On the Sparsity of Direct Preference Optimization: Weight Disentanglement in the NTK Regime

Kensuke Sasaki
Department of Computer Science
The University of Tokyo
sasaki-kensuke0222@g.ecc.u-tokyo.ac.jp

Issei Sato
Department of Computer Science
The University of Tokyo
sato@g.ecc.u-tokyo.ac.jp

Abstract

Direct Preference Optimization (DPO) has been widely adopted for aligning language models with human preferences. Recent empirical work has revealed that DPO induces remarkably sparse parameter updates, yet no theoretical explanation for this phenomenon has been offered. We analyze DPO's optimization dynamics under a neural tangent kernel linearization and show that the logistic structure of the DPO loss gives rise to an implicit maximum-margin bias. Building on this observation, we introduce the concept of *dual sparsity*: the DPO update direction is a sparse linear combination (data-level sparsity, driven by support vectors) of sparse gradient feature vectors (parameter-level sparsity, arising from gradient cancellation between preferred and dispreferred responses). We formalize this structure in our main theorem, bound the number of support pairs by a margin-dependent spectral quantity, and present a framework relating dual sparsity to weight disentanglement—a condition favorable for model merging. Experiments on GPT-2 Small and Medium and on 7–9B instruction-tuned models (Llama-3-8B, Mistral-7B, Gemma-2-9B) with the Anthropic HH-RLHF and UltraFeedback datasets confirm both sources of sparsity, and experiments on GPT-2 show that DPO reduces cross-task interference by up to $35\times$ compared to supervised fine-tuning.

1 Introduction

Direct Preference Optimization [Rafailov et al., 2023] has emerged as a dominant paradigm for aligning large language models with human preferences. By casting the alignment objective as a logistic loss over preference pairs, DPO circumvents the instabilities of reinforcement learning from human feedback while retaining strong empirical performance. As practitioners increasingly train separate DPO models for distinct objectives—helpfulness, harmlessness, instruction following—there is growing interest in composing these independently trained models via *model merging* [Ilharco et al., 2023, Yu et al., 2024], which combines task-specific parameter updates (task vectors) into a single multitask model without additional training.

A striking but poorly understood empirical observation is that DPO fine-tuning induces highly sparse parameter updates. Mukherjee et al. [2025] recently demonstrated that reinforcement learning-based alignment concentrates its modifications onto a small subnetwork of the pre-trained model, leaving the vast majority of parameters essentially unchanged. This sparsity stands in marked contrast to supervised fine-tuning (SFT), which typically distributes updates broadly across the parameter space. Independently, Ortiz-Jiménez et al. [2023] identified *weight disentanglement*—the condition that task vectors act on non-overlapping regions of parameter space—as a key property enabling successful model merging. A natural question arises: is the sparsity of DPO a mere coincidence, or does it reflect a deeper structure in the optimization that is systematically favorable for composing task vectors?

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

In this paper, we provide a theoretical answer to this question. We analyze the implicit bias of DPO under a neural tangent kernel (NTK) linearization [Jacot et al., 2018] and show that the logistic loss structure, combined with the geometry of preference-pair gradients, gives rise to a phenomenon we call *dual sparsity*. The DPO update direction converges to the maximum-margin solution in the NTK feature space, inheriting two complementary forms of sparsity: **data-level sparsity**, where only a small subset of preference pairs (support vectors) contribute to the final update, and **parameter-level sparsity**, where the gradient feature vector for each pair is itself sparse due to gradient cancellation between preferred and dispreferred responses that share common tokens. Building on this, we present a framework relating dual sparsity to weight disentanglement, and verify empirically that DPO produces cross-task inner products an order of magnitude smaller than those under SFT.

Our main contributions are as follows:

- (**C1**) We reduce DPO to maximum-margin classification under NTK linearization and introduce *dual sparsity*—a joint data-level and parameter-level sparsity structure—formalized in theorem 3. This provides the first theoretical explanation for why DPO induces sparse parameter updates without any explicit sparsity-promoting regularization.
- (**C2**) We present a framework relating dual sparsity to weight disentanglement [Ortiz-Jiménez et al., 2023] via a triplet-space formulation of the DPO margin (section 3.3). The framework reduces the disentanglement condition to a per-triplet inner product whose magnitude is governed by the parameter-space overlap between tasks—an empirical question that we evaluate in section 4.
- (**C3**) We validate both sources of sparsity on GPT-2 Small (124M) and Medium (355M) fine-tuned with the Anthropic HH-RLHF [Bai et al., 2022] and UltraFeedback [Cui et al., 2024] datasets, and at modern scale on Llama-3-8B-Instruct, Mistral-7B-Instruct-v0.2, and Gemma-2-9B-it. DPO feature vectors are over 90% near-zero from 124M to 9B parameters, and the time-integrated weights concentrate on a small subset of pairs within three epochs of full fine-tuning at the 7–8B scale. On GPT-2, cross-task interference under DPO is reduced by $11\times$ to $35\times$ relative to SFT across all configurations.

The remainder of the paper is organized as follows. Section 2 reviews the DPO formulation and weight disentanglement framework. Section 3 presents our theoretical analysis, culminating in the dual sparsity theorem and a framework connecting it to weight disentanglement. Section 4 provides experimental validation. Section 5 discusses related work, and section 6 concludes with limitations and future directions.

2 Preliminaries

2.1 Direct Preference Optimization

Let π_{ref} denote a reference policy, typically a pre-trained or supervised fine-tuned (SFT) model [Rafailov et al., 2023]. Given a prompt x and a pair of responses (y^+, y^-) where y^+ is preferred and y^- is dispreferred, the log-probability margin under parameters θ is

$$m_i(\theta) = \beta \left(\log \frac{\pi_\theta(y_i^+ \mid x_i)}{\pi_\theta(y_i^- \mid x_i)} - \log \frac{\pi_{\text{ref}}(y_i^+ \mid x_i)}{\pi_{\text{ref}}(y_i^- \mid x_i)} \right), \tag{1}$$

with temperature $\beta > 0$. For a preference dataset $\mathcal{D} = \{(x_i, y_i^+, y_i^-)\}_{i=1}^n$, the DPO loss is the logistic loss

$$\ell_i(\theta) = \log\left(1 + e^{-m_i(\theta)}\right), \tag{2}$$

whose gradient w.r.t. θ is weighted by $\sigma(-m_i(\theta)) \in (0, 1)$: the gradient vanishes for "easy" pairs already preferred by the policy, focusing optimization on the "hard" subset near the decision boundary.

2.2 Weight Disentanglement

We review the notion of weight disentanglement [Ortiz-Jiménez et al., 2023], which characterizes an ideal condition for composing independently trained task vectors without cross-task interference.

