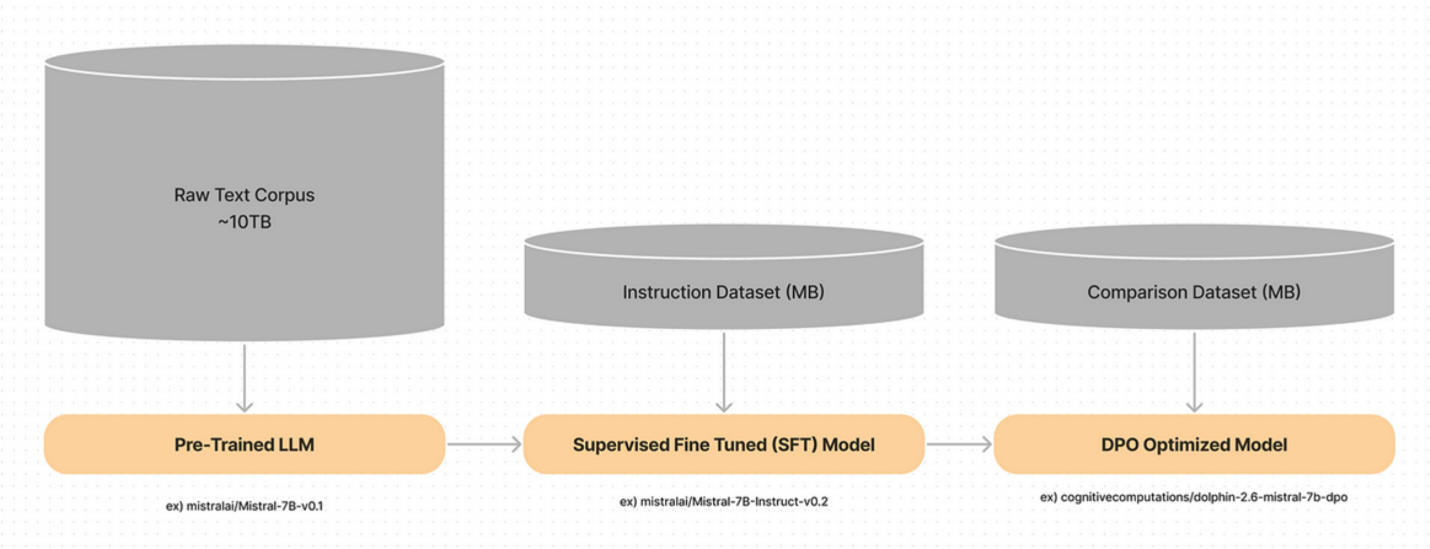
DPO

DPO首先创建人类偏好对的数据集，每个偏好对都包含一个prompt和两种可能的答案：一种是prefer，一种是disprefer的。然后通过对policy model进行优化，最大限度提升生成prefer答案的可能性。

用“一个简单的分类损失”来替换奖励模型，直接对人类的偏好进行预测。DPO是一种隐式优化与现有 RLHF 算法相同目标的算法（用 KL 散度约束的奖励最大化），但易于实现且易于训练。DPO也依赖于Bradley-Terry 模型（衡量奖励函数和经验偏好数据的匹配程度），与PPO需要训练一个奖励模型，通过最小化偏好的损失函数来对人类策略进行更新，以最大化奖励函数并拟合人类的偏好不同，DPO直接将偏好损失定义为policy中的一个函数，使用简单的二元交叉熵目标来优化策略，从而生成匹配偏好数据的隐式奖励函数的最优策略。



1. 基本概念

SFT：通过对预训练的LLM模型进行有监督微调，得到模型 π^{SFT}

奖励模型：使用SFT模型，根据prompt x 生成答案对 $(y1, y2) \sim \pi^{SFT}(y|x)$ ，人类对标注答案进行标注，得到prefer答案 y_w 和disprefer答案 y_l 。为了拟合人类的偏好，常用的偏好模型包括Bradley-Terry模型，人类偏好分布p*可以写为：

$$p^*(y_1 \succ y_2 \mid x) = \frac{\exp\left(r^*(x, y_1)\right)}{\exp\left(r^*(x, y_1)\right) + \exp\left(r^*(x, y_2)\right)}. \tag{1}$$

