中国科学技术大学
University of Science and Technology of China

β-DPO: Direct Preference Optimization with Dynamic β

Authors: Junkang Wu, Yuexiang Xie, Zhengyi Yang, Jiancan Wu, Jinyang Gao, Bolin Ding, Xiang Wang, Xiangnan He

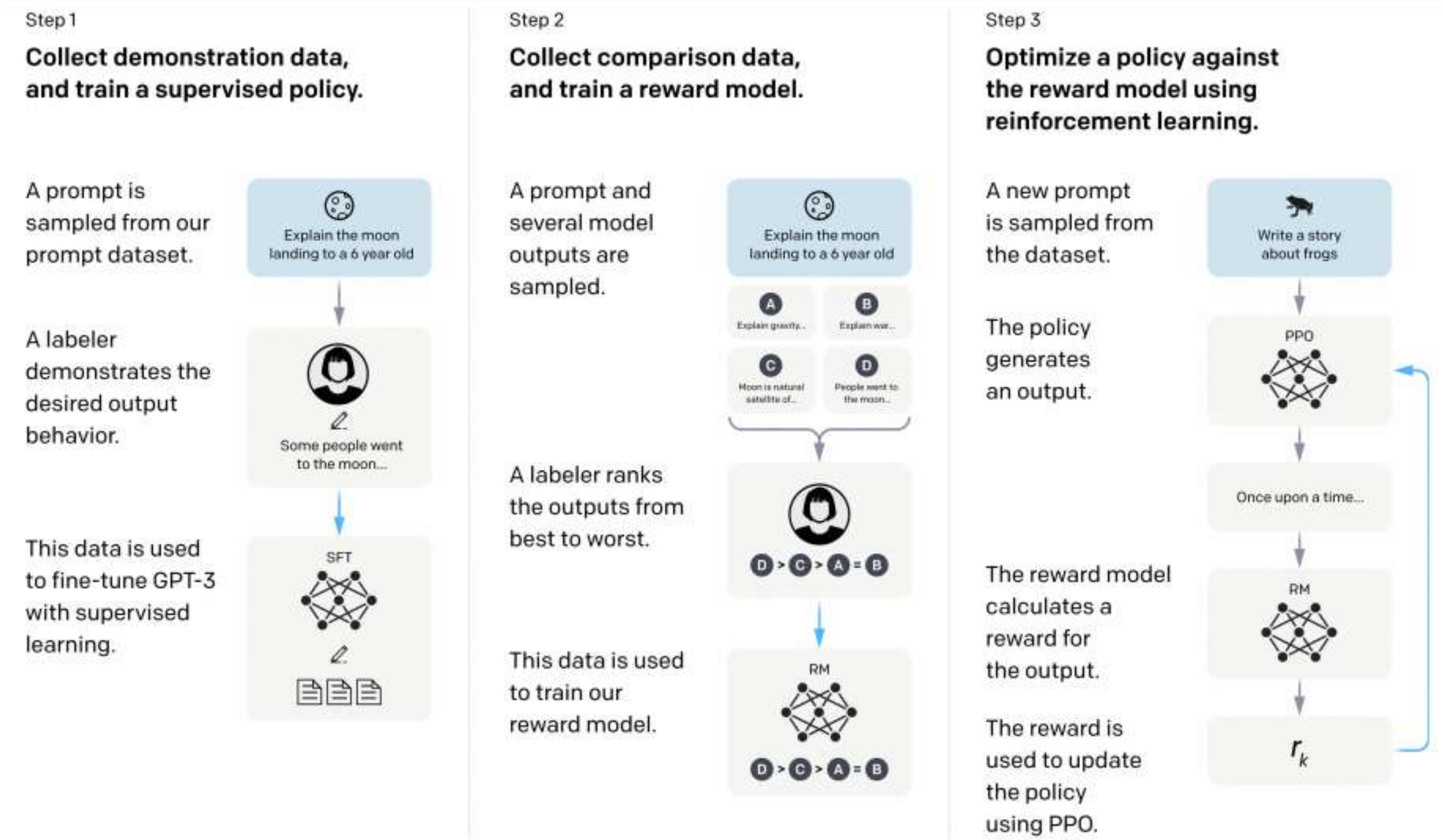
Paper: https://arxiv.org/abs/2407.08639

Code: https://github.com/junkangwu/beta-DPO

Date: Oct 16, 2024

Background and Motivation

❑ RLHF



Ouyang, et. al. Training language models to follow instructions with human feedback. NeurIPS 2022.