Consider a neural network $f\colon \mathcal{X} \times \Theta \to \mathcal{Y}$ with pre-trained parameters θ_0. A task t is specified by a data support $\mathcal{D}_t \subseteq \mathcal{X}$, an input distribution μ_t with $\mathrm{supp}(\mu_t) = \mathcal{D}_t$, and a target function $f_t^*\colon \mathcal{D}_t \to \mathcal{Y}$. Fine-tuning f on task t yields parameters θ_t^*, and the corresponding task vector is defined as $\tau_t := \theta_t^* - \theta_0$.

Definition 1 (Task Arithmetic, Ilharco et al., 2023)**.** Let $\mathcal{T} = \{\tau_t\}_{t\in[T]}$ be a set of task vectors with mutually disjoint supports $\mathcal{D}_t \cap \mathcal{D}_{t'} = \emptyset$ for $t \neq t'$. The network f satisfies the *task arithmetic property* around θ_0 if

$$f\left(x;\, \theta_0 + \sum_{s=1}^{T} \alpha_s \tau_s\right) = \begin{cases} f(x;\, \theta_0 + \alpha_t \tau_t) & x \in \mathcal{D}_t, \\ f(x;\, \theta_0) & x \notin \bigcup_t \mathcal{D}_t. \end{cases} \tag{3}$$

Definition 2 (Weight Disentanglement, Ortiz-Jiménez et al., 2023)**.** The network f is *weight-disentangled* with respect to $\mathcal{T}$ and $\{\mathcal{D}_t\}$ if there exist functions $g_t\colon \mathcal{X} \times \Theta \to \mathcal{Y}$ with $g_t(x;\cdot) = 0$ for $x \notin \mathcal{D}_t$ and a residual g_0 with $g_0(x) = 0$ for $x \in \bigcup_t \mathcal{D}_t$, such that

$$f\left(x;\, \theta_0 + \sum_{t=1}^{T} \alpha_t \tau_t\right) = \sum_{t=1}^{T} g_t(x;\, \alpha_t \tau_t) + g_0(x). \tag{4}$$

It is straightforward to verify that weight disentanglement (definition 2) implies the task arithmetic property (definition 1) by setting $g_t(x; \alpha_t\tau_t) = f(x; \theta_0 + \alpha_t\tau_t)\mathbf{1}(x \in \mathcal{D}_t)$ and $g_0(x) = f(x;\theta_0)\mathbf{1}(x \notin \bigcup_t \mathcal{D}_t)$.

3 Theoretical Analysis

Recent empirical work has shown that alignment fine-tuning via DPO induces highly sparse parameter updates relative to the pre-trained weights [Mukherjee et al., 2025]. Independently, weight disentanglement has been identified as a desirable property for robust model merging [Ortiz-Jiménez et al., 2023]. The following questions remain open:

1. **Why does DPO induce sparsity?** What mechanism in the DPO optimization dynamics leads to sparse parameter updates, despite the absence of any explicit sparsity-promoting regularization?
2. **What are the implications of this sparsity for cross-task interference?** Specifically, does the sparsity inherent in DPO updates reduce the interference between independently trained task vectors?

This section addresses both questions. We analyze the implicit bias of DPO under a neural tangent kernel (NTK) linearization (section 3.1) and identify a *dual sparsity* structure that emerges from the interplay between the logistic loss and the geometry of the gradient features (section 3.2). We then present a framework relating this structure to weight disentanglement, with empirical evaluation deferred to section 4 (section 3.3).

3.1 DPO as Maximum-Margin Classification in Tangent Space

We begin by linearizing the DPO margin around the initial parameters θ_0. Under the neural tangent kernel (NTK) regime [Jacot et al., 2018], the margin admits a first-order approximation:

$$m_i(\theta) \approx m_i(\theta_0) + \langle \Phi_i,\, \Delta\theta \rangle, \tag{5}$$

where $\Delta\theta = \theta - \theta_0$ and the *gradient feature vector* Φ_i is defined as

$$\Phi_i := \beta \left(\nabla_\theta \log \pi_{\theta_0}(y_i^+ \mid x_i) - \nabla_\theta \log \pi_{\theta_0}(y_i^- \mid x_i) \right) \in \mathbb{R}^p. \tag{6}$$

Here p denotes the total number of parameters. Each Φ_i encodes the direction in parameter space along which a perturbation $\Delta\theta$ most effectively increases the preference margin for pair i. Under this linearization, the DPO loss reduces to a binary logistic regression over the features $\{\Phi_i\}_{i=1}^n$ with offsets $b_i := m_i(\theta_0)$:

$$\ell_i(\Delta\theta) = \log(1 + \exp(-\langle \Phi_i,\, \Delta\theta \rangle - b_i))\,. \tag{7}$$

With the standard DPO initialization $\pi_{\theta_0} = \pi_{\mathrm{ref}}$, the offsets vanish: $b_i = m_i(\theta_0) = 0$ by eq. (1).

We make the following assumption, which is standard in the analysis of implicit bias for logistic regression [Soudry et al., 2018]:

Assumption 1 (Linear Separability)**.** There exist $\Delta\theta \in \mathbb{R}^p$ and $\gamma > 0$ such that $\langle \Phi_i,\, \Delta\theta \rangle + b_i \geq \gamma$ for all $i \in [n]$.

Under assumption 1, the gradient flow on the total loss $L(\Delta\theta) = \sum_i \ell_i(\Delta\theta)$ starting from $\Delta\theta(0) = 0$ admits the explicit trajectory

$$\Delta\theta(T) = \sum_{i=1}^{n} A_i(T)\, \Phi_i, \quad A_i(T) := \int_0^T \sigma(-m_i(\theta(t)))\, dt, \tag{8}$$

where $A_i(T)$ is the time-integrated gradient weight for pair i. Applying the implicit-bias result of Soudry et al. [2018] to this setup yields:

Lemma 1 (Implicit Bias of DPO)**.** *Under assumption 1, the normalized parameter update converges to the L_2 maximum-margin direction:*

$$\frac{\Delta\theta(T)}{\|\Delta\theta(T)\|} \xrightarrow{T\to\infty} u^\star, \quad \textit{where } u^\star \textit{ solves} \quad \max_{\|u\|\leq 1} \min_{i\in[n]} \{\langle u,\, \Phi_i \rangle + b_i\}\,. \tag{9}$$

By the Karush–Kuhn–Tucker (KKT) conditions, $u^\star$ admits the representation

$$u^\star = \sum_{i\in \mathrm{SV}} \alpha_i^\star\, \Phi_i, \qquad \alpha_i^\star \geq 0, \tag{10}$$

where $\mathrm{SV} := \{i : \alpha_i^\star > 0\}$ *is the set of support pairs.*

Remark 1 (Empirical observability of dual variables). The normalized time-integrated coefficients $\alpha_i(T) := A_i(T)/\|\Delta\theta(T)\|$ also converge to the dual variables $\alpha_i^\star$ as $T \to \infty$. This justifies tracking A_i as an empirical proxy for the support set in section 4.2.

