8	39.6	67.4	92.6	44.6	57.8
Human	PRM800k	6,949	55.3	49.7	**46.6**	54.7	91.9	43.4	56.9
	Chatbot Arena (2023)	20,465	55.4	50.2	45.9	58.5	67.3	50.8	54.7
	Chatbot Arena (2024)	34,269	55.7	50.4	37.7	56.7	58.1	50.7	51.5
	AlpacaF. Human Pref	9,686	55.3	47.6	43.3	56.1	90.7	44.5	56.2
	HH-RLHF	158,530	54.7	46.0	43.6	65.6	**93.1**	45.4	58.1
	HelpSteer	9,270	55.2	48.2	46.5	60.3	92.5	45.2	58.0
Synthetic	AlpacaF. GPT-4 Pref	19,465	55.3	49.1	43.4	57.7	89.5	46.3	56.9
	Capybara 7k	7,563	55.2	46.4	46.4	57.5	91.5	46.1	57.2
	Orca Pairs	12,859	55.5	46.8	46.0	57.9	90.5	46.2	57.2
	Nectar	180,099	55.3	47.8	43.2	68.2	**93.1**	47.8	59.2
	UltraF. (overall)	60,908	**55.6**	48.8	46.5	67.6	92.1	51.1	60.3
	UltraF. (fine-grained)	60,908	55.3	**50.9**	45.9	**69.3**	91.9	**52.8**	**61.0**

Table 1: **Preference data:** Performance of TÜLU 2 13B models trained on various preference datasets using DPO. Blue indicates improvements over the SFT baseline, orange degradations. Overall, synthetic data works best. DPO training improves truthfulness and instruction-following most, with limited to no improvements in factuality and reasoning.

Experiments: Preference Data

Source		# Samples	Factuality	Reasoning	Coding	Truthfulness	Safety	Inst. Following	Average
-	Llama 2 base	-	52.0	37.0	30.7	32.7	32.7	-	-
-	TÜLU 2 (SFT)	-	55.4	47.8	45.1	56.6	91.8	44.2	56.8
Web	SHP-2	500,000	55.4	47.7	40.3	62.2	90.4	45.6	56.9
	StackExchange	500,000	55.7	46.8	39.6	67.4	92.6	44.6	57.8
Human	PRM800k	6,949	55.3	49.7	**46.6**	54.7	91.9	43.4	56.9
	Chatbot Arena (2023)	20,465	55.4	50.2	45.9	58.5	67.3	50.8	54.7
	Chatbot Arena (2024)	34,269	55.7	50.4	37.7	56.7	58.1	50.7	51.5
	AlpacaF. Human Pref	9,686	55.3	47.6	43.3	56.1	90.7	44.5	56.2
	HH-RLHF	158,530	54.7	46.0	43.6	65.6	**93.1**	45.4	58.1
	HelpSteer	9,270	55.2	48.2	46.5	60.3	92.5	45.2	58.0
Synthetic	AlpacaF. GPT-4 Pref	19,465	55.3	49.1	43.4	57.7	89.5	46.3	56.9
	Capybara 7k	7,563	55.2	46.4	46.4	57.5	91.5	46.1	57.2
	Orca Pairs	12,859	55.5	46.8	46.0	57.9	90.5	46.2	57.2
	Nectar	180,099	55.3	47.8	43.2	68.2	**93.1**	47.8	59.2
	UltraF. (overall)	60,908	**55.6**	48.8	46.5	67.6	92.1	51.1	60.3
	UltraF. (fine-grained)	60,908	55.3	**50.9**	45.9	**69.3**	91.9	**52.8**	**61.0**

- Synthetic data with *per-aspect annotations* performs best (*i.e., UltraF.*)
- Per-aspect annotations (i.e., *UltraF, HelpSteer*): Datasets collected by first getting per-aspect annotations (e.g., helpfulness, harmlessness) then averaging

A Recipe for Learning from Preferences

Preference data → High-quality, synthetic preference dataset

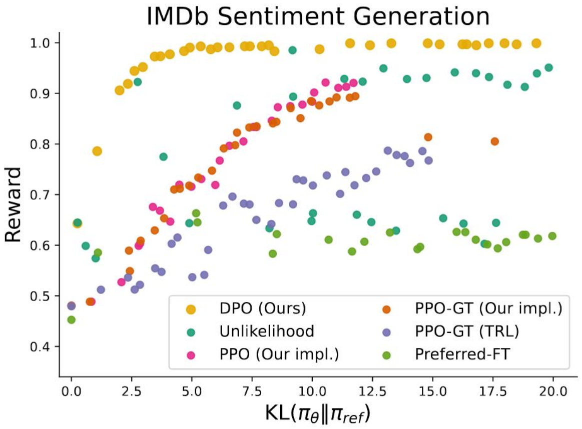
Preference Learning Algorithm

Reward model

Policy training prompt

Experiments: Preference Learning Algorithm (DPO vs. PPO)

