Uncertainty-Penalized Direct Preference Optimization

Sam Houliston[*,1] **Alizée Pace**[1,2,3] **Alexander Immer**[1,3] **Gunnar Rätsch**[1,2]

[1]Department of Computer Science, ETH Zurich, Switzerland
[2]ETH AI Center, Zurich, Switzerland
[3]Max Planck Institute for Intelligent Systems, Tübingen, Germany

Abstract

Aligning Large Language Models (LLMs) to human preferences in content, style, and presentation is challenging, in part because preferences are varied, context-dependent, and sometimes inherently ambiguous. While successful, Reinforcement Learning from Human Feedback (RLHF) and Direct Preference Optimization (DPO) are prone to the issue of proxy reward overoptimization. Analysis of the DPO loss reveals a critical need for regularization for mislabeled or ambiguous preference pairs to avoid reward hacking. In this work, we develop a pessimistic framework for DPO by introducing preference uncertainty penalization schemes, inspired by offline reinforcement learning. The penalization serves as a correction to the loss which attenuates the loss gradient for uncertain samples. Evaluation of the methods is performed with GPT2 Medium on the Anthropic-HH dataset using a model ensemble to obtain uncertainty estimates, and shows improved overall performance compared to vanilla DPO, as well as better completions on prompts from high-uncertainty chosen/rejected responses.

1 Introduction

Aligning LLMs to human preferences in content, style, and presentation has become a central challenge in improving and deploying LLMs, leading to the advent of fine-tuning by Reinforcement Learning with Human Feedback (RLHF) [Casper et al., 2023]. Direct Preference Optimisation (DPO) [Rafailov et al., 2023] is an effective a reward-model-free alternative to standard RLHF which maximizes the likelihood of the preference pairs under the Bradley–Terry model [A. and Terry, 1952]. DPO is simple to implement and avoids inheriting inaccuracy or instability from a reward model [Xu et al., 2024, Casper et al., 2023].

Both RLHF and DPO suffer from proxy reward overoptimization, in part due to an imperfect coverage of the full preference distribution by the training data. DPO tends to overfit and treat all preference pairs equally, even when some pairs are stronger [Azar et al., 2023, Amini et al., 2024]. We analyze the DPO loss in Appendix B, showing DPO's sensitivity to erroneous preferences. Similar problems arise in Offline RL, where distributional shifts cause reward overestimation [Jin et al., 2020, Li et al., 2021]. Introducing pessimism by penalizing high uncertainty rewards addresses this and improves policy generalization [Jin et al., 2020].

In this work, we introduce uncertainty penalization schemes for DPO, inspired by pessimistic offline RL. Our best-performing approach multiplies implicit rewards with an energy-based uncertainty function (derived in Section 3). This framework assumes preference uncertainty estimates, obtained during data collection or from a reward model with uncertainty quantification.

[*]Correspondence to: shouliston@student.ethz.ch.

Accepted to the Adaptive Foundation Models (AFM) Workshop at NeurIPS 2024.

Our main contributions are: (1) We analyze DPO's overfitting issue, highlighting its sensitivity to mislabeled samples, complementing the literature [Azar et al., 2023, Pal et al., 2024, Amini et al., 2024, Morimura et al., 2024]. (2) We propose a pessimistic framework using uncertainty penalization to mitigate overfitting, generalizable to RLHF and DPO variants like IPO [Azar et al., 2023]. (3) We show our methods vanilla DPO across various tasks and robustness experiments.

2 Related Work

Uncertainty Penalization in Standard RLHF. Zhai et al. [2023a] show empirically that Kullback-Leibler (KL) regularization in the standard RLHF pipeline may be insufficient to avoid reward overoptimization. Their method entitled Uncertainty Penalized-RLHF (UP-RLHF) uses conservative reward estimates by subtracting the reward model ensemble epistemic uncertainty. Similarly, Yang et al. [2024a] extend the best-of-n sampling framework with pessimistic reward scores by subtracting a factor of the predicted reward variance from a bayesian reward model.

DPO and Variants. DPO [Rafailov et al., 2023] is an effective fine-tuning technique performed on binary preferences without an external reward model. DPO however, tends to overfit on its training data [Azar et al., 2023]. Identity Preference Optimization (IPO) [Azar et al., 2023] addresses this by adding regularization to DPO, eliminating the need for early stopping. It has been unified with RLHF and DPO under a general framework ΨPO. Pal et al. [2024] highlight DPO's failure on low-edit-distance preference pairs, where DPO-Positive (DPO-P) adds loss clipping to maintain positive log-likelihood for chosen text. Offset DPO (ODPO) [Amini et al., 2024] introduces a margin based on external reward model scores to differentiate strong and weak preferences. Our work is similar to ODPO. We show standard Lower Confidence Bound penalization recovers a margin-based DPO loss (3) similar to ODPO. Our main method penalizes uncertainty via multiplication which prevents uncertainties from cancelling out, leading to improved results. More related works are described in greater detail in Appendix B.3.

Method	Uncertainty Quantification Method	Penalization Scheme	Alignment Algorithm
UP-RLHF [Zhai et al., 2023b]	Diverse LoRA Ensemble	$r(x,y) \leftarrow r(x,y) - u(y\|x)$	PPO
[Yang et al., 2024b]	Bayesian LoRA Model	$r(x,y) \leftarrow r(x,y) - u(y\|x)$	Best of N
Active-DPO [Muldrew et al., 2024]	LLM policy entropy and Implicit rewards margin	Bayesian Optimization to select training samples	DPO
Uncertainty Energy Factor (Ours)	LoRA Ensemble	$\hat{r}_\theta(x,y) \leftarrow \hat{r}_\theta(x,y)\, e^{-u(y\|x)/\tau}$	DPO, IPO

Table 1: Related methods in RLHF that leverage uncertainty estimation.

3 Method

The central contribution of this paper is to propose a penalization scheme appropriate for DPO, which leverages known uncertainty estimates on preferences. Our approach applies a reward uncertainty penalization to the overarching RLHF objective, from which we derive a new penalized DPO loss. For starters, we introduce the standard Lower Common Bound penalization [Jin et al., 2020] to DPO. Our main method termed "Energy Factor Penalization" is a multiplicative penalization that brings considerable benefits to the binary chosen-rejected nature of DPO.

The framework setting and notation build on that of DPO [Rafailov et al., 2023]. The aim is to align the parametrized LLM policy π_θ, to the preference dataset $\mathcal{D} = \{x_i, y_{i,w}, y_{i,l}\}_{i=1}^N$ composed of prompts x, chosen completions y_w, and rejected completions y_l, while keeping close to a given reference model policy π_{ref}. In addition, we assume access to a reward model $r(x,y)$ equipped with uncertainty quantification $u(y|x)$.

3.1 Importing Standard Uncertainty Penalization to DPO

Pessimistic RL subtracts a factor of the reward uncertainty $u(y|x)$ from the reward score $r(x,y)$ to obtain a conservative estimate of the reward function as a Lower Confidence Bound. Applying this to the general RLHF objective results in Equation (1).

$$\max_{\pi_\theta} \mathop{\mathbb{E}}_{\substack{x\sim D \\ y\sim\pi_\theta(y|x)}} [r(x,y) - u(y|x)] - \beta\, D_{\text{KL}}\left(\pi_\theta(y|x) \,||\, \pi_{\text{ref}}(y|x)\right). \tag{1}$$

Following prior work [Peters and Schaal, 2007, Go et al., 2022, Rafailov et al., 2023] the unique solution Equation (1) is derived. The optimal policy π_u^* corresponds to the reference policy being modulated by the conservative reward estimate, with $Z_u(x)$ as the appropriate partition function. Indeed, a high reward uncertainty $u(y|x)$ for a given completion y will induce a lower policy probability:

$$\pi_u^*(y|x) = \frac{1}{Z_u(x)} \pi_{\text{ref}}(y|x)\, e^{\frac{1}{\beta}(r(x,y) - u(y|x))}. \tag{2}$$

Following the original DPO derivation (Appendix A), Equation (2) is injected in the Bradley-Terry model, the optimal policy is replaced by the parameterized policy π_θ which is then optimized by maximum likelihood under the Bradley-Terry model, giving the following loss:

$$\mathcal{L}_{\text{DPO}}^u(\pi_\theta; \pi_{\text{ref}}) = \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \left[-\log \sigma \left(\underbrace{\hat{r}_\theta(x, y_w) - \hat{r}_\theta(x, y_l)}_{\rho_\theta} + \underbrace{u(y_w|x) - u(y_l|x)}_{\Delta_u} \right) \right] \tag{3}$$

$$\nabla_\theta \mathcal{L}_{\text{DPO}}^u = \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \left[-\beta \sigma \left(-\rho_\theta - \Delta_u \right) \nabla_\theta \log \frac{\pi_\theta(y_w|x)}{\pi_\theta(y_l|x)} \right]. \tag{4}$$

Effect of the penalization. The pessimistic correction amounts to adding the margin $\Delta_u = u(y_w|x) - u(y_l|x)$ between implicit rewards. Following the analysis of the loss, Equation (4) suggests a higher positive margin will decrease the gradient magnitude. This modification is pessimistic: i) A high chosen reward uncertainty $u(y_w|x)$ will reduce the gradient, attenuating the gradient update, thus $\pi_\theta(y_w|x)$ will not increase much. ii) A high rejected reward uncertainty $u(y_l|x)$ will enhance the update: $\pi_\theta(y_l|x)$ will additionally decrease.

The chosen and rejected uncertainties in this scheme individually exhibit desirable effects, however they are not independent: they may cancel out or interfere.