其中r*是隐性奖励模型，根据从p*采样的训练数据集 $\mathcal{D} = \{x^i, y_w^i, y_l^i\}_{i=1}^N$ ，可以参数化奖励模型r，并通过最大似然估计参数。将问题转化成二分类任务，其负对数似然损失为：

$$\mathcal{L}_R(r_\phi, \mathcal{D}) = -\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}}\left[\log \sigma(r_\phi(x, y_w) - r_\phi(x, y_l))\right] \tag{2}$$

在现实应用中，奖励模型r用SFT模型+一个线性层初始化。为了确保奖励函数具有较低的方差，对奖励进行归一化

$$\mathbb{E}_{x,y\sim\mathcal{D}}\left[r_\phi(x,y)\right] = 0 \text{ for all } x.$$

策略模型：基于奖励模型对policy π_θ 进行更新：

$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta(y|x)}\left[r_\phi(x,y)\right] - \beta\mathbb{D}_{\mathrm{KL}}\left[\pi_\theta(y\mid x)\mid\mid\pi_{\mathrm{ref}}(y\mid x)\right], \tag{3}$$

其中 π_{ref} 表示参考策略，一般使用SFT模型。π_θ 也用SFT模型初始化，第二项的KL散度是为了防止 π_θ 偏离奖励模型准确的分布，并保证模型模型生成的多样性。

2. DPO

区别于前边需要学习奖励模型，并通过其对强化学习策略进行更新，DPO利用了一种奖励模型参数化方法，可以以闭合形式提取其最优策略，而无需 RL 训练循环。利用从奖励函数到最优策略的映射，将奖励函数的损失函数转换为策略的损失函数。

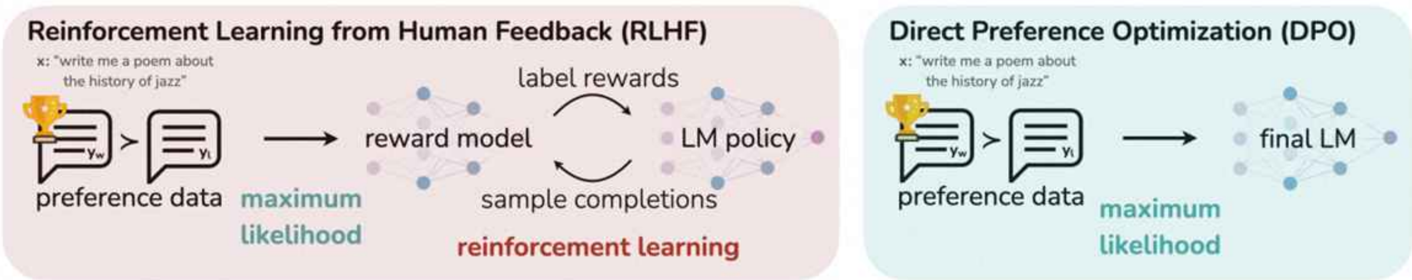


Figure 1: **DPO optimizes for human preferences while avoiding reinforcement learning.** Existing methods for fine-tuning language models with human feedback first fit a reward model to a dataset of prompts and human preferences over pairs of responses, and then use RL to find a policy that maximizes the learned reward. In contrast, DPO directly optimizes for the policy best satisfying the preferences with a simple classification objective, fitting an *implicit* reward model whose corresponding optimal policy can be extracted in closed form.

首先对公式3进行改写，

$$\begin{aligned}
&\max_{\pi} \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi}\left[r(x,y)\right] - \beta\mathbb{D}_{\mathrm{KL}}\left[\pi(y|x)\mid\mid\pi_{\mathrm{ref}}(y|x)\right] \\
&\qquad = \max_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi(y|x)}\left[r(x,y) - \beta\log\frac{\pi(y|x)}{\pi_{\mathrm{ref}}(y|x)}\right] \\
&\qquad = \min_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi(y|x)}\left[\log\frac{\pi(y|x)}{\pi_{\mathrm{ref}}(y|x)} - \frac{1}{\beta}r(x,y)\right] \\
&\qquad = \min_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi(y|x)}\left[\log\frac{\pi(y|x)}{\frac{1}{Z(x)}\pi_{\mathrm{ref}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)} - \log Z(x)\right]
\end{aligned}$$

Z(x)是一个归一化参数，确保为 π 概率分布, $\pi(y|x)$ 所有可能输出y的概率之和为1。通过乘以 $exp(\frac{1}{\beta}r(x,y))$, 将奖励函数r转换为概率的权重，再通过归一化调整这些权重，使得高奖励的输出在