GPT-2-L (774M)



GPT-J (6B)

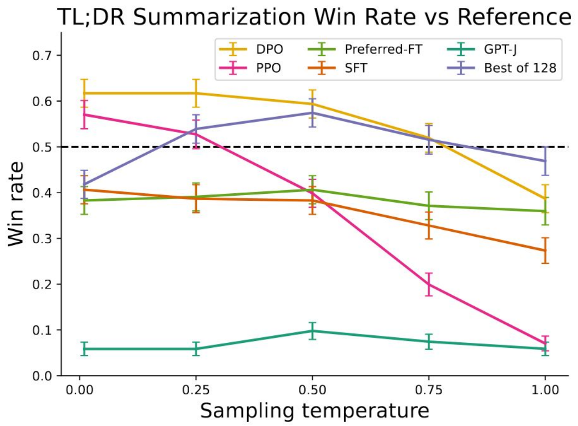


Figure 2: **Left.** The frontier of expected reward vs KL to the reference policy. DPO provides the highest expected reward for all KL values, demonstrating the quality of the optimization. **Right.** TL;DR summarization win rates vs. human-written summaries, using GPT-4 as evaluator. DPO exceeds PPO's best-case performance on summarization, while being more robust to changes in the sampling temperature.

R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C. D. Manning, and C. Finn. *Direct preference optimization: Your language model is secretly a reward model*

Experiments: Preference Learning Algorithm (DPO vs. PPO)

Data / Model	Alg.	Factuality	Reasoning	Coding	Truthfulness	Safety	Inst. Foll.	Average
Llama 2 base	-	52.0	37.0	30.7	32.7	32.7	-	-
TÜLU 2 (SFT)	-	55.4	47.8	45.1	56.6	91.8	44.2	56.8
StackExchange	DPO	**55.3**	**47.8**	42.4	**56.2**	92.0	46.7	56.7
	PPO	55.1	**47.8**	**46.4**	54.2	**92.6**	**47.4**	**57.3**
ChatArena (2023)	DPO	**55.4**	**50.2**	45.9	**58.5**	67.3	**50.8**	54.7
	PPO	55.2	49.2	**46.4**	55.8	**79.4**	49.7	**55.9**
HH-RLHF	DPO	**55.2**	47.6	44.2	**60.0**	**93.4**	46.6	57.8
	PPO	54.9	**48.6**	**45.9**	58.0	92.8	**47.0**	**57.9**
Nectar	DPO	**55.6**	45.8	39.0	**68.1**	**93.3**	**48.4**	58.4
	PPO	55.2	**51.2**	**45.6**	60.1	92.6	47.4	**58.7**
UltraFeedback (FG)	DPO	55.3	50.9	45.9	69.3	**91.9**	52.8	61.0
	PPO	**56.0**	**52.0**	**47.7**	**71.5**	91.8	**54.4**	**62.2**
Avg. Δ b/w PPO & DPO		-0.1	+1.3	+2.9	-2.5	+2.3	+0.1	+0.7

Table 2: **DPO vs PPO:** Average performance of 13B models trained using DPO and PPO across different datasets, along with the performance difference between DPO and PPO (Δ). Blue indicates improvements over the SFT baseline, orange degradations. All datasets are downsampled to 60,908 examples (except ChatArena, which is made up of 20,465 responses). PPO outperforms DPO by an average of 0.7 points, where most improvements are in reasoning, coding, and chat capabilities.