3.2 Dual Sparsity of DPO Updates

Lemma 1 establishes that the DPO update direction is fully determined by the support pairs and the corresponding feature vectors. We now identify two distinct sources of sparsity in this representation.

Definition 3 (Dual Sparsity)**.** The DPO update direction $u^\star = \sum_{i\in\mathrm{SV}} \alpha_i^\star \Phi_i$ is said to exhibit *dual sparsity* if the following two conditions hold:

(i) **Data-level sparsity:** The number of support pairs is small relative to the dataset, i.e., $|\mathrm{SV}| \ll n$.

(ii) **Parameter-level sparsity:** Each gradient feature vector Φ_i is sparse, i.e., $\|\Phi_i\|_0 \ll p$ for $i \in \mathrm{SV}$.

The data-level sparsity is a direct consequence of the maximum-margin structure:

Lemma 2 (Data-Level Sparsity)**.** *Suppose $\pi_{\theta_0} = \pi_{\mathrm{ref}}$ (so that $b_i = 0$) and assumption 1 holds. Let $K := \Phi^\top \Phi \in \mathbb{R}^{n\times n}$, with $K_{ij} = \langle \Phi_i, \Phi_j \rangle$, be the Gram matrix of $\Phi := [\Phi_1 \;\; \Phi_2 \;\; \cdots \;\; \Phi_n]$, and let $\gamma_\star := \min_{i\in[n]} \langle u^\star, \Phi_i \rangle > 0$ be the maximum margin. Then, for the linearized model eq. (7),*

$$|\mathrm{SV}|\, \gamma_\star^2 \;\leq\; \lambda_{\max}(K_{\mathrm{SV,SV}}) \;\leq\; \lambda_{\max}(K), \qquad \textit{hence} \qquad |\mathrm{SV}| \;\leq\; \min\left\{ n,\; \left\lfloor \frac{\lambda_{\max}(K)}{\gamma_\star^2} \right\rfloor \right\}, \tag{11}$$

where $K_{\mathrm{SV,SV}}$ is the principal submatrix of K indexed by SV.

The proof is given in section A.1.

This bound is meaningful because, in the DPO setting, the gradient flow concentrates its cumulative weight $A_i(T)$ on the pairs that remain difficult throughout training—those near the margin boundary—while the remaining pairs receive diminishing gradient signal. Since the bound is governed by the ratio $\lambda_{\max}(K)/\gamma_\star^2$ rather than explicitly by n, the effective support set is provably $\ll n$ rather than merely $\leq \mathrm{rank}(\Phi)$ whenever the margin is large relative to the spectral scale of the features; we confirm empirically that this is the case on real preference data (see section 4.2).

For the parameter-level sparsity, we introduce the following assumption:

Assumption 2 (Sparsity of Gradient Features)**.** For each $i \in [n]$, the gradient feature vector Φ_i satisfies $\|\Phi_i\|_0 \leq s$ for some $s \ll p$.

Remark 2 (Motivation for assumption 2). In autoregressive language models, the preferred and dispreferred responses y_i^+ and y_i^- for a given prompt x_i typically share a substantial number of tokens—particularly in the initial portion of the response. Since Φ_i is defined as the difference of gradients $\beta(\nabla_\theta \log \pi_{\theta_0}(y_i^+ \mid x_i) - \nabla_\theta \log \pi_{\theta_0}(y_i^- \mid x_i))$, the parameter-level contributions from shared tokens cancel, leaving nonzero entries only in the parameters responsible for distinguishing between y_i^+ and y_i^-. This cancellation effect provides a structural basis for assumption 2. We verify this assumption empirically in section 4.

Combining lemmas 1 and 2 with assumption 2, we arrive at the main result:

Theorem 3 (Dual Sparsity of DPO)**.** *Under assumptions 1 and 2, the DPO update direction $u^\star$ exhibits dual sparsity in the sense of definition 3. That is, $u^\star$ is a sparse linear combination of sparse vectors:*

$$u^\star = \sum_{i \in \mathrm{SV}} \alpha_i^\star \, \Phi_i, \quad |\mathrm{SV}| \ll n, \quad \|\Phi_i\|_0 \le s \ll p. \tag{12}$$

Consequently, $u^\star$ is itself sparse with $\|u^\star\|_0 \le |\mathrm{SV}| \cdot s$.

3.3 Connection to Weight Disentanglement

Theorem 3 establishes that DPO updates exhibit dual sparsity. We now examine how this structure relates to the weight disentanglement framework reviewed in section 2.2. The original formulation of weight disentanglement (definition 2) is stated for a generic function $f\colon \mathcal{X} \times \Theta \to \mathcal{Y}$; applying it to the DPO setting requires identifying the appropriate input space and function in preference learning.

Triplet-space formulation. We take the input space to be the space of preference triplets $\tilde{\mathcal{X}} := \{(x, y^+, y^-)\}$ and define $f\colon \tilde{\mathcal{X}} \times \Theta \to \mathbb{R}$ as the DPO log-probability margin

$$f(\tilde{x}; \theta) := \beta\big(\log \pi_\theta(y^+ \mid x) - \log \pi_\theta(y^- \mid x)\big), \quad \tilde{x} = (x, y^+, y^-). \tag{13}$$

Differentiating at θ_0 recovers the gradient feature vector of section 3.1: $\nabla_\theta f(\tilde{x}; \theta_0) = \Phi(\tilde{x})$. The same features that drive dual sparsity therefore serve as the Jacobian relevant for weight disentanglement. A task t is associated with a preference dataset $\mathcal{D}_t \subseteq \tilde{\mathcal{X}}$ collected under task-specific annotation criteria.