新的策略 $\pi(y|x)$ 中获得更高的概率。

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

通过改变变量将奖励优化问题直接转化为策略优化问题，将奖励函数r与策略 π 之间建立一个显式的函数关系。由公式3的改写可知，当满足 θ 取以下值时，公式3取最大值：

$$\pi_r(y \mid x) = \frac{1}{Z(x)} \pi_{\text{ref}}(y \mid x) \exp\left(\frac{1}{\beta} r(x,y)\right), \tag{4}$$

将公式4中奖励函数提取出来，可以表示为

$$r(x,y) = \beta \log \frac{\pi_r(y \mid x)}{\pi_{\text{ref}}(y \mid x)} + \beta \log Z(x). \tag{5}$$

将其带入到Bradley-Terry模型中，可以得到用最优策略 π^* 和参考策略 π_{ref} 表示的人类偏好：

$$p^*(y_1 \succ y_2 \mid x) = \frac{1}{1 + \exp\left(\beta \log \frac{\pi^*(y_2|x)}{\pi_{\text{ref}}(y_2|x)} - \beta \log \frac{\pi^*(y_1|x)}{\pi_{\text{ref}}(y_1|x)}\right)} \tag{6}$$

与公式2类似，就可以用最大似然来估计 π_θ：

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} \left[\log \sigma \left(\beta \log \frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)} - \beta \log \frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\right)\right]. \tag{7}$$

这样就替换掉了奖励模型，直接对策略模型进行优化。

对 θ 求梯度得到：

$$\nabla_\theta \mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = \\ -\beta \mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} \Bigg[\underbrace{\sigma(\hat{r}_\theta(x, y_l) - \hat{r}_\theta(x, y_w))}_{\text{higher weight when reward estimate is wrong}} \Bigg[\underbrace{\nabla_\theta \log \pi(y_w \mid x)}_{\text{increase likelihood of } y_w} - \underbrace{\nabla_\theta \log \pi(y_l \mid x)}_{\text{decrease likelihood of } y_l} \Bigg] \Bigg]$$

$$\hat{r}_\theta(x,y) = \beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$$

这是原论文中的推导，本质就是求导的链式法则，不过红色部分的w和l应该是标反了，

In this section we derive the gradient of the DPO objective:

$$\nabla_\theta \mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\nabla_\theta \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log \sigma\left(\boxed{\beta \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)} - \beta \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)}}\right)\right] \tag{21}$$

We can rewrite the RHS of Equation 21 as

$$\nabla_\theta \mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\frac{\sigma'(u)}{\sigma(u)}\nabla_\theta(u)\right], \tag{22}$$

where $u = \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)} - \beta \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)}$.

Using the properties of sigmoid function $\sigma'(x) = \sigma(x)(1-\sigma(x))$ and $\sigma(-x) = 1-\sigma(x)$, we obtain the final gradient

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

After using the reward substitution of $\hat{r}_\theta(x,y) = \beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$ we obtain the final form of the gradient from Section 4.

训练流程：

1）对每个prompt x生成 $(y1, y2) \sim \pi_{ref}(y|x)$，并进行人类偏好标注，生成收集数据集 $\mathcal{D} = \{x^i, y_w^i, y_l^i\}_{i=1}^N$

2）给定 β 和参考模型 π_{ref}，最小化 $\mathcal{L}_{DPO}$ 以优化生成模型(其实就是策略模型) π_θ。

现实应用中可以采用公开数据集，当 π^{SFT} 不可用时，可以用prefer行为（x, y_w）的最大似然来初始化 π_{ref}，即

$$\pi_{\text{ref}} = \arg\max_\pi \mathbb{E}_{x,y_w\sim\mathcal{D}}\left[\log \pi(y_w \mid x)\right]$$