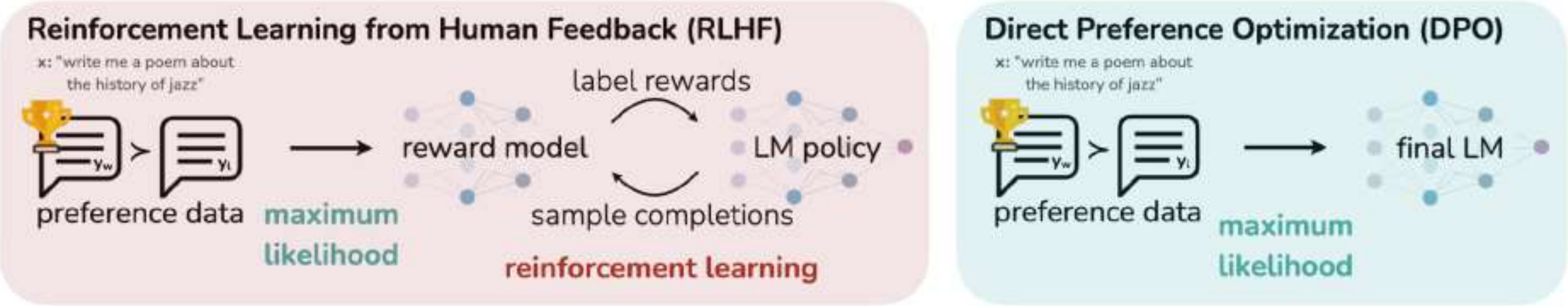
Background and Motivation

- **RLHF – why we need RL**
  - We use RL training because supervised training teaches the model to **lie**
    1) If the model "knows" the answer, the supervised training associates the answer with the question.
    2) If the model does not know the answer, the supervised training **pushes** the model to associate the answer with the question anyhow.
- **The limitations of RL**
  - Instability
  - High computational cost

Ouyang, et. al. Training language models to follow instructions with human feedback. NeurIPS 2022.

Background and Motivation

- **RLHF is a complex and often unstable procedure**
  - Eliminating the need for fitting a reward model



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

$$\pi_r(y\mid x) = \frac{1}{Z(x)}\pi_{\mathrm{ref}}(y\mid x)\exp\left(\frac{1}{\beta}r(x,y)\right),$$

$$r(x,y) = \beta\log\frac{\pi_r(y\mid x)}{\pi_{\mathrm{ref}}(y\mid x)} + \beta\log Z(x).$$

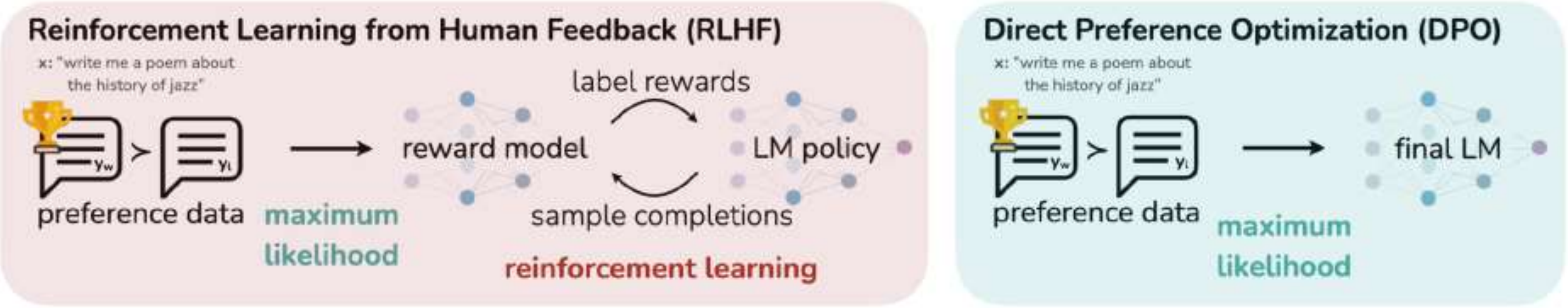
$$p^*(y_1 \succ y_2 \mid x) = \frac{\exp\left(r^*(x,y_1)\right)}{\exp\left(r^*(x,y_1)\right) + \exp\left(r^*(x,y_2)\right)}.$$

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

Rafael Rafailov, et al. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. NeurIPS 2023

Background and Motivation

- **RLHF is a complex and often unstable procedure**
  - Eliminating the need for fitting a reward model



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

$$\pi_r(y\mid x) = \frac{1}{Z(x)}\pi_{\text{ref}}(y\mid x)\exp\left(\frac{1}{\beta}r(x,y)\right),$$

$$r(x,y) = \beta\log\frac{\pi_r(y\mid x)}{\pi_{\text{ref}}(y\mid x)} + \beta\log Z(x).$$

$$p^*(y_1 \succ y_2 \mid x) = \frac{\exp\left(r^*(x,y_1)\right)}{\exp\left(r^*(x,y_1)\right)+\exp\left(r^*(x,y_2)\right)}.$$

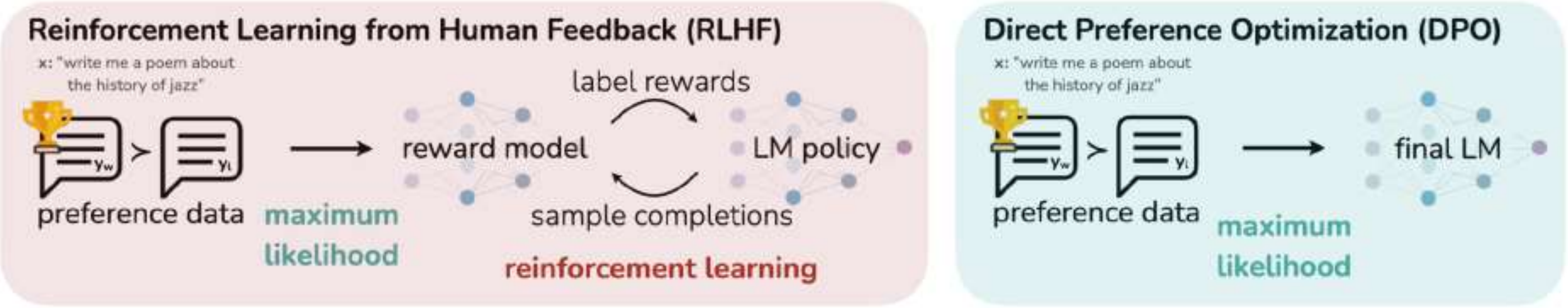
$$\mathcal{L}_R(r_\phi,\mathcal{D}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma(r_\phi(x,y_w)-r_\phi(x,y_l))\right]$$