Linearized decomposition. For task vectors $\{\tau_t\}_{t\in[T]}$ and coefficients $\{\lambda_t\}_{t\in[T]}$, a first-order Taylor expansion of f around θ_0 gives

$$f\left(\tilde{x};\, \theta_0 + \sum_{t=1}^{T} \lambda_t \tau_t\right) \;\approx\; f(\tilde{x}; \theta_0) + \sum_{t=1}^{T} \lambda_t \langle \tau_t,\, \Phi(\tilde{x}) \rangle. \tag{14}$$

Equation (14) matches the structural form of the weight disentanglement decomposition (definition 2) under the identification

$$g_0(\tilde{x}) := f(\tilde{x}; \theta_0), \qquad g_t(\tilde{x};\, \lambda_t \tau_t) := \lambda_t \langle \tau_t,\, \Phi(\tilde{x}) \rangle. \tag{15}$$

With this identification, the disentanglement condition $g_t(\tilde{x};\, \lambda_t \tau_t) = 0$ for $\tilde{x} \notin \mathcal{D}_t$ reduces to the inner-product condition

$$\langle \tau_t,\, \Phi(\tilde{x}) \rangle \;\approx\; 0 \quad \text{for } \tilde{x} \notin \mathcal{D}_t. \tag{16}$$

Observation 1 (Linearized weight disentanglement under DPO)**.** Under the NTK linearization, the DPO margin f admits the decomposition eq. (14), and the identification eq. (15) places this decomposition in the form of weight disentanglement (definition 2). The disentanglement condition holds to the extent that the cross-task inner product eq. (16) is small.

Connection to dual sparsity. Lemma 1 establishes that, in the limit, the normalized task vector for task t aligns with the maximum-margin direction $u_t^\star = \sum_{i \in \mathrm{SV}_t} \alpha_{t,i}^\star \, \Phi(\tilde{x}_i^{(t)})$. Substituting into eq. (15), the local function g_t takes the form

$$g_t(\tilde{x};\, \lambda_t \tau_t) \;\propto\; \langle u_t^\star,\, \Phi(\tilde{x}) \rangle \;=\; \Big\langle \sum_{i \in \mathrm{SV}_t} \alpha_{t,i}^\star \, \Phi(\tilde{x}_i^{(t)}),\; \Phi(\tilde{x}) \Big\rangle. \tag{17}$$

The cross-task interference eq. (16) is therefore governed by $\langle u_1^\star, \Phi(\tilde{x})\rangle$ for $\tilde{x} \in \mathcal{D}_2$. By theorem 3, $u_1^\star$ is a sparse combination of sparse vectors; assumption 2 ensures $\Phi(\tilde{x})$ is also sparse. When the active parameter supports of $u_1^\star$ and $\{\Phi(\tilde{x})\}_{\tilde{x}\in\mathcal{D}_2}$ have small overlap, this inner product is small. Whether the overlap is in fact small for naturally occurring preference tasks is an empirical question, which we examine in section 4.

Remark on the disjoint-support assumption. Definition 1 assumes disjoint task supports $\mathcal{D}_t \cap \mathcal{D}_{t'} = \emptyset$. In the triplet-space formulation this is naturally satisfied: distinct preference datasets arise from distinct annotation processes, and an ordered triplet (x, y^+, y^-) is tied to the task it was annotated under.

4 Experiments

We empirically validate the theoretical predictions of section 3. Specifically, we verify (i) data-level sparsity of the dual coefficients $\alpha_i^\star$ (lemma 2), (ii) parameter-level sparsity of the gradient features Φ_i (assumption 2), and (iii) suppression of cross-task interference under DPO compared to SFT (section 4.4).

4.1 Experimental Setup

Models. We use GPT-2 Small and GPT-2 Medium [Radford et al., 2019], together with three instruction-tuned models with 7–9B parameters: Llama-3-8B-Instruct [Grattafiori et al., 2024], Mistral-7B-Instruct-v0.2 [Jiang et al., 2023], and Gemma-2-9B-it [Gemma Team, 2024]. Each checkpoint serves as both the initialization and the reference policy, $\pi_{\theta_0} = \pi_{\text{ref}}$.

Datasets. For our primary experiments, we use the Anthropic HH-RLHF dataset [Bai et al., 2022], which consists of two subsets: *helpful* (evaluating which response is more useful to the user) and *harmless* (evaluating which response avoids harmful content). These two distinct subsets serve as the tasks $\mathcal{T}_1$ and $\mathcal{T}_2$ in our cross-task interference analysis. To assess generality, we additionally use the UltraFeedback Binarized dataset [Cui et al., 2024], a large-scale preference dataset annotated by GPT-4 across four evaluation axes (instruction-following, truthfulness, honesty, and helpfulness).

Gradient features. The gradient feature vectors $\Phi_i = \beta(\nabla_\theta \log \pi_{\theta_0}(y_i^+ \mid x_i) - \nabla_\theta \log \pi_{\theta_0}(y_i^- \mid x_i))$ are computed empirically using the Jacobian of the model at initialization, since the analytical NTK cannot be obtained in closed form for self-attention architectures. Because storing full-dimensional gradient vectors ($\Phi_i \in \mathbb{R}^p$ with $p > 10^8$) is memory-intensive, experiments that require Φ_i are performed on subsamples of the datasets (see details in section B).

The models, datasets, and numbers of preference pairs used in each experiment, together with the computation of the time-integrated weights A_i, are detailed in section B.

4.2 Data-Level Sparsity

Lemma 2 predicts that the DPO update direction is driven by a small number of support pairs, with the remaining pairs contributing negligibly. To verify this, we examine how the time-integrated weights $A_i(T)$ are distributed across pairs as training progresses.

Figure 1 visualizes A_i for GPT-2 Small and Medium on $N = 100$ pairs from HH-RLHF at 1, 100, and 1000 epochs. At early stages, the weights are distributed nearly uniformly. As training progresses, the distribution becomes increasingly polarized: a small subset of pairs accumulates large weights while the rest stop growing (see section C for the corresponding log-probabilities). This is the behavior predicted by lemma 1, under which the normalized coefficients $\alpha_i(T)$ converge to the dual variables $\alpha_i^\star$, which are nonzero only for the support pairs.

Table 1 quantifies this concentration on $N = 1000$ pairs by the top-10% mass, i.e., the fraction of $\sum_i A_i$ held by the top 10% of pairs (10% under a uniform distribution). For GPT-2, the top-10% mass grows from about 11% to 40–44% over 1000 epochs; for Llama-3-8B-Instruct and Mistral-7B-Instruct-v0.2, it grows from about 11% to 44–52% within three epochs of full fine-tuning DPO. The numerical evaluation of the bound in lemma 2 in the setting of fig. 1 is reported in section A.2.