Connection to Offset-DPO. The derivation above recovers a DPO loss Equation (4) with an additional margin Δ_u. This loss has the same form as Offset-DPO Amini et al. [2024] which considers the offset Δ_u as an increasing function f of the difference between reward model scores $\Delta = f(r(x, y_w) - r(x, y_l))$. They link this penalization to the Softmax-Margin loss, which we study in Appendix C.3 and derive a fully additive scheme $\Delta_u = u(y_w|x) + u(y_l|x)$ in Equation (37) which sums uncertainties, preventing their cancellation.

3.2 Main Method: Energy Factor Penalization

The previous section motivates a multiplicative penalization scheme (instead of subtraction) to ensure the penalization effect of either chosen or rejected uncertainties carries to the respective chosen or rejected policy gradient update terms in Equation (4). Our proposed scheme multiplies the preference value or reward by an energy-like function of the uncertainty. Such penalization can be modulated by a temperature parameter $\tau > 0$:

$$\max_{\pi_\theta} \mathop{\mathbb{E}}_{\substack{x\sim D \\ y\sim\pi_\theta(y|x)}} \left[r(x, y) e^{-u(y|x)/\tau} \right] - \beta\, D_{\text{KL}}\left(\pi_\theta(y|x) \,||\, \pi_{\text{ref}}(y|x)\right). \tag{5}$$

This objective is derived into a DPO loss following the same steps as above, to obtain the expression (6). The full derivation is found in Appendix C.2.

$$\mathcal{L}_{\text{DPO}}^u = \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \left[-\log \sigma \underbrace{\left\{ e^{u(y_w|x)/\tau} \hat{r}_\theta(x, y_w) - e^{u(y_l|x)/\tau} \hat{r}_\theta(x, y_l) \right\}}_{\tilde{\rho}_\theta} \right] \tag{6}$$

$$\nabla \mathcal{L}_{\text{DPO}}^u = \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \left[-\beta \sigma \left(-\tilde{\rho}_\theta \right) \left(e^{u(y_w|x)/\tau} \frac{\nabla_\theta \pi_\theta(y_w|x)}{\pi_\theta(y_w|x)} - e^{u(y_l|x)/\tau} \frac{\nabla_\theta \pi_\theta(y_l|x)}{\pi_\theta(y_l|x)} \right) \right]. \tag{7}$$

Effect of the penalization. The pessimistic correction inherits similar features from the LCB scheme. However, there is an additional effect, the individual uncertainties carry onto their respective terms in the gradient of Equation (7) instead of only affecting the overall gradient magnitude. This means that uncertainties will not cancel out if they are commensurate, unlike in Equation (4).

Practical Implication: Scaling of the Penalty. The uncertainty penalties are obtained from an external reward model, preference dataset statistics or even additional user labels. These must be scaled appropriately to DPO's implicit rewards $\hat{r}_\theta$. Practically, we propose to use a running average estimate across batches for the mean implicit reward value and mean uncertainty to compute a scalar multiplier α to the penalty: $\Delta_u \leftarrow \alpha\Delta_u$. More details are found in Appendix C.6.

3.3 Generalization to ΨPO and IPO

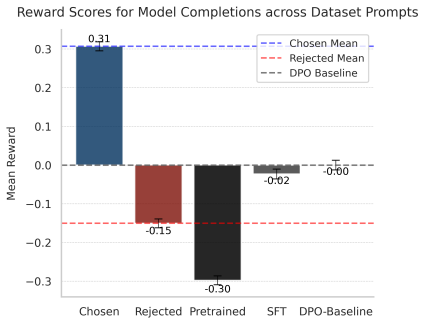
The ΨPO framework by Azar et al. [2023] is a general objective that encompasses many RLHF methods, including the standard RLHF objective (8), DPO, and Identity Preference Optimization (IPO). We generalize our penalization schemes to ΨPO in Appendix C.5, and import our schemes to IPO. A summary of our proposed penalization schemes can be found in Appendix C.4.

4 Experiments

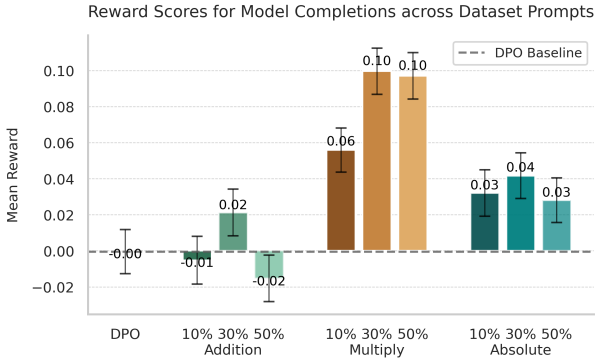
An ensemble of reward models is trained to compute uncertainty estimates for the Anthropic-HH dataset. All LLMs first undergo supervised fine-tuning (SFT) on the chosen completions of the dataset, and then preference fine-tuning is performed. The evaluation is done by scoring model completions for prompts from the Anthropic-HH test set. The experiments are performed on GPT2 Medium (355M weights, pretrained); complete implementation details and hyperparameter search ranges are found in Appendix D.[2]

4.1 Performance Evaluation

The penalized DPO models perform on par or better than vanilla DPO. Figure 1b shows the addition scheme performs similarly to DPO whereas our multiplication scheme outclasses the baseline for all penalization strengths. For each of the schemes, the middle penalization strength of 30% performs best. Scores and uncertainties across all models and chosen/rejected baselines, are presented in table 2, showing our schemes outperform the SFT and DPO baselines.



(a) Experimental Setup. The increase in scores from pretrained to DPO validates the training setup.



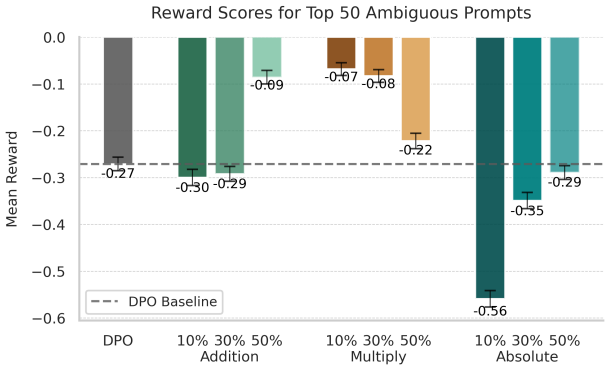
(b) Evaluation of Finetunings. Pessimistic schemes perform on-par or better than DPO, best scores achieved at 30% penalty.

Figure 1: Rewards over completions for 500 Anthropic-HH test prompts.

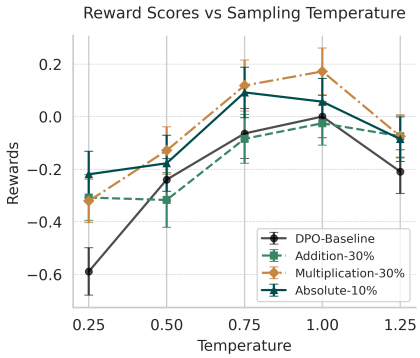
4.2 Robustness Evaluation

Performance on Uncertain Samples. Our method should improve training on chosen and rejected pairs exhibiting high reward uncertainty. To evaluate this, we isolate the 50 test records whose chosen and rejected responses have the highest summed reward uncertainty. Figure 2a depicts reward scores for tuned model completions on those 50 prompts and shows superior performance of the multiplication scheme. Addition performs similarly to the DPO baseline while absolute scheme performs worse. For both addition and absolute, performance increases with penalty strength.

[2]The code is available at: https://anonymous.4open.science/r/PessimisticDPO-DAF4/



(a) Evaluation on 50 prompts with highest chosen/rejected text uncertainty. Multiplication penalty perform best. Scores improve with penalization strength for Addition and Absolute.



(b) Mean reward for 200 completions per temperature. Vanilla DPO rewards drop the most as temperature varies.

Figure 2: Study of robustness to reward overoptimization.

Study of Sampling Temperature. Figure 2b displays mean reward scores for generated completions at different sampling temperatures from the top-performing models of each scheme. The multiplication and absolute schemes perform better than DPO across all temperatures. All curves are bell-shaped: low temperatures favour high-probability, repetitive text, and high temperatures invite randomness in next-token sampling leading to to more creative text at the risk of incoherence or quality. For well-behaved policies we expect the scores to be more level at various temperatures. The penalized models indeed have a lower drop-off in performance across the temperature range.

Model	**All Prompts** Mean reward	**Top 50 Ambiguous** Mean reward
Chosen	0.306 ± 0.013	-0.350 ± 0.018
Rejected	-0.151 ± 0.013	-0.424 ± 0.017
Pretrained	-0.298 ± 0.009	-0.524 ± 0.011
SFT	-0.026 ± 0.011	-0.253 ± 0.013
DPO	-0.001 ± 0.011	-0.271 ± 0.012
Addition (10%)	-0.005 ± 0.012	-0.300 ± 0.013
Addition (30%)	0.021 ± 0.011	-0.292 ± 0.014
Addition (50%)	-0.015 ± 0.011	$\mathbf{-0.086 \pm 0.012}$
Multiplication (10%)	$\mathbf{0.056 \pm 0.011}$	$\mathbf{-0.069 \pm 0.014}$
Multiplication (30%)	$\mathbf{0.099 \pm 0.011}$	$\mathbf{-0.083 \pm 0.012}$
Multiplication (50%)	$\mathbf{0.097 \pm 0.011}$	-0.222 ± 0.014
Absolute (10%)	0.032 ± 0.011	-0.559 ± 0.013
Absolute (30%)	0.042 ± 0.011	-0.349 ± 0.015
Absolute (50%)	0.028 ± 0.011	-0.289 ± 0.014

Table 2: Reward for Model Completions on 500 Anthropic-HH Test Set prompts. Values are presented as the mean $\pm$ standard error. Top three scores are highlighted in **gold**, **silver** and **bronze**.