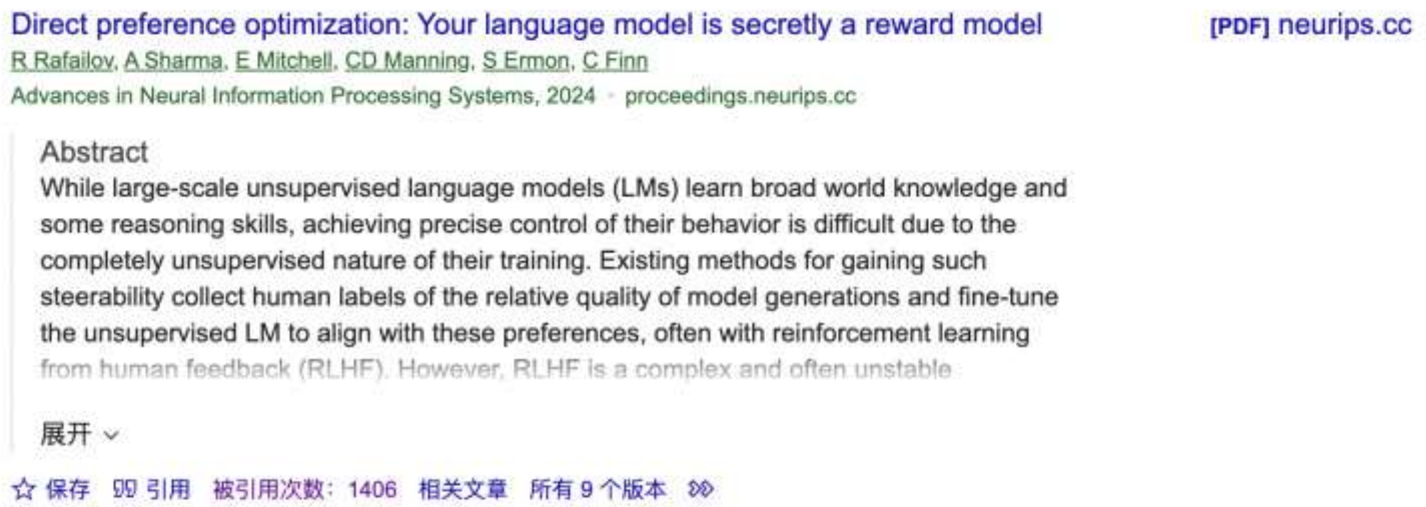
$$\mathcal{L}_{\text{DPO}}(\pi_\theta;\pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w\mid x)}{\pi_{\text{ref}}(y_w\mid x)} - \beta\log\frac{\pi_\theta(y_l\mid x)}{\pi_{\text{ref}}(y_l\mid x)}\right)\right].$$

Rafael Rafailov, et al. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. NeurIPS 2023

Background and Motivation

- **RLHF is a complex and often unstable procedure**
  - Eliminating the need for fitting a reward model




Direct preference optimization: Your language model is secretly a reward model

R Rafailov, A Sharma, E Mitchell, CD Manning, S Ermon, C Finn

Advances in Neural Information Processing Systems, 2024 · proceedings.neurips.cc

[PDF] neurips.cc

Abstract

While large-scale unsupervised language models (LMs) learn broad world knowledge and some reasoning skills, achieving precise control of their behavior is difficult due to the completely unsupervised nature of their training. Existing methods for gaining such steerability collect human labels of the relative quality of model generations and fine-tune the unsupervised LM to align with these preferences, often with reinforcement learning from human feedback (RLHF). However, RLHF is a complex and often unstable