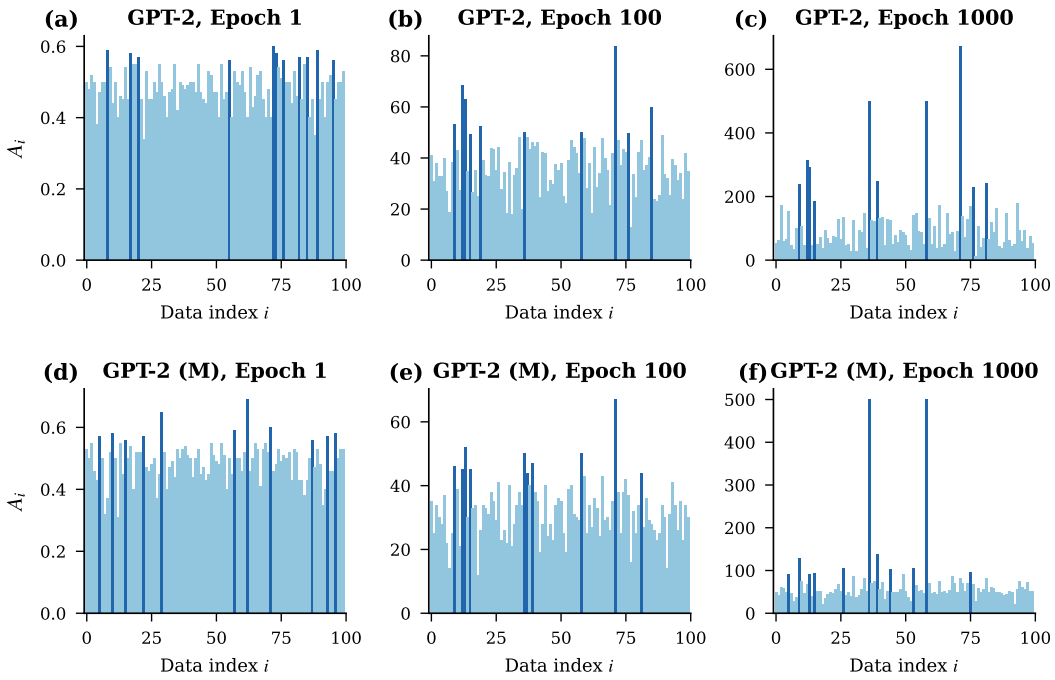

Figure 1: Distribution of A_i across data pairs for GPT-2 Small (top) and Medium (bottom) at 1 epoch (left), 100 epochs (middle), and 1000 epochs (right) on HH-RLHF. Dark bars indicate the top 10% of pairs by weight. As training progresses, the distribution polarizes onto a small number of support pairs; this effect is more pronounced in the larger model.

Table 1: Top-10% mass of A_i on $N = 1000$ pairs, i.e., the fraction of $\sum_i A_i$ held by the top 10% of pairs (10% under a uniform distribution). Top: GPT-2 trained with DPO on HH-RLHF. Bottom: 7–8B models trained with full fine-tuning DPO on UltraFeedback Binarized.

Model	**Dataset**	**Top-10% mass (%)**		
		Epoch 1	Epoch 100	Epoch 1000
GPT-2 Small	HH-RLHF	10.88	28.78	39.57
GPT-2 Medium	HH-RLHF	11.02	32.32	43.70
		Epoch 1	Epoch 2	Epoch 3
Llama-3-8B-Instruct	UltraFeedback	11.75	41.82	52.48
Mistral-7B-Instruct-v0.2	UltraFeedback	11.23	39.92	43.77

4.3 Parameter-Level Sparsity

Assumption 2 posits that the DPO gradient feature vectors Φ_i are sparse due to gradient cancellation between preferred and dispreferred responses. To verify this and to contrast it with SFT, fig. 2 visualizes the element-wise distribution of a representative Φ_i under both methods (GPT-2 Medium, HH-RLHF, $N = 100$), and table 2 quantifies the fraction of near-zero entries for all five models ($N = 1000$, UltraFeedback Binarized).

Under DPO (fig. 2a), the vast majority of entries are concentrated near zero, with only a small fraction taking on appreciable magnitude. Under SFT (fig. 2b), the distribution is substantially broader, reflecting the fact that the SFT feature vector $\Phi_i^{\mathrm{SFT}} = \nabla_\theta \log \pi_\theta(y_i^+ \mid x_i)$ involves only the preferred response and does not benefit from the gradient cancellation that arises in the DPO difference $\nabla_\theta \log \pi_{\theta_0}(y_i^+ \mid x_i) - \nabla_\theta \log \pi_{\theta_0}(y_i^- \mid x_i)$. The same contrast holds at every scale we test (table 2): from 124M to 9B parameters, more than 90% of the entries of the DPO feature vector are near zero, whereas only 4–20% of those of the SFT feature vector are.

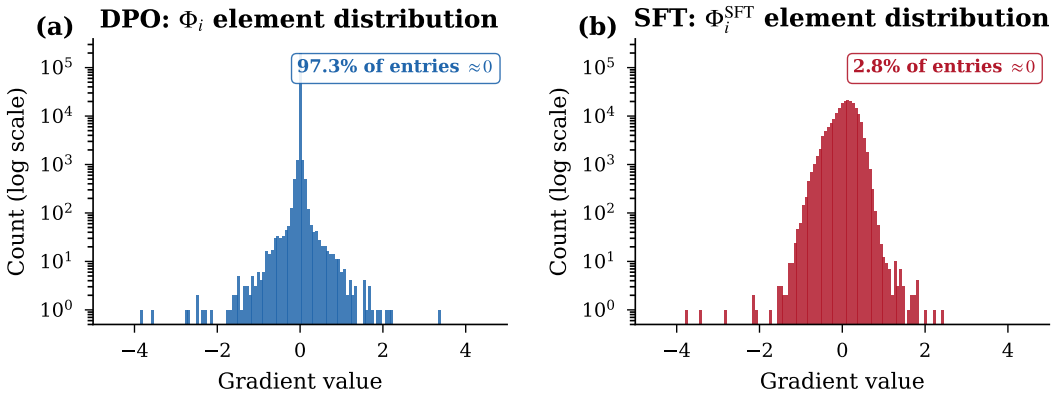

Figure 2: Element-wise distribution of gradient feature vectors on a logarithmic scale for GPT-2 Medium on $N = 100$ pairs from HH-RLHF. **(a)** DPO: the vast majority of entries are near zero, confirming assumption 2. **(b)** SFT: the distribution is substantially denser, since no gradient cancellation occurs. Near-zero fractions for all five models, computed in a separate setting, are reported in table 2.

Table 2: Fraction of near-zero entries in a representative gradient feature vector under DPO (Φ_i) and SFT (Φ_i^{SFT}), computed from 1,000 pairs sampled from UltraFeedback Binarized. Higher values indicate sparser features.