实验结果：

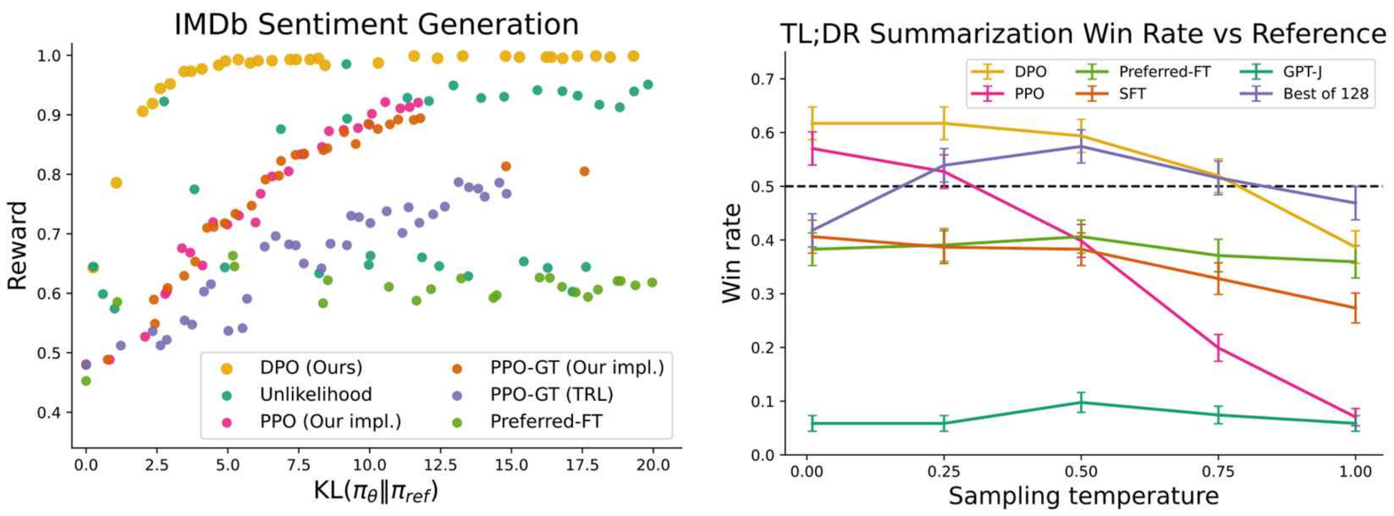
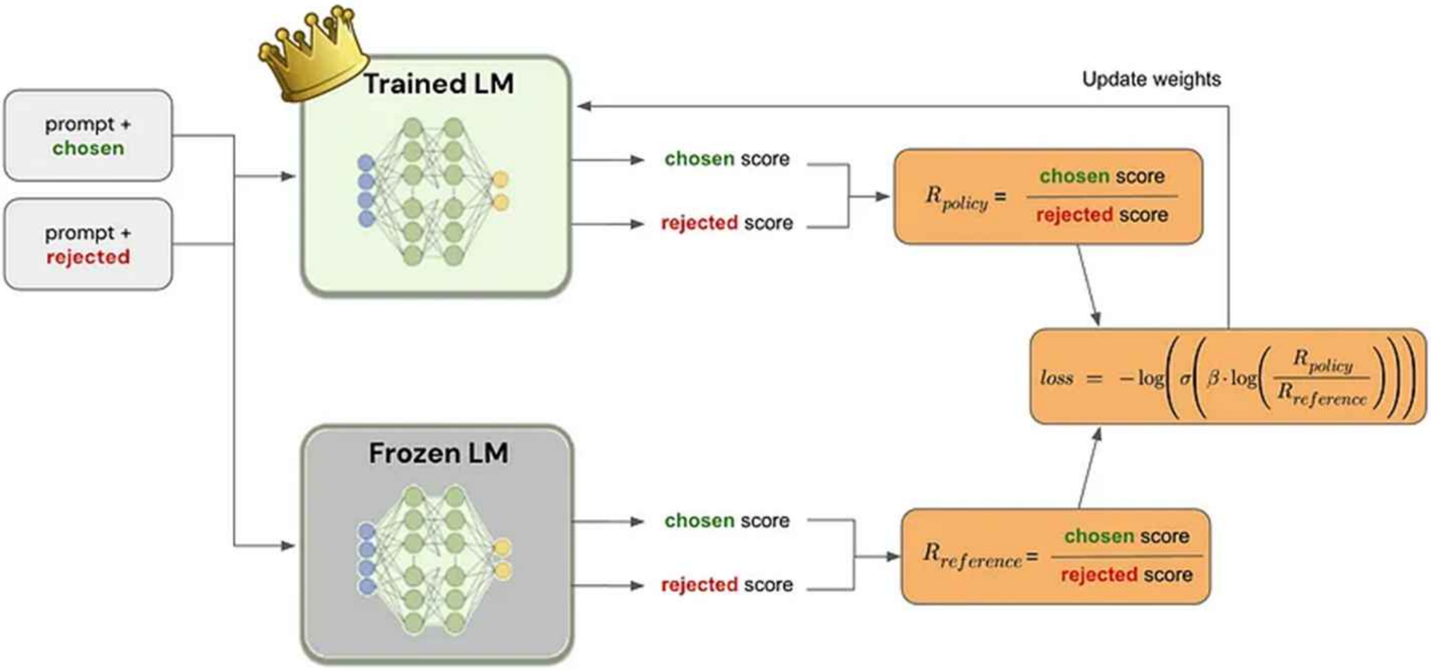