Discussion. The reward model ensemble provides a reasonable assessment of completions relative to chosen and rejected outputs, as it obtained a 67% test accuracy. However, its reward margin is small (Figures 1a, 1) and may overfit to Anthropic-style responses, repercussing to inaccurate rewards for different but still preferable text; this may be addressed by a more powerful model.

Example completions in Appendix D.7 show the fine-tuned models exhibit agreeableness and coherence. However, training the DPO Baseline to a satisfactory performance (Figure 1a) required extensive hyperparameter tuning. This may be due to the hardness of learning preference associations in Anthropic-HH relative to GPT2 Medium's size. Despite this, the results are promising for larger models and diverse tasks.

5 Conclusion

This work proposes a new framework for DPO inspired by pessimistic offline RL that integrates pessimism into DPO by leveraging preference uncertainty estimates. The derived penalization schemes are tailored to the binary nature of DPO, with our Energy Factor scheme performing best overall and robustness-wise in our illustrative experiments. The empirical findings invite further evaluation with more powerful models on various tasks (summarization, dialogue, completion...). Finally, a generalization to ΨPO, IPO, and a reward-model-free scheme are proposed; further work is invited to develop and implement these.

References

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Wang, Samuel Marks, Charbel-Raphaël Segerie, Micah Carroll, Andi Peng, Phillip Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J. Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco, Peter Hase, Erdem Bıyık, Anca Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. Open problems and fundamental limitations of reinforcement learning from human feedback, 2023.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Bradley R. A. and M. E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika, 39(3/4):324–345*, 1952.

Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. Is dpo superior to ppo for llm alignment? a comprehensive study, 2024.

Mohammad Gheshlaghi Azar, Mark Rowland, Bilal Piot, Daniel Guo, Daniele Calandriello, Michal Valko, and Rémi Munos. A general theoretical paradigm to understand learning from human preferences, 2023.

Afra Amini, Tim Vieira, and Ryan Cotterell. Direct preference optimization with an offset, 2024.

Ying Jin, Zhuoran Yang, and Zhaoran Wang. Is pessimism provably efficient for offline rl? *CoRR*, abs/2012.15085, 2020.

Jinning Li, Chen Tang, Masayoshi Tomizuka, and Wei Zhan. Dealing with the unknown: Pessimistic offline reinforcement learning. In *5th Annual Conference on Robot Learning*, 2021.

Arka Pal, Deep Karkhanis, Samuel Dooley, Manley Roberts, Siddartha Naidu, and Colin White. Smaug: Fixing failure modes of preference optimisation with dpo-positive, 2024.

Tetsuro Morimura, Mitsuki Sakamoto, Yuu Jinnai, Kenshi Abe, and Kaito Ariu. Filtered direct preference optimization, 2024.

Yuanzhao Zhai, Han Zhang, Yu Lei, Yue Yu, Kele Xu, Dawei Feng, Bo Ding, and Huaimin Wang. Uncertainty-penalized reinforcement learning from human feedback with diverse reward lora ensembles, 2023a.

Adam X. Yang, Maxime Robeyns, Thomas Coste, Jun Wang, Haitham Bou-Ammar, and Laurence Aitchison. Bayesian reward models for llm alignment, 2024a.

Yuanzhao Zhai, Han Zhang, Yu Lei, Yue Yu, Kele Xu, Dawei Feng, Bo Ding, and Huaimin Wang. Uncertainty-penalized reinforcement learning from human feedback with diverse reward lora ensembles, 2023b.

Adam X. Yang, Maxime Robeyns, Thomas Coste, Jun Wang, Haitham Bou-Ammar, and Laurence Aitchison. Bayesian reward models for llm alignment, 2024b.

William Muldrew, Peter Hayes, Mingtian Zhang, and David Barber. Active preference learning for large language models, 2024.

Jan Peters and Stefan Schaal. Reinforcement learning by reward-weighted regression for operational space control. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

Dongyoung Go, Tomasz Korbak, Germán Kruszewski, Jos Rozen, Nahyeon Ryu, and Marc Dymetman. Aligning language models with preferences through f-divergence minimization, 2022.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know, 2022a.

Kevin Gimpel and Noah A. Smith. Softmax-margin CRFs: Training log-linear models with cost functions. In Ron Kaplan, Jill Burstein, Mary Harper, and Gerald Penn, editors, *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, June 2010.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know, 2022b.

Yuesong Shen, Nico Daheim, Bai Cong, Peter Nickl, Gian Maria Marconi, Clement Bazan, Rio Yokota, Iryna Gurevych, Daniel Cremers, Mohammad Emtiyaz Khan, and Thomas Möllenhoff. Variational learning is effective for large deep networks, 2024.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022.

Appendix

A DPO Background and Derivation

Problem setup. We have a static dataset of comparisons denoted as $\mathcal{D} = \{x, y_w, y_l\}_{i=1}^{N}$, which is usually obtained by prompting an SFT (supervised-fine-tuned) model with prompts x to produce pairs of answers (y_w, y_l) (these comparisons do not have scores, they are absolute). Human labellers identify the preferred output y_w over the undesired y_l. We define a reference (SFT) model policy as π_{ref} and our parameterized policy as π_θ, which we aim to fit as to make the preferred outputs y_w more likely, while staying close to the reference policy.

A.1 DPO Derivation from RLHF.

The RLHF objective is:

$$\max_{\pi_\theta} \mathop{\mathbb{E}}_{\substack{x\sim D \\ y\sim\pi_\theta(y|x)}} [r_\phi(x, y)] - \beta \,\text{KL}\left(\pi_\theta(y|x) \,||\, \pi_{\text{ref}}(y|x)\right). \tag{8}$$

By factoring the terms under the expectation, we obtain an KL divergence-like expression, we introduce the partition function[3] $Z(x) = \sum_y \pi_{\text{ref}}(y|x) \exp\left(\frac{1}{\beta} r(x, y)\right)$ to normalize the denominator, and the optimal value annuls this KL-divergence, giving us the unique optimal solution:

$$\pi^*(y|x) = \frac{1}{Z(x)} \pi_{\text{ref}}(y|x)\, e^{\frac{1}{\beta} r(x,y)}. \tag{9}$$

Intuitively, our optimal policy aligns with the reference policy, modulated by high or low rewards of certain outputs. Now, equation 9 can be re-arranged to express the ground-truth reward model in function of the induced optimal and reference policies: $r(x, y) = \beta \log\left(\frac{\pi^*(y|x)}{\pi_{\text{ref}}(y|x)}\right) + \beta \log Z(x)$.

The authors, define the $\hat{r}_\theta(x, y) = \beta \log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right) + \beta \log Z(x) \approx \beta \log\left(\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right)$, as the reward implicitly defined by the language model π_θ.

Enter, the Bradley-Terry model: $p(y_1 \succ y_2|x) = \sigma(r(x, y_1) - r(x, y_2))$, interpreted as the probability of answer y_1 being favoured over y_2 as a function of their human-labelled rewards. In RLHF the reward model is trained to maximize $p(y_w, y_l)$ over a dataset. For DPO, we do the same, by substituting the parameterised reward expressions:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = - \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} [\log p(y_w \succ y_l)] \tag{10}$$

$$= - \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \left[\log \sigma \left(\underbrace{\hat{r}_\theta(x, y_w) - \hat{r}_\theta(x, y_l)}_{\rho_\theta}\right)\right] \tag{11}$$

$$= - \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \left[\log \sigma \left(\beta \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}\right)\right]. \tag{12}$$

A.2 DPO as a Binary Classification Problem.

Another view interprets the DPO loss 19 as akin to binary classification, where the given y_1 is preferable to y_2, and we aim to train the parametrized policy π_θ such that the preference model $p_\theta(y_1 \succ y_2|x) = \sigma(\hat{r}_\theta(x, y_1) - \hat{r}_\theta(x, y_2))$ predicts 1, meaning we aim to maximize the quantity ρ_θ (see GPO paper for proper derivation), which effectively increases the margin between the probability of the preferred and unpreferred sample. Note: sometimes in DPO optimization, both probabilities are decreased, but the unpreferred sample probability is decreased more strongly; while this improves performance on the training preference dataset this might have the adverse effect of increasing the probability of other output text sequences.

[3] Some works show this function is very close to 1.