Model	**Params**	**DPO Φ_i (%)**	**SFT Φ_i^{SFT} (%)**
GPT-2 Small	124M	91.29	3.86
GPT-2 Medium	355M	94.82	4.34
Llama-3-8B-Instruct	8B	92.51	16.44
Mistral-7B-Instruct-v0.2	7B	90.93	19.72
Gemma-2-9B-it	9B	92.12	13.57

As discussed in remark 2, this contrast arises naturally from the structure of preference data in autoregressive models: since y_i^+ and y_i^- typically share a large number of tokens—particularly in the initial portion of the response—the parameter-level gradients for these shared tokens cancel in the DPO feature vector, leaving nonzero contributions only for the parameters that distinguish between the two responses.

4.4 Cross-Task Interference

We evaluate whether the dual sparsity of DPO updates translates into reduced cross-task interference, as discussed in section 3.3. We consider two task pairs of increasing semantic distance:

(a) **HH helpful ↔ harmless**: two subsets of the same dataset (Anthropic HH-RLHF) with different annotation criteria—helpfulness vs. harmlessness.

(b) **HH helpful ↔ UltraFeedback**: two entirely different datasets with different annotation methods (human vs. AI), different prompt distributions, and different preference dimensions.

For the cross-dataset pair (b), only the gradient features $\Phi_j^{(2)}$ of UltraFeedback at θ_0 are required; no DPO training on UltraFeedback is performed.

Baseline. We compare DPO against SFT. Unlike DPO, SFT does not produce the logistic loss structure that gives rise to implicit maximum-margin bias; consequently, all data pairs contribute to the update direction, and data-level sparsity does not emerge. Moreover, the SFT feature vector $\Phi_i^{\text{SFT}} = \nabla_\theta \log \pi_\theta(y_i^+ \mid x_i)$ involves only the preferred response, so it does not benefit from the gradient cancellation that induces parameter-level sparsity in DPO.

Table 3: Cross-task interference for DPO and SFT across two model scales (GPT-2 Small/Medium) and two task pairs: *HH help↔harm* (HH-RLHF helpful vs. harmless subsets) and *HH help↔UF* (HH-RLHF helpful vs. UltraFeedback). Lower values indicate greater disentanglement. Across all configurations, DPO reduces the interference $\langle u_1^\star, V_2 \rangle$ by $11\times$ to $35\times$ relative to SFT.

Task Pair	Method	Model	Jaccard (%)	$\langle u_1^\star, V_2 \rangle$
HH help↔harm	SFT	Small	54.9	9.12
	DPO	Small	18.9	0.258
	SFT	Medium	49.3	3.22
	DPO	Medium	11.4	0.191
HH help↔UF	SFT	Small	68.4	21.2
	DPO	Small	25.2	1.91
	SFT	Medium	60.5	19.8
	DPO	Medium	17.7	0.852

Metrics. The disentanglement condition derived in section 3.3 is a per-triplet condition $\langle \tau_t,\, \Phi(\tilde{x}) \rangle \approx 0$ for $\tilde{x} \notin \mathcal{D}_t$ (eq. (16)). Verifying it directly would require evaluating $\langle u_1^\star,\, \Phi(\tilde{x}) \rangle$ for every $\tilde{x} \in \mathcal{D}_2$, which is computationally prohibitive given the scale of $\Phi(\tilde{x}) \in \mathbb{R}^p$ with $p > 10^8$. We therefore aggregate the per-triplet quantity into a single scalar summary by introducing the mean feature vector for task t:

$$V_t := \frac{1}{|\mathcal{D}_t|} \sum_{\tilde{x} \in \mathcal{D}_t} \Phi(\tilde{x}). \tag{18}$$

On this aggregated quantity, we evaluate two measures of cross-task overlap between V_1 and V_2:

- **Top-1% Jaccard overlap**: the Jaccard coefficient between the index sets of the top 1% largest-magnitude entries in each mean feature vector. This captures the degree of overlap between the most active parameters of the two tasks—the structural condition under which the per-triplet inner product $\langle u_1^\star,\, \Phi(\tilde{x}) \rangle$ is small (section 3.3).
- **Interference** $\langle u_1^\star,\, V_2 \rangle$: the inner product between the DPO update direction for $\mathcal{T}_1$ and the mean feature vector for $\mathcal{T}_2$. This is the average of $\langle u_1^\star,\, \Phi(\tilde{x}) \rangle$ over $\tilde{x} \in \mathcal{D}_2$ and serves as a tractable summary of the per-triplet disentanglement condition eq. (16).

Results. Table 3 reports the results. DPO suppresses cross-task interference by an order of magnitude or more in every configuration. Two qualitative patterns are notable: under DPO, both the Jaccard overlap and the interference decrease at the larger scale (e.g., $0.258 \to 0.191$ on the within-dataset pair and $1.91 \to 0.852$ on the cross-dataset pair), consistent with the sharper polarization of A_i observed in section 4.2; and the same pattern holds for the cross-dataset pair (HH helpful $\leftrightarrow$ UltraFeedback), demonstrating that the effect generalizes beyond subsets of a single dataset to entirely independent preference data sources. These observations empirically support the framework of section 3.3: DPO's dual sparsity, combined with the small parameter-space overlap exhibited by semantically distinct preference tasks, yields cross-task inner products an order of magnitude smaller than those under SFT.

Model merging. To examine whether reduced interference carries over to merged models, we merge task vectors obtained by DPO or SFT on each task pair as $\theta_{\text{merged}} = \theta_0 + \alpha(\tau_{\text{target}} + \tau_{\text{other}})$, $\alpha \in [0, 2]$ and measure the degradation Δ of each task's performance relative to its single-task model (section D). As shown in table 4, DPO yields a smaller Δ than SFT for both tasks in both task pairs, consistent with the reduced interference in table 3.

5 Related Work

Optimization dynamics of DPO. Since Rafailov et al. [2023] introduced DPO as a stable alternative to reinforcement learning from human feedback, several works have analyzed its dynamics theoretically. One line modifies the loss to address specific failure modes—IPO [Gheshlaghi Azar et al., 2024] avoids the Bradley–Terry assumption, and KTO [Ethayarajh et al., 2024] adopts a prospect-theoretic utility—but these are orthogonal to our analysis, which concerns the original

Table 4: Performance degradation Δ (single-task $-$ merged; lower is better) under task-vector merging on GPT-2 Medium. Task 1 is HH helpful; Task 2 is HH harmless or UltraFeedback. Full results are reported in section D.