Figure 2: **Left.** The frontier of expected reward vs KL to the reference policy. DPO provides the highest expected reward for all KL values, demonstrating the quality of the optimization. **Right.** TL;DR summarization win rates vs. human-written summaries, using GPT-4 as evaluator. DPO exceeds PPO's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

3. DPO的改进



在训练过程中，始终要保留训练模型和参照模型。训练过程中去掉参照模型，例如ORPO，ORPO观察到生成了非常多的disprefer的样本，其直接对序列的生成概率进行优化（DPO中还是在对奖励函数进行优化，只不过是通过最大似然替换掉了奖励模型，通过最大化奖励值使得生成prefer的样本的概率增大），优化目标就是最大化生成prefer概率。考虑到存在很多没有标注偏好的SFT生成的数据，为了利用这些数据加上 L_{SFT} 。

Define odd: $$\mathbf{odds}_\theta(y|x) = \frac{P_\theta(y|x)}{1-P_\theta(y|x)}$$

Define ratio between odd: $$\mathbf{OR}_\theta(y_w, y_l) = \frac{\mathbf{odds}_\theta(y_w|x)}{\mathbf{odds}_\theta(y_l|x)}$$

Define loss: $$\mathcal{L}_{ORPO} = \mathbb{E}_{(x,y_w,y_l)}\left[\mathcal{L}_{SFT} + \lambda\cdot\mathcal{L}_{OR}\right] \qquad \mathcal{L}_{OR} = -\log\sigma\left(\log\frac{\mathbf{odds}_\theta(y_w|x)}{\mathbf{odds}_\theta(y_l|x)}\right)$$

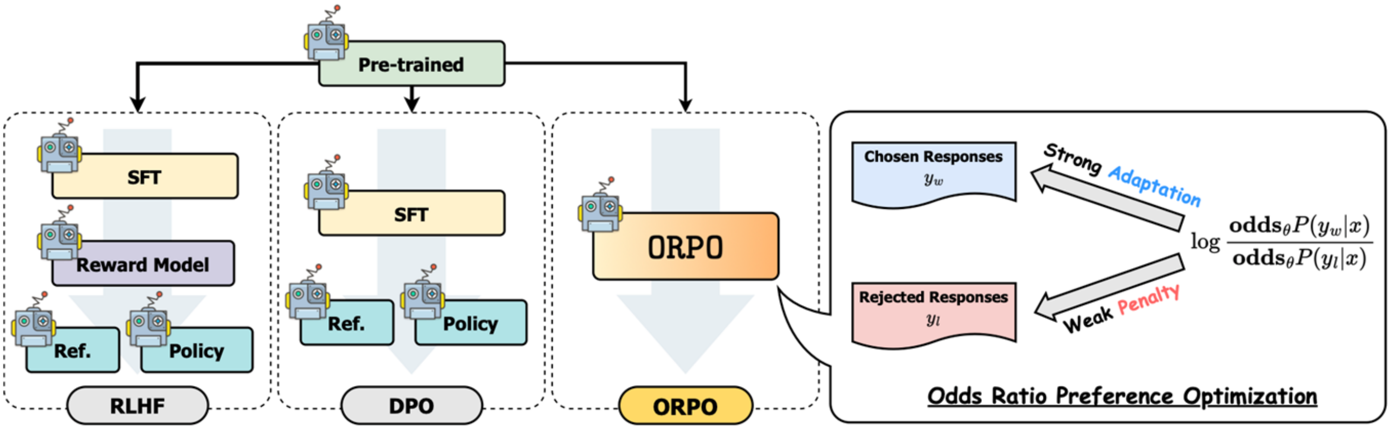


Figure 2: Comparison of model alignment techniques. ORPO aligns the language model *without a reference model* in a single-step manner by assigning a weak penalty to the rejected responses and a strong adaptation signal to the chosen responses with a simple log odds ratio term appended to the negative log-likelihood loss.