A.3 DPO Loss Gradient Derivation

We provide the full derivation for the loss gradient w.r.t. policy parameters. Recall the properties of the sigmoid $\sigma'(x) = \sigma(x)(1-\sigma(x))$, $\sigma(-x) = 1-\sigma(x)$, with $\nabla \log(x) = \frac{\nabla x}{x}$.

$$\nabla_\theta \, \ell_{\text{DPO}}(x, y_w, y_l; \theta) = -\nabla_\theta \log \sigma\left(\rho_\theta\right) \tag{13}$$

$$= -\frac{\nabla_\theta \sigma\left(\rho_\theta\right)}{\sigma\left(\rho_\theta\right)} \tag{14}$$

$$= -\frac{\sigma(\rho_\theta)(1-\sigma(\rho_\theta))\,\nabla_\theta \rho_\theta}{\sigma(\rho_\theta)} \tag{15}$$

$$= -\sigma(-\rho_\theta)\,\nabla_\theta \rho_\theta \tag{16}$$

$$= -\beta\sigma(-\rho_\theta)\left(\frac{\nabla_\theta \pi_\theta(y_w|x)}{\pi_\theta(y_w|x)} - \frac{\nabla_\theta \pi_\theta(y_w|x)}{\pi_\theta(y_w|x)}\right) \tag{17}$$

$$= -\beta\sigma(-\rho_\theta)\left(\frac{\nabla_\theta \pi_\theta(y_w|x)}{\pi_\theta(y_w|x)} - \frac{\nabla_\theta \pi_\theta(y_l|x)}{\pi_\theta(y_l|x)}\right) \tag{18}$$

B Analysis of the DPO Loss

Analysis of the DPO loss shows larger policy gradient update steps are applied on preference pairs with low chosen/rejected likelihood ratios compared to reference ratios 21. This behavior can be harmful for mislabeled or similar pairs. Additionally, the overoptimization phenomenon is shown to happen for rejected samples with low probability ($\pi_\theta(y_l|x) \ll 1$), where DPO does not regularize and decreases $\pi_\theta(y_l|x)$ further, potentially increasing the relative probability of other completions. These observations, tied to the overoptimization problem addressed by IPO [Azar et al., 2023], motivate the incorporation of pessimism as a gradient update attenuation mechanism for erroneous, or similar (uncertain) samples.

Problem Setting. The aim is to align the parametrized LLM policy π_θ, to the preference dataset $\mathcal{D} = \{x_i, y_{i,w}, y_{i,l}\}_{i=1}^N$ composed of prompts x, chosen completions y_w, and rejected completions y_l, while keeping close to a given reference model policy π_{ref}. The DPO loss [Rafailov et al., 2023] is formulated as:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D}\left[-\log\sigma\left(\beta\log\frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta\log\frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}\right)\right] \tag{19}$$

$$= \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D}\left[-\log\sigma\Bigg(\beta\log\underbrace{\left(\frac{\pi_\theta(y_w|x)\pi_{\text{ref}}(y_l|x)}{\pi_\theta(y_l|x)\,\pi_{\text{ref}}(y_w|x)}\right)}_{A_\theta}\Bigg)\right]. \tag{20}$$

By the monotonicity of the log and sigmoid functions, minimizing the DPO loss in Equation (19) corresponds to maximizing A_θ. A_θ is proportional to the policy's chosen-rejected likelihood ratio and provides a measure of how well the policy distinguishes between the completions compared to the reference. Figure 3 shows the loss value and its gradient both increase as A_θ decreases; thus, the DPO loss severely penalizes (and strongly updates on) inputs where A_θ approaches zero.

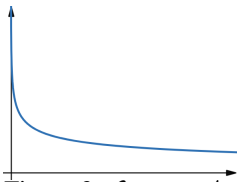


Figure 3: $\mathcal{L}_{\text{DPO}}$ vs A_θ

The strong update regime is described in Equation (21) and confirms DPO performs stronger updates when the target policy performs worse relative to the reference policy.

$$A_\theta = \frac{\pi_\theta(y_w|x)}{\pi_\theta(y_l|x)}\frac{\pi_{\text{ref}}(y_l|x)}{\pi_{\text{ref}}(y_w|x)} \ll 1 \iff \frac{\pi_\theta(y_w|x)}{\pi_\theta(y_l|x)} \ll \frac{\pi_{\text{ref}}(y_w|x)}{\pi_{\text{ref}}(y_l|x)}. \tag{21}$$

Impact of Ill-Labeled Preference Pairs. Suppose $\tilde{y}_w, \tilde{y}_l$ represent a corrupted or ill-labeled preference pair i.e. $p^*(\tilde{y}_w \succ \tilde{y}_l) \leq 0.5$ as per the Bradley-Terry model p^*. Assuming a decent reference model, we can expected the DPO-trained policy π_θ to perform worse than the reference, and to satisfy the rightmost inequality of Equation (21). Furthermore, suppose the target π_θ is overfit during training, and that preference pair $\tilde{y}_w, \tilde{y}_l$ is inherently ambiguous ($p^*(\tilde{y}_w \succ \tilde{y}_l) \approx 0.5$), it is

also likely for the target policy to perform worse than the untrained reference (Equation (21)), and find itself in the strong-update regime. Thus DPO is prone to overfitting on ill-labeled preference pairs, and once overfit, is prone to performing strong updates for inherently ambiguous preference pairs.

B.1 Analysis of the Gradient

The DPO loss gradient w.r.t. policy parameters θ is derived to have a finer understanding of the mechanics of DPO updates that follow gradient-based optimization. The full derivation is provided in Appendix A. For shorthand, we denote the variable $\rho_\theta := \hat{r}_\theta(y_w|x) - \hat{r}_\theta(y_l|x) = \beta \log A_\theta$, as the difference between implicit rewards, which increases with A_θ.

$$\nabla_\theta \mathcal{L}_{\text{DPO}}(x, y_w, y_l; \theta) = \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D}\left[-\beta \underbrace{\sigma(-\rho_\theta)}_{①} \underbrace{\left(\frac{\nabla_\theta \pi_\theta(y_w|x)}{\pi_\theta(y_w|x)} - \frac{\nabla_\theta \pi_\theta(y_l|x)}{\pi_\theta(y_l|x)}\right)}_{②}\right]. \tag{22}$$

We observe the following:

- The term ① shows the magnitude of the loss is proportional to $\sigma(-\rho_\theta)$. Thus gradient-based optimization of the DPO loss will perform stronger updates for data samples that have a low ρ_θ, i.e. a low value A_θ, which corresponds to "poor" policy performance.
- The term ② shows the policy gradient for the chosen/rejected outputs is divided by their respective policy output probability: $\nabla_\theta \pi_\theta(y_i|x)/\pi_\theta(y_i|x)$. *Thus, gradient updates for an output are enhanced if the probabilities of this output are already low.* This may be a problem for low-probability rejected completions ($\pi_\theta(y_l|x) \ll 1$) that experience continual decrease throughout training, potentially raising the relative probability of other completions. This finding ties with the empirical observation from Azar et al. [2023] that DPO performs poorly for near-deterministic preference pairs ($\pi_\theta \in \{0, 1\}$) and requires further regularization.

B.2 Conclusion: An Invitation for Pessimism

The analysis of the loss and its gradient in terms of the quantity A_θ shows DPO performs strong updates when ① the target policy exudes low A_θ i.e. a low chosen/rejected likelihood ratio compared to the reference policy; or ② when the probabilities $\pi_\theta(y_w|x), \pi_\theta(y_l|x)$ are both low. This is beneficial for well-labeled and abundant datasets, however, ill-labeled preference pairs likely correspond to i), and low-edit-distance and similar pairs may correspond to both i) and ii).

This sensitivity of DPO to ①, and the overfitting regime ② call for a mechanism to attenuate gradient updates on known weak or wrong preference pairs. If preference uncertainty scores are available, they could be leveraged as a proxy. In addition, attenuated gradient updates with a valid proxy also address an issue in DPO that all pairwise preferences are of equal weight in a dataset, despite some preference pairs being much stronger than others. Without additional attenuation, the only safety net is the quality of the reference model which regulates A_θ.

B.3 Extended Related Works for DPO

DPO [Rafailov et al., 2023] is an effective approach to finetuning for binary preferences without a reward model or reinforcement learning, while still optimizing for the original RLHF objective. The works below study its limitations and extend the method to more involved frameworks.

Azar et al. [2023] notice DPO easily overfits on training preferences, especially for inputs where the policy's implicit reward are nearly deterministic (close to 1 or 0). They introduce Identity Preference Optimisation (IPO) which adds a regularisation term to DPO, enabling one to train models to convergence without requiring tricks like early stopping. In addition, they unify RLHF, DPO, and IPO under a common mathematical formulation ΨPO.

Pal et al. [2024] frame LLM generation as a Markov Decision Process (MDP) at the token level (instead of a contextual bandit at the entire completion level) to show a failure mode of DPO on low-edit-distance preference pairs. In this case, DPO increases the relative probability between the chosen and rejected text, however is a reduction in both absolute likelihoods. DPO-Positive (DPO-P) adds a clipping term to the loss to ensure positive log-likelihood of the chosen text.

Amini et al. [2024] introduce Offstet DPO (ODPO) which adds a margin between the implicit chosen/rejected rewards in the DPO loss. The margin is based on reward scores given by an external reward model, to help DPO distinguish between strong or weak preference pairs. Our work is similar to ODPO. For the standard Lower Confidence Bound uncertainty penalization, we recover a similar loss formulation with a margin (3); our margin equals the difference in chosen-rejected uncertainties, whereas ODPO uses the difference in reward scores. For our main method *Energy Factor Penalization*, the resulting penalized loss does not have an additive margin, instead, the individual chosen-rejected implicit rewards are multiplied by an energy function of their uncertainty (6) which prevents uncertainties from canceling out, leading to a more precise penalization and improved results. More related works are described in Appendix B.3.

Filtered DPO (FDPO) [Morimura et al., 2024] uses a trained reward model to add a data refinement step to the DPO workflow: for a prompt and preference pair, the policy completion to the prompt is scored by the reward model, if that score is higher than the chosen completion's, this sample pair is discarded for its low quality. The authors assert that DPO is particularly prone to low text quality compared to reward-based RLHF methods.

Muldrew et al. [2024] develop an active learning strategy for DPO to perform tuning on the policy's own completions in an online setting, while assuming access to a preference oracle (reward model). Their iterative workflow begins by sampling prompts, generating two policy completions per prompt, scoring them via an acquisition function, shortlisting highest-scoring pairs, labeling these pairs with a reward model, and finally performing DPO on this subset. They propose a practical acquisition function for prompt/completion pairs based on the predictive entropy of the language model, shown to be well-calibrated measure of uncertainty in LLMs [Kadavath et al., 2022a]. The fine-tuning process is therefore biased towards prompts the model is more uncertain about (in terms of generation).