展开

☆ 保存 99 引用 被引用次数：1406 相关文章 所有 9 个版本

$$\pi_{\text{ref}}(y \mid x)\exp\left(\frac{1}{\beta}r(x,y)\right),$$

$$\frac{\pi_r(y \mid x)}{\pi_{\text{ref}}(y \mid x)} + \beta\log Z(x).$$

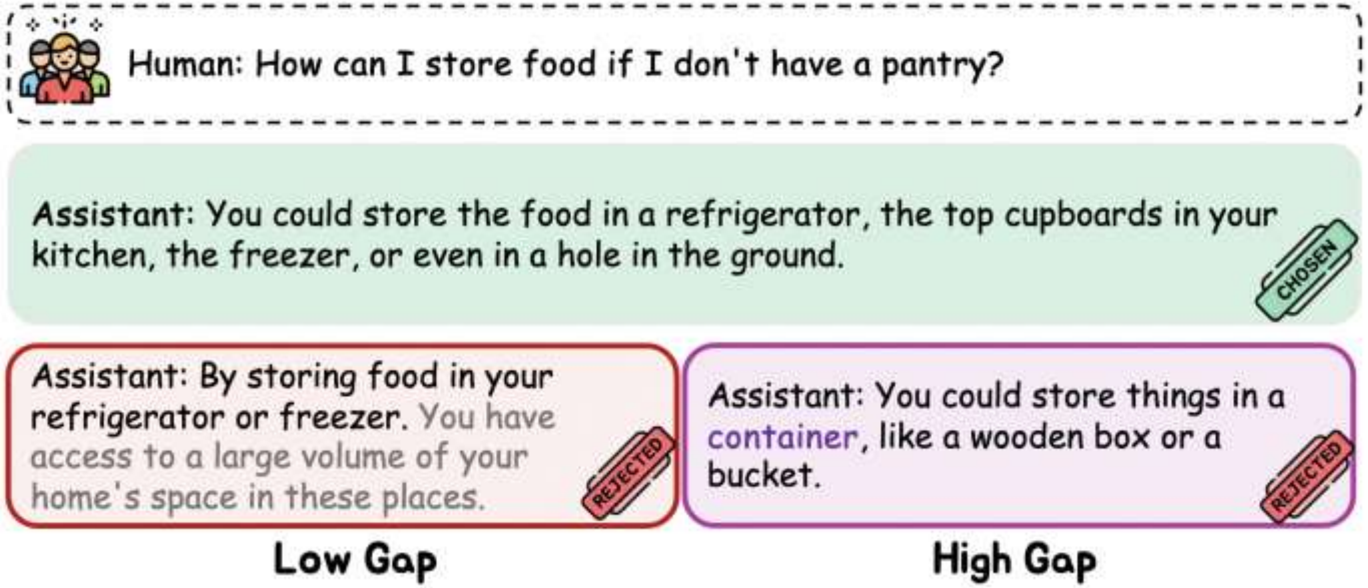
$$\frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\Bigg)\Bigg].$$

Rafael Rafailov, et al. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. NeurIPS 2023

❑ The Impact of Pairwise Data Quality on β Selection

Dataset: Anthropic HH

- ✓ *low gap* denotes cases where the chosen and rejected examples are **closely similar**, typically indicating high-quality, informative pairs.
- ✓ *High gap* signifies pairs with **larger differences**, implying lower-quality data.



Background and Motivation

❑ The Impact of Pairwise Data Quality on $\boldsymbol{\beta}$ Selection

Dataset: Anthropic HH

- ✓ *low gap* denotes cases where the chosen and rejected examples are **closely similar**, typically indicating high-quality, informative pairs.
- ✓ *High gap* signifies pairs with **larger differences**, implying lower-quality data.

Models: Pythia-410M, -1.4B, and -2.8B

Metrics: win rate

Background and Motivation

❑ The Impact of Pairwise Data Quality on $\boldsymbol{\beta}$ Selection

- The optimal value of $\boldsymbol{\beta}$ **varies with data quality**, reflecting divergent performance patterns across datasets.
- The dataset exhibits notable **outliers**.

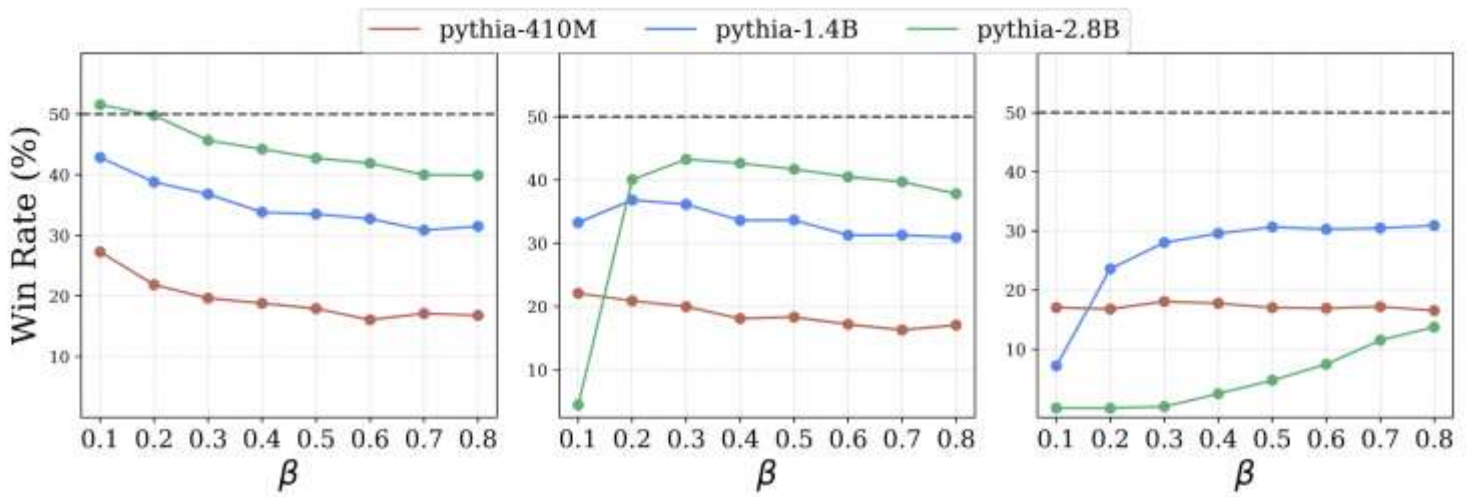


Figure 2: Win rate performance of DPO across different β settings on the *low gap*, *mixed gap*, and *high gap* datasets.