Task Pair	Method	Δ Task 1	Δ Task 2
HH help↔harm	SFT	0.110	0.102
	DPO	**0.027**	**0.026**
HH help↔UF	SFT	0.096	0.030
	DPO	**0.036**	**0.014**

objective. Closer to our approach, Ren and Sutherland [2025] linearize DPO under the NTK regime and use the resulting framework to characterize the *squeezing effect*, a counter-intuitive decrease in the probability of preferred responses also studied by Razin et al. [2025] under the term *likelihood displacement*. We adopt the same NTK lens but extract a structurally distinct property: the *direction* of the update in parameter space. Combining the classical implicit-bias result of Soudry et al. [2018] for separable logistic regression with the specific geometry of preference-pair gradients yields dual sparsity (theorem 3), which in turn provides a theoretical explanation for the small-subnetwork behavior observed empirically by Mukherjee et al. [2025] on RL-based alignment. Likelihood displacement and dual sparsity are thus two complementary consequences of the same dynamics: one concerning the *probabilities* the update changes, the other the *parameters* it changes.

Task arithmetic, weight disentanglement, and model merging. Ilharco et al. [2023] showed that linear arithmetic on task vectors compositionally steers model behavior, and Ortiz-Jiménez et al. [2023] formalized weight disentanglement as a sufficient condition for this property, identifying via NTK tangent-space analysis that task vectors should act on disjoint regions of parameter space. A complementary line of work—DARE [Yu et al., 2024] and TIES-Merging [Yadav et al., 2023]—engineers sparsity into task vectors as a *post-hoc* intervention to suppress interference during merging. Our framework offers a different perspective: under DPO, sparsity emerges *intrinsically* from the optimization dynamics, and the resulting parameter-space overlap between semantically distinct preference tasks is small enough that the disentanglement condition holds approximately without any explicit intervention (section 4.4). Empirically, merging independently preference-optimized models has been reported to be competitive with or superior to data mixing, both for balancing safety and general-purpose capabilities [Aakanksha et al., 2024] and for balancing helpfulness, honesty, and harmlessness [Yang et al., 2025]. Our analysis provides theoretical support for these empirical findings, attributing the effectiveness of merging DPO-trained models to the dual sparsity of their updates and the resulting small cross-task interference. Our merging experiments (section 4.4) show that this translates into smaller performance degradation than SFT under task-vector merging; a larger-scale study of merging is left to future work.

6 Conclusion

We have presented a theoretical analysis of why DPO induces sparse parameter updates. Under the NTK linearization, the logistic structure of the DPO loss drives the update direction toward a maximum-margin solution that exhibits dual sparsity: it is determined by a small number of support pairs, and the gradient feature vector of each pair is itself sparse due to gradient cancellation between preferred and dispreferred responses. Relating this structure to weight disentanglement, we showed empirically that DPO reduces cross-task interference by $11\times$ to $35\times$ relative to SFT and degrades less under model merging, and that both sources of sparsity hold from GPT-2 to 7–9B models.

Limitations. Our analysis relies on the NTK linearization, which is not guaranteed to hold throughout DPO training of large models. More fundamentally, parameter-level sparsity (assumption 2) is an empirically motivated assumption rather than a consequence derived from the architecture. It relies on token overlap between the preferred and dispreferred responses: for pairs whose responses are entirely disjoint (e.g., different topics or formats), the gradient cancellation does not occur, and this source of sparsity may weaken or vanish. A formal derivation of assumption 2, and a characterization of the degree of overlap it requires in practice, are left to future work.

References

Aakanksha, Arash Ahmadian, Seraphina Goldfarb-Tarrant, Beyza Ermis, Marzieh Fadaee, and Sara Hooker. Mix data or merge models? optimizing for diverse multi-task learning. *arXiv preprint arXiv:2410.10801*, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DaSilva, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, et al. Ultrafeedback: Boosting language models with scaled ai feedback. In *International Conference on Machine Learning*, pages 9722–9744. PMLR, 2024.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *International Conference on Machine Learning*, 2024.

Gemma Team. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 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. In *International Conference on Artificial Intelligence and Statistics*, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, volume 31, pages 8571–8580, 2018.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, et al. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Sagnik Mukherjee, Lifan Yuan, Dilek Hakkani-Tur, and Hao Peng. Reinforcement learning finetunes small subnetworks in large language models. In *Advances in Neural Information Processing Systems*, 2025.

Guillermo Ortiz-Jiménez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models. In *Advances in Neural Information Processing Systems*, volume 36, pages 66727–66754, 2023.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 2019.

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 *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741, 2023.

Noam Razin, Sadhika Malladi, Adithya Bhaskar, Danqi Chen, Sanjeev Arora, and Boris Hanin. Unintentional unalignment: Likelihood displacement in direct preference optimization. In *International Conference on Learning Representations*, 2025.

Yi Ren and Danica J. Sutherland. Learning dynamics of LLM finetuning. In *International Conference on Learning Representations*, 2025.

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19(70):1–57, 2018.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-Merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Jinluan Yang, Dingnan Jin, Anke Tang, Li Shen, Didi Zhu, Zhengyu Chen, Ziyu Zhao, Daixin Wang, Qing Cui, Zhiqiang Zhang, Jun Zhou, Fei Wu, and Kun Kuang. Mix data or merge models? balancing the helpfulness, honesty, and harmlessness of large language model via model merging. In *Advances in Neural Information Processing Systems*, volume 38, pages 124703–124732, 2025.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

A Proof and Numerical Evaluation of Lemma 2

A.1 Proof

Throughout, we work with the frozen-NTK surrogate of the margin, $m_i^{\mathrm{lin}}(\theta) := m_i(\theta_0) + \langle \Phi_i, \theta - \theta_0 \rangle$, under which the DPO loss reduces exactly to the logistic regression eq. (7); lemma 2 holds exactly for this model. Under $\pi_{\theta_0} = \pi_{\mathrm{ref}}$, the offsets vanish, $b_i = m_i(\theta_0) = 0$, so that $u^\star$ solves $\max_{\|u\|\le 1} \min_{i\in[n]} \langle u, \Phi_i \rangle$. By assumption 1, the optimal value $\gamma_\star$ is positive, and hence the constraint is active: $\|u^\star\| = 1$.