A Recipe for Learning from Preferences

Preference data	High-quality, synthetic preference dataset
Preference Learning Algorithm	PPO
Reward model	
Policy training prompt	

Experiments: Reward Models

- Scaling up the training data for RM:
  - **Mix RM**: Construct a data mixture of the top performing preference datasets (i.e., UltraFeedback, HelpSteer, Nectar, StackExchange, HH-RLHF, PRM800k)
  - **UltraF. RM**: Reward model trained only on UltraFeedback

- Scaling up the reward model size: 13B and 70B

Experiments: Reward Models

Reward Model	Direct Eval.	
	RewardBench Score	Best-of-N over SFT Avg. Perf. (Δ)
13B UltraF. RM	61.0	56.9 (+5.8)
13B Mix RM	**79.8**	58.3 (+7.3)
70B UltraF. RM	73.6	**61.1 (+10.3)**
70B Mix RM	73.9	60.6 (+9.5)

- Best-of-N:
  - Sample 16 responses for each evaluation task
  - Pick the top-scoring response according to the RM as the final output
- RewardBench: evaluating if the relative scores given by reward models match a test set of chosen-rejected pairs from diverse sources

Experiments: Reward Models

Reward Model	Direct Eval.		PPO Training Perf. (w. UltraF. prompts)		
	RewardBench Score	Best-of-N over SFT Avg. Perf. (Δ)	GSM Acc.	AlpacaEval2 winrate	Avg. on All Evals.
13B UltraF. RM	61.0	56.9 (+5.8)	53.0	26.1	62.2
13B Mix RM	**79.8**	58.3 (+7.3)	51.0	25.7	61.6
70B UltraF. RM	73.6	**61.1 (+10.3)**	**58.0**	26.7	**62.8**
70B Mix RM	73.9	60.6 (+9.5)	51.5	**31.6**	61.8

- It is difficult to translate improvements in reward models to the underlying policy
- Increasing scale and data improves reward models, but these only minimally impact the average downstream performance

A Recipe for Learning from Preferences

Preference data	High-quality, synthetic preference dataset
Preference Learning Algorithm	PPO
Reward model	Large reward model
Policy training prompt	

Experiments: Policy Training Prompts

- UltraF. Prompts: 20 random prompts from UltraFeedback
- Mined Math: math-related prompts from varied dataset
- GSM train: prompts from GASM train set

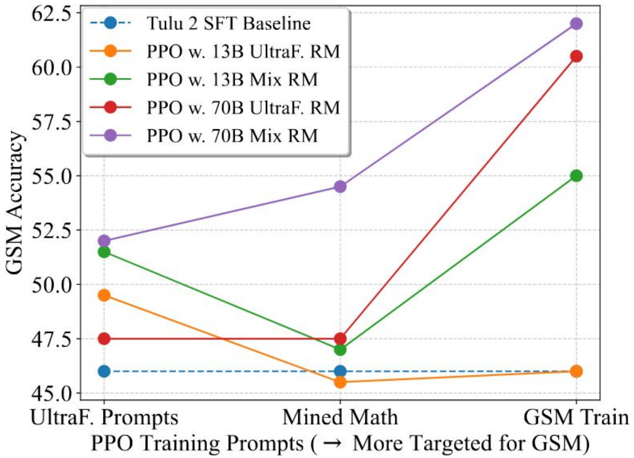


Figure 3: **Policy training prompt math evaluation:** Performance of models trained on 20K prompts from varying sources using PPO and evaluated on GSM. Training with larger RMs trained on more data benefits more from in-domain prompts (i.e., prompts directly from the GSM train set), while weaker RMs struggle to generalize beyond their training prompts.

Experiments: Policy Training Prompts

- **Mixed**: mix math- and code-related prompts with UltraFeedback prompts, then downsample to the same size of UF
- **UF**: UltraFeedback

Reward Model	Prompts	GSM %	Coding	Avg. Across All Evals
Tulu 2 SFT	-	46.0	45.1	56.8
13B UltraF.	UF	53.0	47.7	**62.2**
13B UltraF.	Mixed	**54.5**	**47.8**	61.9
13B Mix	UF	**51.0**	**46.8**	**61.6**
13B Mix	Mixed	50.5	43.8	60.9
70B UltraF.	UF	**58.0**	47.3	**62.8**
70B UltraF.	Mixed	56.5	**48.4**	62.4
70B Mix	UF	51.5	**46.1**	**61.8**
70B Mix	Mixed	**52.0**	44.9	61.1

Table 4: **Policy training prompt overall evaluation:** Performance of PPO policy models trained with the given reward models on 60K prompts from either UltraFeedback or the remixed prompt set that adds additional unlabeled math and coding-related prompts. Using the remixed prompt set does not improve performance, either on specific evaluations (math, code) or in terms of overall performance.