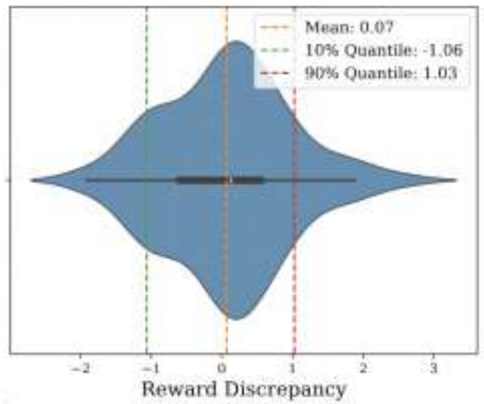


Figure 3: The distribution of individual reward discrepancy $(r(\mathbf{y}_w^{(i)};\mathbf{x}^{(i)}) - r(\mathbf{y}_l^{(i)};\mathbf{x}^{(i)}))$ on the training dataset of HH.

Background and Motivation

❑ The Impact of Pairwise Data Quality on β Selection

- The optimal value of β **varies with data quality**, reflecting divergent performance patterns across datasets.
- The dataset exhibits notable **outliers**.

Principle 1: The optimal β value should be responsive to pairwise data's quality.
Principle 2: The selection of β value should minimize the influence of outliers

Method: $\boldsymbol{\beta}$-DPO

❑ Dynamic $\boldsymbol{\beta}$ Calibration at Batch-Level

- ➢ Define the reward discrepancy $M = \beta_0 \log\left(\frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)}\right) - \beta_0 \log\left(\frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\right).$
- ➢ Instance-level dynamic $\boldsymbol{\beta}$ adaptation

$$\beta_i = \beta_0 + \alpha(M_i - M_0)\beta_0 = [1 + \alpha(M_i - M_0)]\beta_0,$$

- β_0 is the DPO benchmark hyperparameter (typically 0.1),
- M_0 is a threshold.
- $\alpha \in [0,1]$ scales M_i's influence on β_i.
- When $\alpha = 0$, $\beta_i = \beta_0$ (standard DPO)

Method: $\boldsymbol{\beta}$-DPO

❑ Dynamic $\boldsymbol{\beta}$ Calibration at Batch-Level

- Define the reward discrepancy $M = \beta_0 \log\left(\frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)}\right) - \beta_0 \log\left(\frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\right).$
- Instance-level dynamic $\boldsymbol{\beta}$ adaptation

$$\beta_i = \beta_0 + \alpha(M_i - M_0)\beta_0 = [1 + \alpha(M_i - M_0)]\beta_0,$$

- Batch-level dynamic estimation methodology

$$\beta_{\text{batch}} = [1 + \alpha(\mathbb{E}_{i\sim\text{batch}}[M_i] - M_0)]\beta_0.$$

- Estimate M_0 with moving average updating scheme.

$$M_0 \leftarrow mM_0 + (1-m)\mathbb{E}_{i\sim\text{batch}}[M_i],$$

Method: $\boldsymbol{\beta}$-DPO

❑ $\boldsymbol{\beta}$-Guided Data Filtering

- Define the importance of each triplet (x, y_w, y_l)

$$p(M_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(M_i - M_0)^2}{2\sigma^2}\right),$$

  - M_0 and σ represent the mean and standard deviation of M_{i} across the training dataset.

- Dynamically estimate the value ofσusing the moving average method:

$$\sigma \leftarrow m\sigma + (1-m)\sqrt{\mathbb{V}_{i\sim\text{batch}}[M_i]}.$$

Method: $\boldsymbol{\beta}$-DPO

❑ $\boldsymbol{\beta}$-Guided Data Filtering

- Define the importance of each triplet (x, y_w, y_l)

$$p(M_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(M_i - M_0)^2}{2\sigma^2}\right),$$

  - M_0 and σ represent the mean and standard deviation of M_i across the training dataset.

- Dynamically estimate the value ofσusing the moving average method:

$$\sigma \leftarrow m\sigma + (1-m)\sqrt{\mathbb{V}_{i\sim\text{batch}}[M_i]}.$$

Note: It is important to highlight that this work does not propose a novel filtering method, but we find that filtering enhances stability.

Method: β-DPO

❑ Highlights of β-DPO

- **Simplicity**: Easy to implement with dynamic β adjustment and data filtering
- **Efficiency**: No additional gold model needed; insensitive to hyperparameters
- **Model-agnostic**: Plug-and-play module compatible with future DPO enhancements.