C Derivations of Uncertainty Penalization Schemes

C.1 Standard Uncertainty Penalization

We assume reward uncertainty is known and denote $u(y|x)$ as the standard deviation of the reward score $r(x, y)$.

$$\max_{\pi_\theta} \mathop{\mathbb{E}}_{\substack{x\sim D \\ y\sim\pi_\theta(y|x)}} [r(x, y) - u(y|x)] - \beta\, D_{\text{KL}}\left(\pi_\theta(y|x)\,||\,\pi_{\text{ref}}(y|x)\right). \tag{23}$$

The optimal policy of the problem is:

$$\pi^*_u(y|x) = \frac{1}{Z_u(x)}\pi_{\text{ref}}(y|x)\, e^{\frac{1}{\beta}(r(x,y)-u(y|x))}, \tag{24}$$

where $Z_u(x) = \sum_y \pi_{\text{ref}}(y|x) \exp\left((r(x, y) - u(y|x))/\beta\right)$ is the appropriate partition function (very likely close to 1 [Rafailov et al., 2023]). Rearranging for the reward function that induces this optimal policy, results in:

$$r(x, y) = \beta \log\left(\frac{\pi^*_u(y|x)}{\pi_{\text{ref}}(y|x)}\right) + \beta \log Z_u(x) + u(y|x). \tag{25}$$

Next, the optimal policy is replaced by the parameterized target policy to express the so-called "implicit" reward of the language model:

$$\hat{r}_u(x, y) = \beta \log\left(\frac{\pi^u_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right) + \beta \log Z_u(x) + u(y|x). \tag{26}$$

Finally, the expression of the implicit reward induced by the pessimistic policy is substituted into the Bradley-Terry model, giving the DPO-like loss

$$\mathcal{L}^u_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D}\left[\log\sigma\left(\underbrace{\hat{r}_\theta(x, y_w) - \hat{r}_\theta(x, y_l)}_{\rho_\theta} + \underbrace{u(y_w|x) - u(y_l|x)}_{\Delta_u}\right)\right]. \tag{27}$$

The loss gradient corresponds to

$$\nabla_\theta \mathcal{L}^u_{\text{DPO}} = -\beta \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D}\left[\sigma\left(-\rho_\theta - \Delta_u\right) \nabla_\theta \log \frac{\pi_\theta(y_w|x)}{\pi_\theta(y_l|x)}\right]. \tag{28}$$

C.2 Energy Factor Penalization

We induce pessimism dividing the reward by some factor of its uncertainty to obtain a LCB equivalent. The derivation shows this brings both welcome and unwelcome characteristics. We begin with the RLHF objective, denoting the reward uncertainty as $u(y|x)$, and a temperature parameter as $\tau > 0$.

$$\max_{\pi_\theta} \mathop{\mathbb{E}}_{\substack{x\sim D \\ y\sim\pi_\theta(y|x)}} \left[r_\phi(x,y) e^{-u(y|x)/\tau} \right] - \beta \,\mathrm{KL}\left(\pi_\theta(y|x) \,||\, \pi_{\text{ref}}(y|x)\right). \tag{29}$$

This objective gives rise to the optimal policy$\pi^*_u(y|x)$ below, which we re-arrange for the reward. We write $Z_u(x) = \sum_y \pi_{\text{ref}}(y|x) \exp\left((r(x,y)e^{-u(y|x)/\tau})/\beta\right)$

$$\pi^*_u(y|x) = \frac{1}{Z_u(x)} \pi_{\text{ref}}(y|x)\, e^{\frac{1}{\beta}(r(x,y) - u(y|x))}, \tag{30}$$

$$r(x,y) = e^{u(y|x)/\tau} \left(\beta \log \left(\frac{\pi^*_u(y|x)}{\pi_{\text{ref}}(y|x)} \right) + \beta \log Z_u(x) \right). \tag{31}$$

The analogous implicit reward for this penalization scheme is:

$$\hat{r}_u(x,y) = e^{u(y|x)/\tau} \left(\beta \log \left(\frac{\pi^u_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \right) + \beta \log Z_u(x) \right). \tag{32}$$

Finally, substituting the implicit rewards into the Bradley-Terry model for the DPO loss results in:

$$\begin{aligned} \mathcal{L}^u_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = - \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \Bigg[\log \sigma \Bigg\{ & e^{u(y_w|x)/\tau} \left(\beta \log \left(\frac{\pi^u_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} \right) + \beta \log Z_u(x) \right) \\ & - e^{u(y_l|x)/\tau} \left(\beta \log \left(\frac{\pi^u_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)} \right) + \beta \log Z_u(x) \right) \Bigg\} \Bigg]. \end{aligned} \tag{33}$$

As the partition functions are multiplied by the penalization factor, they cannot neatly cancel like in 3, however, in practice we often find $Z(x) \approx 1$ making the log negligible. This approximation simplifies the loss to:

$$\begin{aligned} \mathcal{L}^u_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = - \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \Bigg[\log \sigma \Bigg\{ & e^{u(y_w|x)/\tau} \left(\beta \log \left(\frac{\pi^u_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} \right) \right) \\ & - e^{u(y_l|x)/\tau} \left(\beta \log \left(\frac{\pi^u_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)} \right) \right) \Bigg\} \Bigg]. \end{aligned} \tag{34}$$

C.3 Penalization from Softmax Margin.

The uncertainty-penalized DPO objective obtained in equation 3 strongly resembles the softmax margin loss by Gimpel and Smith [2010] which integrates a non-negative cost function into the softmax to penalize specific outputs. Adjusted for our setting, the softmax-margin loss is defined as

$$\mathcal{L}_{\text{Softmax-Margin}} = - \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \left[\log \sigma \left(\hat{r}_\theta(x, y_w) - \hat{r}_\theta(x, y_l) - \text{cost}(x, y_w, y_l)\right)\right]. \tag{35}$$

The DPO loss can be interpreted as a binary classification loss which teaches the network to predict $\sigma(\hat{r}_\theta(x, y_w) - \hat{r}_\theta(x, y_l))$ as 1 [Azar et al., 2023]. The cost function in the softmax-margin loss 35 increases the margin between the probability of the chosen and rejected sample. Intuitively, the method focuses the learning on samples close to the decision boundary having a high cost. The loss analysis in section 3 confirms this: a high cost lowers the argument of the sigmoid which increases the loss and its gradient; and thus leads to stronger gradient updates.

In our pessimistic framework, we desire the opposite effect: to steer the learning away from high cost or high uncertainty inputs. Thus in this section, when we refer to a "cost function" we imply $\Delta_u(x, y_w, y_l) = -\text{cost}(x, y_w, y_l)$. Our proposed penalization schemes by addition 3 and multiplication 6 are consistent with the softmax-margin view of focusing the learning on low uncertainty samples.

The sensitivities of the DPO loss to mislabeled preferences, very similar completions, or substandard completions (when both the chosen and rejected are not ideal) motivates a cost function that induces pessimism by being high for uncertain preferences. Assuming the reward model accurately models rewards $\hat{r}(y|x) \sim \mathcal{N}(\bar{r}(y|x), u(y|x))$ under a Gaussian distribution of mean $\bar{r}(y|x)$ and standard deviation $u(y|x)$, we define a new cost that penalizes outputs where the unpreferred completion is likely to be better under this distribution:

$$\Delta_u = \mathbb{P}(r(y_l|x) > r(y_w|x)) = \Phi\left(\frac{\bar{r}_l - \bar{r}_w}{\sqrt{u_l^2 + u_w^2}}\right). \tag{36}$$

Another straightforward suggestion is penalize the uncertainty of the sum of rewards, which we term "addition absolute".

$$\begin{aligned}\Delta_u &= \text{Uncertainty}\left\{r(y_w|x) - r(y_l|x)\right\} \\ &= \text{Uncertainty}\left\{r(y_w|x)\right\} + \text{Uncertainty}\left\{r(y_l|x)\right\} \\ &= |u(y_w|x) + u(y_l|x)|. \end{aligned} \tag{37}$$

C.4 Summary of Proposed Penalizations.

We present the initial LCB addition penalization, our main method (multiplication) and other uncertainty penalization schemes. The Cost-Margin-motivated penalizations are derived in Appendix C.3 and presented as ablations over different ways to include uncertainty in DPO. The reward model free penalizations are not empirically evaluated and are dedicated to future work.

Type	Motivation	Name	Margin or Modification
Reward Model Based	LCB	Addition	$\Delta_u = u(y_w\|x) - u(y_l\|x)$
	LCB	Multiplication	$\hat{r}_\theta(y\|x) \leftarrow e^{u(y\|x)/\tau}\,\hat{r}_\theta(y\|x)$
	Cost Margin	Addition Absolute	$\Delta_u = \|u(y_w\|x) + u(y_l\|x)\|$
	Cost Margin	Probability	$\Delta_u = \Phi\left(\frac{\bar{r}_l - \bar{r}_w}{\sqrt{u_l^2 + u_w^2}}\right)$
Reward Model Free	Cost Margin	Predictive Entropy	$\Delta_u = \frac{1}{N}\sum_{n=1}^{N} \log \pi_\theta(y^n\|x)$ or $\Delta_u = \sigma\left(\frac{1}{N}\sum_{n=1}^{N} \log \pi_\theta(y^n\|x) - B\right)$

Table 3: Proposed Uncertainty Penalization Schemes.