A Recipe for Learning from Preferences

Preference data	High-quality, synthetic preference dataset
Preference Learning Algorithm	PPO
Reward model	Large reward model
Policy training prompt	Domain-specific prompts for policy training (if focus on a specific domain)

A Recipe for Learning from Preferences

Model	Factuality	Reasoning	Coding	Truthfulness	Safety	Instr. Foll.	Average
Llama 2 Chat 13B [52]	53.2	24.7	36.9	**88.0**	91.9	51.2	57.7
Nous Hermes 13B [51]	53.2	43.5	47.7	80.5	43.9	38.7	51.3
Vicuna 1.5 13B [64]	54.5	39.3	38.5	62.8	92.4	45.8	55.6
Llama 2 13B Base	52.0	37.0	30.7	32.7	32.7	-	-
TÜLU 2 13B SFT	55.4	47.8	45.1	56.6	91.8	44.2	56.8
TÜLU 2+DPO 13B	55.3	50.9	45.9	69.3	91.9	52.8	61.0
TÜLU 2+PPO 13B (13B UFRM)	**56.0**	52.0	47.7	71.5	91.8	54.4	62.2
TÜLU 2+PPO 13B (70B UFRM)	55.4	**53.9**	47.3	72.3	91.9	**55.8**	**62.8**
TÜLU 2+PPO 13B (70B UFRM+MP)	55.3	53.1	**48.4**	71.0	**92.7**	54.0	62.4

Table 5: **Putting together a recipe for preference-based learning:** Performance of our best-performing models along with popular open models based on Llama 2 13B. 'MP' refers to using the mixed prompt set described in §4. Using PPO with a large reward model performs best overall.

Summary

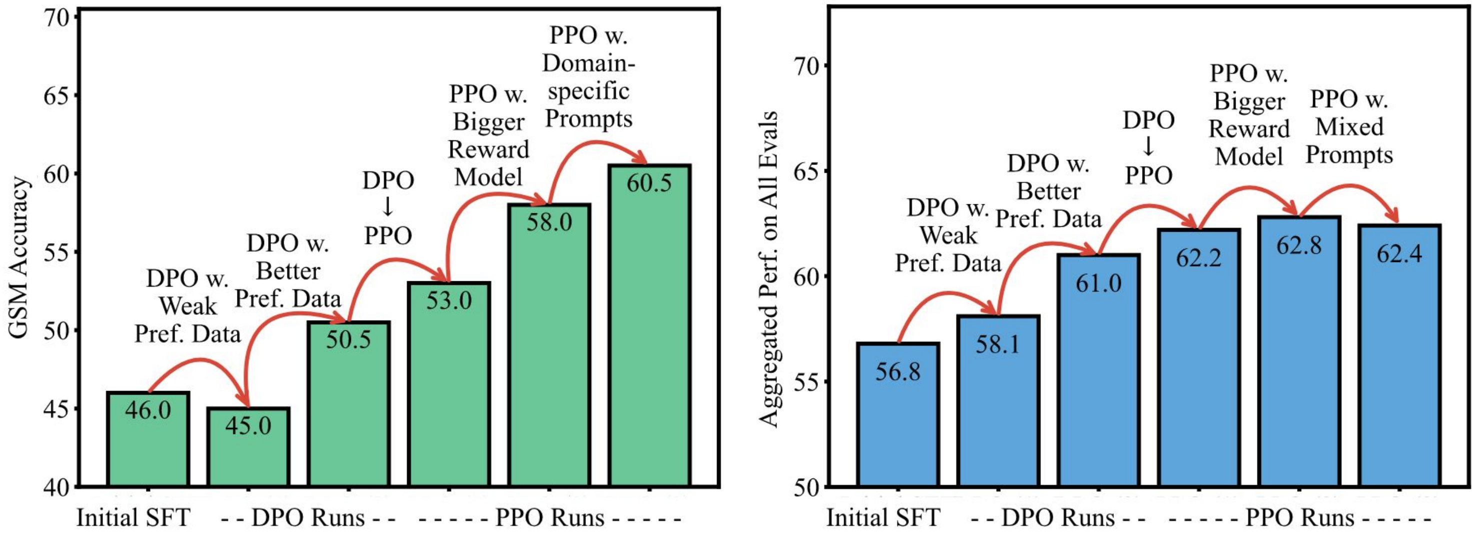


Figure 1: Performance improvements resulted by changing different components in the preference training of TÜLU. Left: Accuracy on GSM [9], for testing math capabilities. Right: Overall performance, aggregated over the 11 benchmarks described in §2.2.