Experiment

❑ Dialogue Generation and Summarization

Win Rate Across different Sampling Temperature

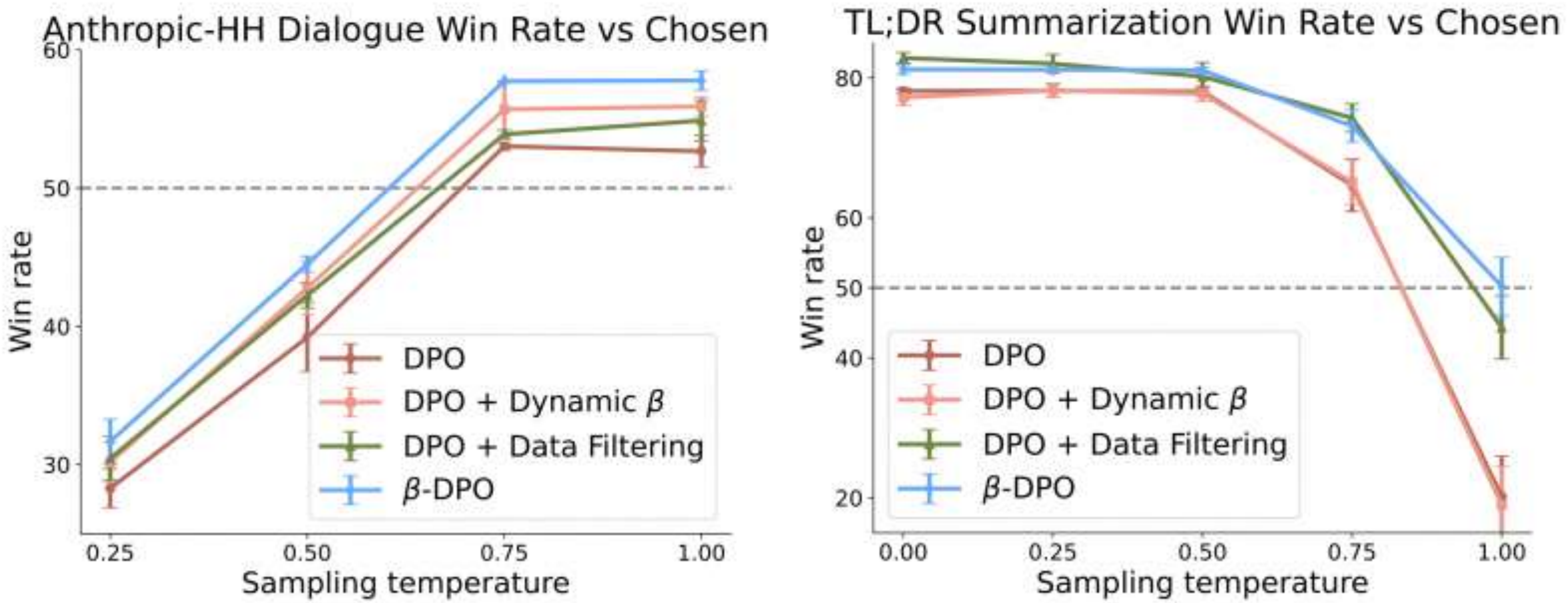


Figure 4: **Left.** The win rates computed by GPT-4 evaluations for the Anthropic-HH one-step dialogue; β-DPO consistently outperforms across all sampling temperatures. **Right.** In the comparison of TL;DR summarization win rates versus chosen summaries with GPT-4 as the evaluator, β-DPO is distinguished as the only strategy achieving a win rate over 50% across different sampling temperatures.

Experiment

❑ Dialogue Generation and Summarization

Win Rate Across different Model Sizes

Table 1: Win rate comparison of Pythia-410M, -1.4B, and -2.8B models on the Anthropic HH dataset, evaluated using GPT-4.

Method	**410M**	**1.4B**	**2.8B**
DPO	26.19	42.78	51.51
DPO + Dynamic β	$27.15^{+3.67\%}$	$43.51^{+1.71\%}$	$55.19^{+7.14\%}$
DPO + Data Filtering	$29.03^{+10.84\%}$	$46.99^{+9.84\%}$	$53.42^{+3.71\%}$
β-DPO	$30.18^{+15.23\%}$	$48.67^{+13.77\%}$	$57.07^{+10.79\%}$

Experiment

❑ Adaptations of $\boldsymbol{\beta}$-DPO

✓ Selective **filtering of the top 20%** of samples markedly enhances model performance.

✓ Dynamic $\boldsymbol{\beta}$ adapts to and improves upon **existing filtering strategies**.