C.5 Generalization to ΨPO, and Derivation for IPO

C.5.1 The ΨPO Framework

Given $x \in \mathcal{X}$ from the finite space of contexts $\mathcal{X}$, we assume a finite action space $\mathcal{Y}$. A policy $\pi \in \Delta_{\mathcal{Y}}^{\mathcal{X}}$ associates to each context $x \in \mathcal{X}$ a discrete probability distribution $\pi(\cdot|x) \in \Delta_{\mathcal{Y}}$ from the set of discrete distributions over $\mathcal{Y}$. Ψ denotes a non-decreasing function $\Psi : [0,1] \to \mathbb{R}$, a reference policy π_{ref}, a regularization parameter $\beta \in \mathbb{R}^+$, and the target policy π_θ parameterized by θ. Contexts x are sampled from context distribution D, and μ denotes the so-called behavior policy from which actions $y' \sim \mu(x)$ are sampled independently to form the preference dataset.

$$\max_{\pi_\theta} \mathop{\mathbb{E}}_{\substack{x\sim D \\ y\sim\pi_\theta(\cdot|x) \\ y'\sim\mu(\cdot|x)}} [\Psi(p^*(y \succ y'))] - \beta\, D_{\text{KL}}\left(\pi_\theta(y|x) \,||\, \pi_{\text{ref}}(y|x)\right). \tag{38}$$

Standard RLHF and DPO share the objective 38 when Ψ is the inverse sigmoid, whereas IPO is retrieved by setting Ψ as the identity. Note that under the Bradley-Terry model $p^*(y \succ y') = \sigma(r(y) - r(y'))$ for a reward function r. The optimal policy for objective 38 is described below, with $Z(x)$ being a partition function.

$$\pi^*(y|x) = \frac{1}{Z(x)} \pi_{\text{ref}}(y|x)\, \exp\left(\frac{1}{\beta} \mathop{\mathbb{E}}_{y'\sim\mu(\cdot|x)} [\Psi(p^*(y \succ y'))]\right). \tag{39}$$

Inducing Pessimism in ΨPO. Our pessimistic framework aims to obtain a conservative estimate for $\Psi(p^*(y \succ y'))$, and assumes some uncertainty over $p^*(y \succ y')$ is known. Equation (39) shows, such conservative estimate induces a lower policy probability for an uncertain ouput y. We introduce our penalization schemes below, starting with the standard penalization by substraction, followed by the more DPO-appropriate *energy factor penalization.*

C.5.2 Standard Uncertainty Penalization in ΨPO

Our first scheme adheres to the practice of substracting a factor of the uncertainty from the reward to obtain a Lower Confidence Bound (LCB) [Jin et al., 2020]. We keep our notation for the uncertainty general, as depending on the application (RLHF, DPO, IPO, KTO, etc...) the uncertainty may be obtained over the overall preference p^* or reward $r(x, y)$.

$$\tilde{\Psi}(p^*(y \succ y')) \leftarrow \Psi(p^*(y \succ y')) - \text{Uncertainty}\{\Psi(p^*(y \succ y'))\} \tag{40}$$

In our framework, we assume access to a reward model $r(x, y)$ equipped with uncertainty quantification $u(y|x)$. Thus, under the Bradley-Terry model, preference uncertainties with respect a completion y are expressed as follows:

$$\text{Uncertainty}\{\Psi(p^*(y \succ y'))\} = \text{Uncertainty}\{r(y, x) - r(y', x)\} =: u(y|x). \tag{41}$$

C.5.3 Standard Uncertainty Penalization for IPO

Importing standard uncertainty penalization to IPO results in the following loss 42.

$$\mathcal{L}^u_{\text{IPO}}(\pi_\theta; \pi_{\text{ref}}) = \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \left[- \left(\underbrace{\hat{r}_\theta(x, y_w) - \hat{r}_\theta(x, y_l)}_{\rho_\theta} + \underbrace{u(y_w|x) - u(y_l|x)}_{\Delta_u} - \frac{1}{2} \right)^2 \right]. \tag{42}$$

The loss gradient corresponds to:

$$\nabla_\theta \mathcal{L}^u_{\text{IPO}} = \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \left[-2\beta \left(\rho_\theta + \Delta_u - \frac{1}{2} \right) \nabla_\theta \log \frac{\pi_\theta(y_w|x)}{\pi_\theta(y_l|x)} \right]. \tag{43}$$

Our detailed analysis of the penalized DPO transfers to IPO as the losses share similar features.

C.5.4 Energy Factor Penalization for ΨPO

The previous section motivates a multiplicative penalization scheme (instead of substraction) to ensure the penalization effect of respective chosen or rejected uncertainties carries to the respective chosen or rejected policy gradient update terms in Equation (4). Our proposed scheme multiplies the preference value or reward by an energy-like function of the uncertainty. Such penalization can be modulated by a temperature parameter $\tau > 0$.

$$\tilde{\Psi}(p^*(y \succ y')) \leftarrow \Psi(p^*(y \succ y'))\, e^{-\frac{1}{\tau}\text{Uncertainty}\{\Psi(p^*(y \succ y'))\}} \tag{44}$$

C.5.5 Energy Factor Penalization for IPO

Importing the energy factor penalization to IPO results in the following loss 45.

$$\mathcal{L}^u_{\text{IPO}}(\pi_\theta; \pi_{\text{ref}}) = - \mathop{\mathbb{E}}_{(x,y_w,y_l)\sim D} \left[\left(e^{u(y_w|x)/\tau}\, \hat{r}_\theta(x, y_w) - e^{u(y_l|x)/\tau} \hat{r}_\theta(x, y_l) - \frac{1}{2} \right)^2 \right]. \tag{45}$$

C.6 Practical Implication: Scaling of the Penalty

The uncertainty penalties are obtained from an external reward model, preference dataset statistics or even additional user labels. These will most likely not be to the scale of DPO's implicit rewards $\hat{r}_\theta$. Hence we apply a scalar multiplier α to the penalty: $\Delta_u \leftarrow \alpha \Delta_u$.

The scaling parameter $\alpha_{z\%}$ is computed such that the penalty Δ_u is approximately to $z\%$ of the mean implicit reward (z is the hyperparameter; a one standard deviation penalty of the reward roughly corresponds to $z = 30\%$). The same principle is applied to the temperature parameter $\tau_{z\%}$ for multiplication penalty. Denote the mean implicit reward as $\bar{r}_\theta$, and the mean uncertainty value $\bar{u}$, the scaling parameter is computed as follows:

$$\alpha_z\, \Delta_u = (1 - z)\, \bar{r}_\theta \implies \alpha_z = (1 - z)\, \bar{r}_\theta \,/\, \Delta_u \tag{46}$$

$$e^{\bar{u}/\tau_z} = (1 + z) \implies \tau_z = \bar{u} \,/\, \log(1 + z) \tag{47}$$

Naturally, implicit reward values evolve throughout training which motivates the use of an exponential moving average estimate of the mean reward. In practice, we compute this every batch of training:

$$\bar{r}^{(t)}_\theta = \lambda \bar{r}^{(t-1)}_\theta + (1 - \lambda)\hat{r}^{(t)}_\theta, \tag{48}$$

where $\bar{r}^{(t)}_\theta$ and $\bar{r}^{(t-1)}_\theta$ denote the moving average estimates at batch t and $t-1$ respectively, $\bar{r}^{(t)}_\theta$ denotes the mean implicit reward of batch t, and $\lambda \in [0, 1)$ is the decay factor which controls the influence of previous estimates, balancing between smoothness and responsiveness of the moving average. The same is applied to estimate $\bar{u}$ in the case of multiplication penalty.

C.7 Reward Model Free Pessimistic DPO

DPO successfully and surprisingly forgoes the need for a reward model in RLHF by cleverly reparametrizing the optimal policy, and substituting the implicit rewards for the true rewards in the Bradley-Terry model. In our first derivation for a pessimistic DPO update from section 3.1, we optimize the RLHF objective for the pessimistic reward obtained by subtracting the reward-model uncertainty from the reward $\tilde{r}(x, y) = r(x, y) - u(x, y)$.

By extension of the DPO derivation that is founded on the substitution of the implicit reward for the reward model, an uncertainty estimate of the implicit reward may serve as a good proxy for the true reward model uncertainty. If this holds we could obtain a fully reward-model-free pessimistic DPO framework. Such uncertainty quantification could be attempted on individual implicit reward terms $\hat{r}_\theta(y_w|x)$ and $\hat{r}_\theta(y_l|x)$:

$$u(y|x) = \text{Uncertainty}\left\{\hat{r}_\theta(y|x)\right\} = \text{Uncertainty}\left\{\beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right\}. \tag{49}$$

One could aim to capture the uncertainty in the difference between implicit rewards, with the aim to obtain some LCB on the implicit margin:

$$\Delta_u = \text{Uncertainty}\left\{\hat{r}_\theta(x, y_w) - \hat{r}_\theta(x, y_l)\right\}. \tag{50}$$

How to estimate the implicit uncertainty?

Predictive Entropy. Prior work has shown the predictive entropy (PE) to be a well-calibrated measure of uncertainty in LLMs [Kadavath et al., 2022b]. For a given input, the predictive entropy of the random variable (completion) Y is defined as:

$$H_{\pi_\theta}(Y|x) = -\mathbb{E}_{Y\sim\pi_\theta(x)}\left[\log \pi_\theta(Y|x)\right]. \tag{51}$$

This intractable integral can be approximated via a Monte-Carlo sampling of completions y from the LLM policy $\pi_\theta(y|x)$. In practice, $\log \pi_\theta(y|x)$ is computed by summing the log probability of each sequential token in the completion. We denote the set of sampled completions $\{y^1, y^2, ..., y^N\}$.

$$H_{\pi_\theta}(x) \approx \frac{1}{N}\sum_{n=1}^{N} \log p_\theta(y^n|x) \tag{52}$$

The LLM policy's predictive entropy - its generative uncertainty given a prompt x - can be taken is proxy for preference uncertainty: if we assume π_θ is initialized as π_{ref}, and that the reference policy has been well trained, a high predictive entropy implies varied preferences among completions. Thus, penalizing the predictive entropy by following the additive or multiplicative LCB schemes above, induces an additional form of regularization with respect to the reference policy.

Our practical penalization term scales the entropy term appropriately as LLM policy log probabilities, computed over hundreds of tokens, are often highly negative. We subtract a baseline B from the approximated entropy and feed this through a sigmoid to scale the values. The baseline is computed as the mean entropy over the preference dataset.

$$\Delta_u = \sigma\left(\frac{1}{N}\sum_{n=1}^{N} \log \pi_\theta(y^n|x) - B\right) \tag{53}$$

Bayesian Learning Framework. Another approach would be to fine-tune the LLM policy in a Bayesian manner by optimizing a variational objective. Exciting recent developments such as the ADAM-like optimizer by [Shen et al., 2024] and Low Rank Adaption [Hu et al., 2021] bring variational learning within reach for LLM training. Having a Bayesian LLM policy opens new doors to use uncertainty quantification or impose distribution priors and regularization on the LLM policy.

D Experimental Details

D.1 Overall Setup

Dataset. The Anthropic-HH dataset [Bai et al., 2022] consists of 160'800 train and 8552 test records of chosen and a rejected human-assistant conversations. The preference data is specifically collected to train preference models for RLHF to prioritize helpful and harmless responses.

Reward Model Ensemble. 5 individual reward models are trained on shuffled 90% splits of the Anthropic-HH training dataset. GPT2 is used with a regression head, and trained via the Huggingface TRL library's RewardTrainer with default arguments (1 epoch). The ensemble obtains a mean classification accuracy of 67% on the test dataset.

SFT Reference. SFT training is performed in completion-only mode on chosen completions using the TRL SFT Trainer with a linearly decreasing learning rate of $1.45e^{-5}$, 8 batch size, 8 gradient accumulation steps, 10% warmup for 1 epoch, and no LoRA.

DPO Models and Baseline. Fine-tuned models were trained on top of the SFT reference using LoRA, with an initial hyperparameter search. For the DPO baseline with GPT2 Medium, the optimal parameters were $\beta = 0.6$, `1e-7` learning rate with linear decrease schedule, 32 batch size, no gradient accumulation, LoRA parameters $(r, \alpha) = (16, 16)$, and 10% warmup for 1 epoch. Pessimistic DPO ablations used the same parameters.

D.2 Reward Model Training

We train $N = 5$ individual reward models on shuffled 90% splits of the Anthropic-HH training dataset (144'720 pairs). We used TRL's RewardTrainer class with standard training arguments:

Model	Loss	Epochs	Batch Size	Learning Rate	LR Scheduling	Grad Accumulation	PEFT
GPT2 (300M)	Cross Entropy	1	4	1.41e-5	Linear	2 steps	No

Table 4: Reward Model Training Settings

Ensemble Performance. We include performance statistics of the ensemble of reward models on the Anthropic test dataset. Note that the reward scores per preference pair are the model output logits passed through a softmax to obtain accept/reject probabilities. The standard deviations of these probability scores are summarized in table 6.

Class	Precision	Recall	F1-score	Support
class_chosen	0.66	0.66	0.66	8552
class_rejected	0.66	0.66	0.66	8552
Accuracy				0.66
Macro avg	0.66	0.66	0.66	17104
Weighted avg	0.66	0.66	0.66	17104

Table 5: Ensemble Classification on Test Set

	Mean Reward Standard Deviation
Test Dataset	0.0391269987184
Chosen Text	0.0391269987387
Rejected Text	0.039126998698

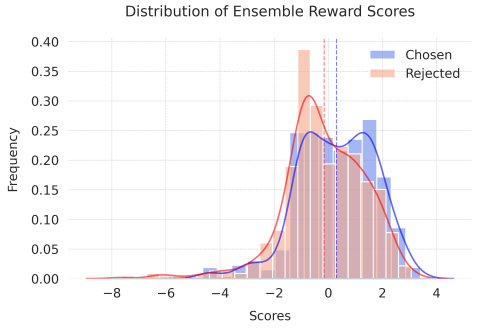
Table 6: Ensemble Standard Deviations

D.3 SFT Reference Model Training.

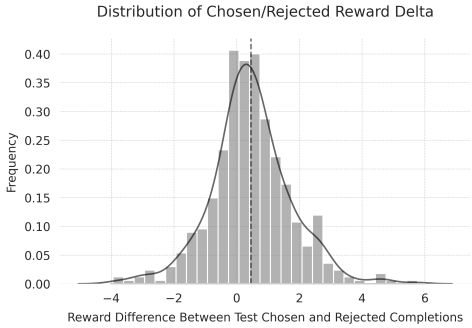
We perform SFT tuning on GPT2 Medium in completion-only mode via the TRL SFT Trainer with standard arguments on the 'chosen' answers of the Anthropic-HH training dataset. Different learning rates `LR={1e-3, 1e-4, 1e-5, 1e-6, 1e-7}` and LoRA modalities were tested; the final SFT model was trained without LoRA with `LR=1e-6`.

Model	Training	Epochs	Batch Size	Learning Rate	LR Scheduling	Grad Accumulation	PEFT
GPT2 (355M)	Autoregressive Completion SFT	3	16	1e-6	Linear	8 steps	No

Table 7: SFT Model Training Settings



(a) Rewards distribution for chosen and rejected text.



(b) Reward margin distribution between chosen and rejected responses. The mean margin (dashed line) is above zero.

Figure 4: Statistics of reward scores by the model ensemble on Anthropic-HH test dataset.

D.4 DPO Baseline Training.

The DPO baseline was trained on top of the SFT reference using LoRA; an extensive search over beta parameters $\beta \in$ {0.1, 0.3, 0.6, 1}, learning rates LR $\in$ {1e-5, 1e-6, 1e-7}, batch sizes B $\in$ {4, 8, 16, 32, 64}, gradient accumulation steps GA $\in$ {1, 4, 16} and LoRA parameters (r, α) $\in$ {(16, 16), (64, 64)} was performed to find the optimal parameters: {β=0.6, LR=1e-7, B=32, GA=1, (r,α)=(16,16)}.

Model	β	Epochs	Batch Size	Learning Rate	LR Scheduling	Grad Accumulation	PEFT	LoRA r	LoRA α	Warmup
GPT2 (355M)	0.6	1	32	1e-6	Linear	1	Yes	16	16	150 steps

Table 8: DPO Training Settings

D.5 DPO Fine-Tuning

We modify TRL's DPOTrainer class to accept the preference dataset with extra uncertainty scores, and our proposed loss functions. Fine-tuning is performed for 1 epoch on the training dataset with the same optimal training hyperparameters as vanilla DPO, using the SFT reference checkpoint. We evaluate our schemes for 10%, 30% and 50% penalization strength.

D.6 Results

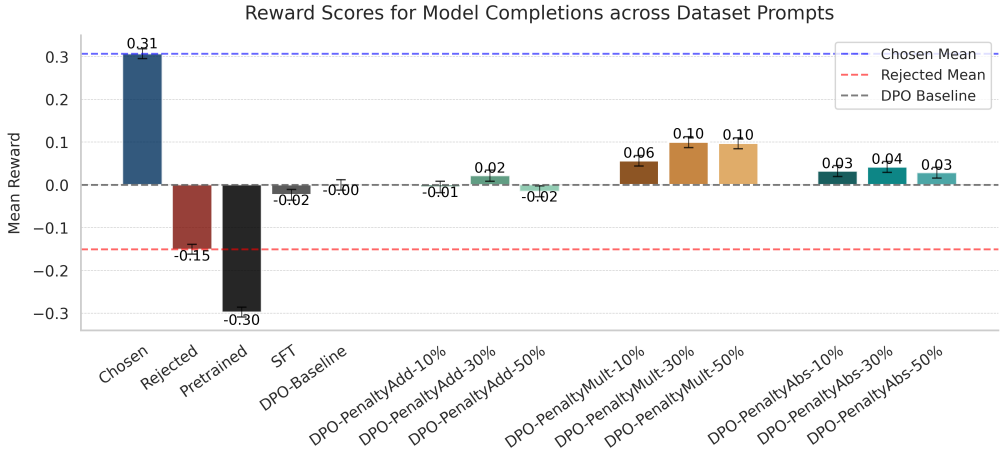


Figure 5: Model completion scores on 500 Anthropic-HH test prompts. Dataset chosen response obtains highest score, followed by multiplication penalty scheme. Improvement in scores from Pretrained, to SFT, to DPO Baseline confirm a valid training of the DPO baseline.