✓ Dynamic $\boldsymbol{\beta}$ Enhancement **across DPO Variants.**

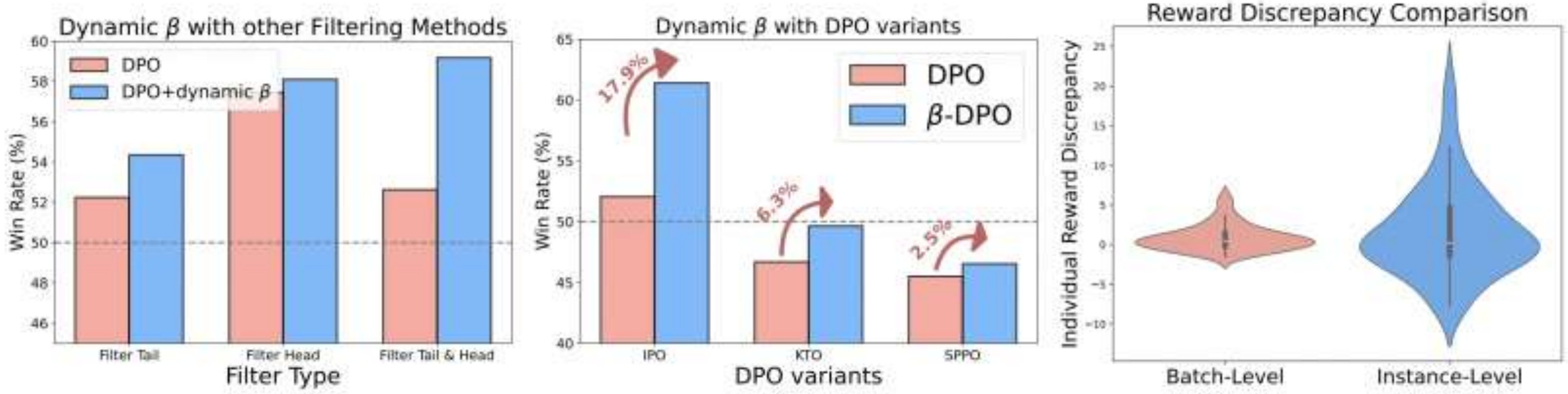


Figure 5: **Left:** Win rates from GPT-4 evaluations on Anthropic-HH single-turn dialogues, showcasing β-DPO's adaptability to diverse filtering strategies. **Middle:** Win rates of β-DPO across various DPO variants as evaluated by GPT-4. **Right:** Distribution of individual reward discrepancies following fine-tuning through batch-level and instance-level calibration.

Experiment

❑ Necessity of Batch-Level Dynamic β Calibration

✓ Batch-level calibration surpasses both instance-level and population-level approaches.

✓ Instance-level calibration **magnifies** the impact of outliers.

Table 2: Comparison of win rates across varying mixture ratios on the Anthropic HH dataset, with each ratio indicating the proportion of *high-gap* to *low-gap* datasets, e.g., a 40% mixture ratio reflects a blend of 40% *high-gap* and 60% *low-gap*.

Mixture Ratio	**10%**	**20%**	**30%**	**40%**
Vanilla DPO	50.17	50.56	47.95	29.15
+ Instance-level calibration	$49.18^{-1.97\%}$	$49.82^{-1.46\%}$	$44.42^{-7.36\%}$	$16.82^{-42.30\%}$
+ Batch-level calibration	$57.68^{+14.69\%}$	$56.15^{+11.06\%}$	$51.25^{+6.88\%}$	$34.92^{+19.79\%}$

Experiment

❑ Necessity of Batch-Level Dynamic β Calibration

- ✓ Batch-level calibration surpasses both instance-level and population-level approaches
- ✓ Instance-level calibration **magnifies** the impact of outliers.
- ✓ Our β-calibration strategy consistently outperforms baseline methods.

Table 5: Comparison of different methods on Llama3-Instruct (8B) with explicit reward model

Method	Llama3-Instruct (8B)	Llama3-Instruct (8B)
	LC (%)	WR (%)
DPO (Implicit RM)	40.44	37.38
β-DPO (Implicit RM)	**43.38**	**38.21**
SimPO (Implicit RM)	44.38	38.97
β-SimPO (Implicit RM)	**46.03**	**40.18**
SimPO (PairRM)	44.70	38.98
β-SimPO (PairRM, Instance-Level)	43.84	38.54
β-SimPO (PairRM, Batch-Level)	**45.65**	**39.76**
SimPO (ArmoRM)	53.70	47.50
β-SimPO (ArmoRM, Instance-Level)	49.05	45.47
β-SimPO (ArmoRM, Batch-Level)	**54.86**	**49.66**

Conclusion

- Introduction of $\boldsymbol{\beta}$-DPO:
  - Dynamically adjusts $\boldsymbol{\beta}$ parameter based on pairwise data informativeness
- Key Components:
  - $\boldsymbol{\beta}$-guided data filtering
  - Batch-level dynamic $\boldsymbol{\beta}$ calibration
- Results and Implications:
  - Significant performance improvements across various models and datasets
  - Offers an adaptable training paradigm for Large Language Models (LLMs) with human feedback

Limitations and Future Work

- Adaptive $\boldsymbol{\beta}$ in Self-Play
  - Explore dynamic $\boldsymbol{\beta}$ adjustments in self-play scenarios
  - Aim to evolve superior model strategies

- Automated Parameter Tuning
  - Pursue automation in $\boldsymbol{\beta}$ tuning

Dr. DPO

- An enhancement to DPO that addresses **label flipping noise** in training datasets with distributionally robust optimization.

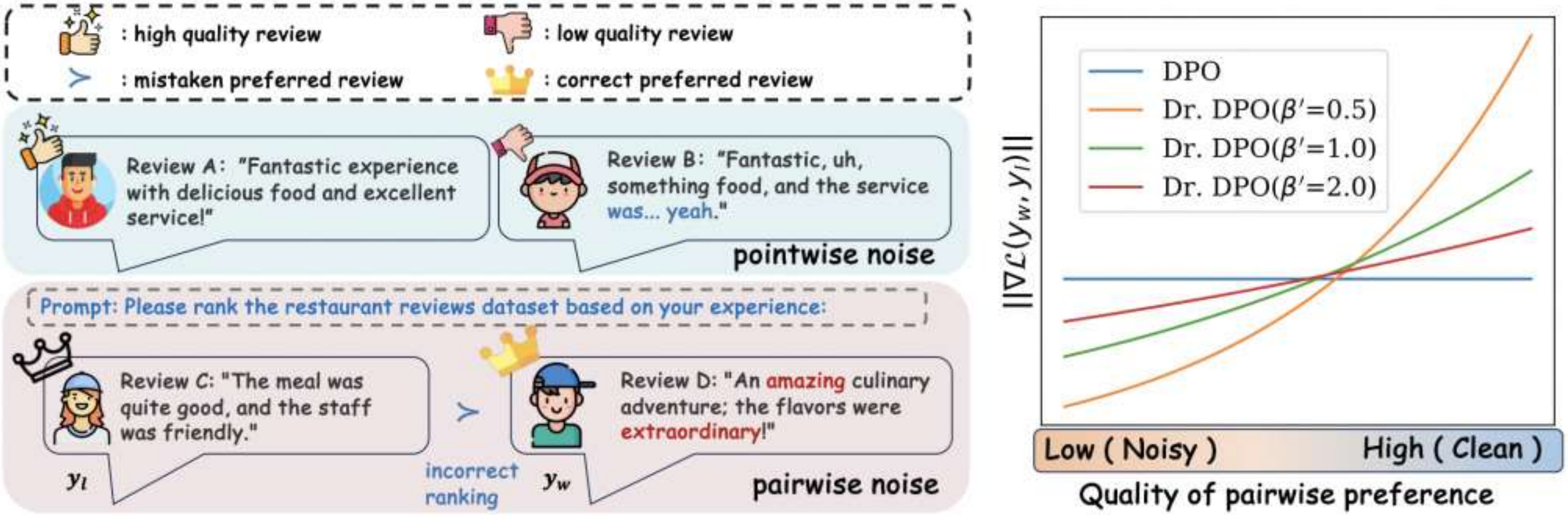
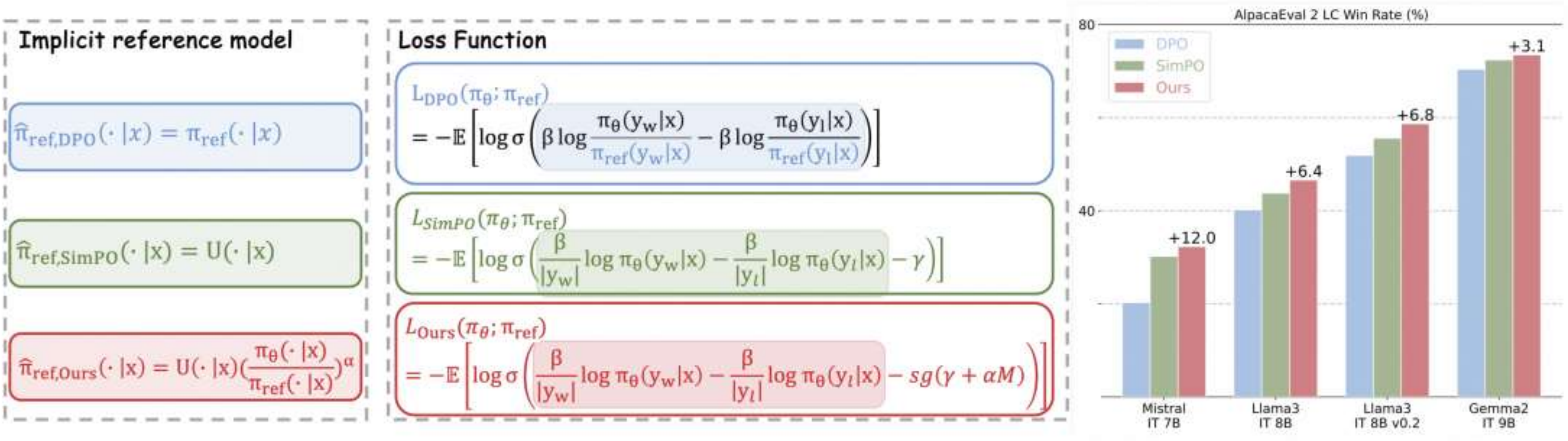


Figure 1: **Left**: An example illustrating pointwise and pairwise noise. **Right**: Comparison of gradients between DPO and Dr. DPO under varying levels of pairwise noise.

Wu, et. al. Towards Robust Alignment of Language Models: Distributionally Robustifying Direct Preference Optimization. under review.

α-DPO

- Addressing limitations in previous methods like DPO and SimPO by **balancing alignment and diversity through KL divergence.**



Wu, et. al. α-DPO: Adaptive Reward Margin is What Direct Preference Optimization Needs. under review.

Thanks