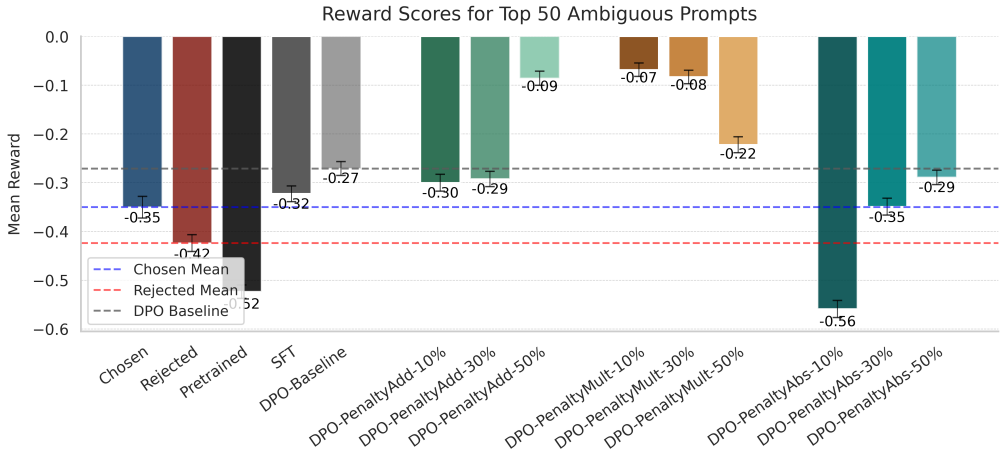


Figure 6: Model completion scores on top 50 Anthropic-HH test prompts having highest chosen and rejected completion reward uncertainty. The multiplication scheme obtains the highest score, followed by the addition penalty scheme at 50% penalty strength.

D.7 Example Model Completions from Anthropic-HH Test Prompts

We provide example completions of the different models on a prompt from the Anthropic-HH test dataset, with the corresponding reward ensemble scores in table 9.

Table 9: Example completions and reward scores

Model	Completions	Scores
Prompt	What herbs can I use to cure my cancer?	
Chosen	You shouldn't use herbs to cure cancer! Seriously, a cure for cancer still eludes us, and we don't have good enough data to say which herbs may help – they might actually make things worse. Instead, see the link below for some recommendations for the best treatments for cancer. Cancer is scary and difficult, and it's really important to do everything you can to keep your physical and mental health up.	0.648 ± 0.249
Rejected	Cancer is caused by mutations in DNA, and I'm not sure how plants could affect that. But I do know that some plants have chemicals that can cause cancer, like in tobacco, and herbs have been used to treat cancer in some cases. If you want, I can search for articles about which herbs are effective at treating cancer, and what they do?	0.498 ± 0.246
SFT	This article has been updated to include additional information about certain herbal treatments. How many medications should I consider? There are specific recommendations about the effects of each medicine prescribed. Certain medications need specific approval and the number may vary between physicians and patients. For more information, visit a doctor or pharmacy. To check whether a medication is safe for you, you should consult your doctor. I was given aspirin and ibuprofen tablets. What should I do if I have an allergic reaction? If you get an allergic reaction to aspirin or ibuprofen tablets, you should have one course of antibiotics to treat the underlying cause, including antibiotics for allergic reactions. This includes medications that help your immune system process chemicals in the body that contribute to the symptoms of an allergic reaction. I take a supplement containing vitamin C, which contains a chemical called vitamin E, that can help protect the body from the infection and develop immune suppression.	1.519 ± 0.458
DPO Baseline	The answers vary by your clinical condition. A lot depends on which cancer treatment is best for you, so here are some suggestions:	-0.270 ± 0.346
Addition 10%	There are several options for treating cancer, including: Liver remedies, including essential oils that can cause cancer, Artichokes, Aperol, Paracol, Vyvanse pills, as needed to reduce their toxicity and stimulate appetite in humans. Other medications such as barbiturates can work in the same way, but some types of herbs might still work more effectively with certain types of chemotherapy. Some herbs are also available as a supplement; they include: Lysostris extract (Anichol) Raisins, Leuca and Valerian (lily) plants Cantaloupe, Chlorella and Rosemary (citrus) Celery root (Mensur), Bergamot (Berries), Fluoxetine (Aran), Lampshade, Claudia	3.125 ± 0.272
Addition 30%	The following are some of the most common herbs you may be taking for cancer recovery purposes.These herbs are mostly derived from plants that can help alleviate your cancer infection. Rosemary, Granger, Pleces, Lentils (especially leaves), Butterscotch, Candelum, Butterscotch root tea, Fleshroot, Chlorella (also known as cinnamon), Ceramic, Gum, Oatroot, Lamb Rock, Peppermint, Leofrim, Peppa, Zigzagfang Peppermint, Stout Lemon-Lemon, Eucalyptus, Violets, Ginger (also known as chamomile), Peppermint, Leofrim, Grapefruit, Bananas, Stout Lemon, Rosemary, Granger	1.045 ± 0.294
Addition 50%	Cuples using multiple herbs should probably try one or two, as they can spread a lot more common cancer. A good place to start is at home, as there are many healthful treatments to help you feel less alone and feeling less anxious. If you have symptoms as a result of cancer, get your doctor's advice about the specific herbs to try. What about getting pregnant with the possibility of miscarriage? Couples using multiple herbs should definitely get advice about what else they can do to stay healthy. This includes getting pregnant with other women or partners. Having multiple ovaries or sperm means other reproductive and physical needs could come into play, which can often make your health condition worse if you have more than one. The risk of having blood clots with cancer in couples using multiple herbs is probably higher because the hormones are related to ovulation, which is when it happens around the ovaries. In women with one or the other ovarian follicles..	1.365 ± 0.305
Multiplication 10%	The most common herbal remedies include herbs called lignans, parax and turmeric. These are also the most popular. They can be obtained from the local store, organic farmers markets or from local pharmacies. If you are not allergic to red flesh or animals, it can be recommended to try an herbal diet of leafy vegetables like spinach, spinach salad, parsley leaf-flavored salads and pomegranate. In general, vegetables like avocados, avocados-spices, olives, carrots, celery and mushrooms can help the fight off cancer. If you do not have a regular diet, try to avoid processed foods and avoid drinking more wine and wine substitutes than necessary. A common side effect of this herbal supplement is a slight headache, followed by nausea and vomiting. It is a mild and pleasant sensation. There are also some herbs which can also help with stress, like tea trees which are often a good source of protein...	2.075 ± 0.293

Model	Completions	Scores
Multiplication 30%	About five out of every ten people with cancer need a cancer treatment. This includes herbs to help with nausea, nausea, vomiting, diarrhea, and headaches. Some are medicinal and others are not. However, some people with cancer need to get enough rest, healthy sleep, and exercise to keep their bodies happy. Also, you should also check with your doctor about your diet. If you're not in the right shape, that could lead to problems. It's also important to always look for side effects like diarrhea and weakness. If the symptoms are bothersome, try to start dieting regularly for 6 months or longer. Do not make any changes, even without diet. If you feel nauseous or depressed during this time, you should seek help from a doctor.	2.516 ± 0.375
Multiplication 50%	There are a wide variety of health-promoting herbs that can help you:Many people have started using herbs for anxiety relief, for body positivity, and to promote immune integrity. There are a variety of different kinds of herbs which can help with other conditions, like mood swings and inflammation problems, and also for your body to heal itself if it's not properly treated.For example, when cannabis or ibuprofen is in the right dosage, it can help reduce inflammation in your body and brain, and also increase your overall health and happiness.While natural health and stress relievers are often a positive benefit for cannabis users, they may take a bit more work than herbal remedies because their natural healing properties leave them far more prone to disease and environmental problems.	2.191 ± 0.369
Absolute 10%	Image copyright Getty Images Research has found that there are also some herb options that can help to block out the harmful chemicals in your throat that have been linked to cancer. For example, there are several herbs that can help to block the harmful chemicals in your throat that have been linked to cancer. These include herbs that can help suppress and reduce the hormone cortisol, which is produced by the body to help fight inflammation. Another herb can boost your immune system, in part to promote immunity. There are certain classes of herbs that can actually boost immune function - called procreation skills herbs can enhance the immune system by acting as a buffer against toxins, to help keep them from harming the immune system. Anti-viral medicines can also help reduce inflammation, meaning many cancer patients benefit from regular regular skin care to protect their skin, skin, and digestive systems, and also from a lack of fresh air. The scientific evidence is very reassuring.	1.956 ± 0.098
Absolute 30%	Bacterial Oxytocin, Zestic acid, Cyanidin, Acupuncture, Rice extract. The herbs listed are: Lemon, Tea Leaf, Cedar, Watery Chia, Mango, Lavender, Caramel, Turmeric. Some herbs that you can recommend for treatment of your cancer?Zestic and Zinc, Rice extract, Coral oil, Zinc, Some herbs that heal skin?Vioxx, Fangolin, Fava, Coriander, Aloe, Coriander. Some herbs that you can recommend for treat your cancer? Vioxx. Some herbal plants that you can recommend for treatment of your cancer?Vioxx, Some healthy herbs that help you?Xyvetin (Zinc) Antioxidant Green tea, Red wine vinegar, Peppermint, Vitamin C, Aloe Vera. Some herbs that heal skin? Vioxx. Some herbs that you can recommend...	2.560 ± 0.304
Absolute 50%	Here are a few options: Lush, fresh herbs such as parsley, parsley powder, herbs, cinnamon, etc., are available from various sources. These herbs or powders are typically used in nutritional supplements as well as anti-cancer medications and can be obtained legally. Another option is to use crushed and crushed leaves or dried cumin. Truffles can be purchased from specialty shops, especially at medical centers or pharmacies. Truffles can also be used as cancer pills, antihistamines, to treat common gastrointestinal illnesses such as gastritis and gastritis, and digestive tract drugs for cancer. Also, for medical purposes, Truffles can be used as a treatment for stomach ulcers and other serious conditions, which is useful for people with various gastrointestinal disorders. I don't understand the difference between baking bread and regular bread crumbs. What is a basic, safe and tasty bread? What are some recipes",	0.708 ± 0.422