KKT conditions. The problem is equivalent to the hard-margin problem $w^\star = \arg\min_w \frac{1}{2}\|w\|^2$ subject to $\langle w, \Phi_i \rangle \ge 1$ for all i, with $u^\star = w^\star/\|w^\star\|$ and $\gamma_\star = 1/\|w^\star\|$. Its KKT conditions give $w^\star = \sum_i \hat{\alpha}_i \Phi_i$ with $\hat{\alpha}_i \ge 0$ and $\hat{\alpha}_i(\langle w^\star, \Phi_i \rangle - 1) = 0$. Thus $\alpha_i^\star = \hat{\alpha}_i/\|w^\star\|$ in eq. (10), and complementary slackness yields

$$\langle u^\star, \Phi_i \rangle = \gamma_\star \qquad \text{for all } i \in \mathrm{SV}. \tag{19}$$

Bound. Let $S := \mathrm{SV}$, $s := |S|$, and $\mathbf{1}_S \in \mathbb{R}^s$ the all-ones vector. Summing eq. (19) over S and applying the Cauchy–Schwarz inequality with $\|u^\star\| = 1$,

$$s\,\gamma_\star = \Big\langle u^\star, \sum_{i\in S} \Phi_i \Big\rangle \le \Big\| \sum_{i\in S} \Phi_i \Big\| = \big(\mathbf{1}_S^\top K_{S,S}\, \mathbf{1}_S\big)^{1/2}. \tag{20}$$

By the Rayleigh quotient, $\mathbf{1}_S^\top K_{S,S} \mathbf{1}_S \le \lambda_{\max}(K_{S,S})\,\|\mathbf{1}_S\|^2 = s\,\lambda_{\max}(K_{S,S})$. Combining the two, $s^2\gamma_\star^2 \le s\,\lambda_{\max}(K_{S,S})$, i.e., $s\,\gamma_\star^2 \le \lambda_{\max}(K_{S,S})$. Since $K_{S,S}$ is a principal submatrix of the positive semidefinite matrix K, Cauchy's interlacing theorem gives $\lambda_{\max}(K_{S,S}) \le \lambda_{\max}(K)$. Finally, since s is an integer with $s \le n$, we obtain $s \le \min\{n, \lfloor \lambda_{\max}(K)/\gamma_\star^2 \rfloor\}$. □

A.2 Numerical Evaluation

Computing $\gamma_\star$ exactly requires solving the max-margin problem. We instead evaluate the bound through the finite-time margin of the normalized trajectory, which is justified by the following corollary.

Corollary 4 (Finite-time evaluation of the bound)**.** *Under the assumptions of lemma 2, let $\Delta\theta(T) \neq 0$ and define the finite-time margin*

$$\tilde{\gamma}_T := \min_{i\in[n]} \left\langle \frac{\Delta\theta(T)}{\|\Delta\theta(T)\|}, \Phi_i \right\rangle. \tag{21}$$

Then $\tilde{\gamma}_T \le \gamma_\star$. If, moreover, $\Delta\theta(T)$ follows the gradient flow of the frozen-NTK surrogate, then $\tilde{\gamma}_T \to \gamma_\star$ as $T \to \infty$. Consequently, whenever $\tilde{\gamma}_T > 0$,

$$|\mathrm{SV}| \;\le\; \min\left\{n,\; \left\lfloor \frac{\lambda_{\max}(K)}{\gamma_\star^2} \right\rfloor \right\} \;\le\; \min\left\{n,\; \left\lfloor \frac{\lambda_{\max}(K)}{\tilde{\gamma}_T^2} \right\rfloor \right\}. \tag{22}$$

Proof. Let $u_T := \Delta\theta(T)/\|\Delta\theta(T)\|$. Since $\|u_T\| = 1$, u_T is feasible for the max-margin problem, and hence $\tilde{\gamma}_T = \min_i \langle u_T, \Phi_i \rangle \leq \max_{\|u\|\leq 1} \min_i \langle u, \Phi_i \rangle = \gamma_\star$. Under the frozen-NTK gradient flow, lemma 1 gives $u_T \to u^\star$; since $u \mapsto \min_i \langle u, \Phi_i \rangle$ is continuous as the minimum of finitely many continuous functions, $\tilde{\gamma}_T \to \min_i \langle u^\star, \Phi_i \rangle = \gamma_\star$. Finally, if $\tilde{\gamma}_T > 0$, then $1/\gamma_\star^2 \leq 1/\tilde{\gamma}_T^2$, which, combined with eq. (11), $|\mathrm{SV}| \leq n$, and the integrality of $|\mathrm{SV}|$, yields the chain of inequalities. □

The values reported below are computed from the actual training trajectory of the network rather than from the frozen-NTK gradient flow. For such a trajectory, the convergence statement of corollary 4 does not apply, but the inequality $\tilde{\gamma}_T \leq \gamma_\star$ still holds, since it uses only $\|u_T\| = 1$; the reported ratio $\lambda_{\max}(K)/\tilde{\gamma}_T^2$ is therefore a conservative (upper) estimate of $\lambda_{\max}(K)/\gamma_\star^2$. Table 5 reports this quantity in the setting of fig. 1 (GPT-2 Small/Medium, HH-RLHF, $n = 100$). For both models it is about $1/7$ to $1/9$ of n, guaranteeing that $|\mathrm{SV}|$ is roughly an order of magnitude smaller than n, consistent with the concentration observed in section 4.2.

Table 5: Conservative evaluation of the support-set bound eq. (11) in the setting of fig. 1 ($n = 100$).

Model	$\lambda_{\max}(K)/\tilde{\gamma}_T^2$	**Implied bound on** $\lvert\mathrm{SV}\rvert$
GPT-2 Small	14.58	≤ 14
GPT-2 Medium	11.33	≤ 11

B Experimental Details

Time-integrated weights. The weights $A_i(T) = \int_0^T \sigma(-m_i(\theta(t)))\, dt$ are approximated by accumulating the discrete-time analogue $\sum_t \sigma(-m_i(\theta_t))$ over training steps. Since A_i depends only on the margins, it requires no gradient features.

Experimental settings. Table 6 summarizes the models, datasets, and numbers of preference pairs used in each experiment. The full fine-tuning DPO runs for the 7–8B models use up to eight NVIDIA A100 GPUs.

Table 6: Experimental settings. "Pairs" is the number of preference pairs on which each quantity is computed. HH: Anthropic HH-RLHF; UF: UltraFeedback Binarized; ep.: epochs.

Experiment	**Models**	**Dataset**	**Pairs**	**Training**
Data-level (fig. 1)	GPT-2 S/M	HH	100	DPO, 1000 ep.
Data-level (table 1)	GPT-2 S/M	HH	1000	DPO, 1000 ep.
	Llama-3-8B	UF	1000	Full-FT DPO, 3 ep.
	Mistral-7B	UF	1000	Full-FT DPO, 3 ep.
Parameter-level (fig. 2)	GPT-2 M	HH	100	–
Parameter-level (table 2)	All five	UF	1000	–
Interference (table 3)	GPT-2 S/M	HH, UF	100	DPO / SFT
Model merging (table 4)	GPT-2 M	HH, UF	section D	DPO / SFT

C Evolution of Log-Probabilities During DPO Training

We record the log-probabilities of the preferred and dispreferred responses along the training trajectory that produces fig. 1 (GPT-2 Medium, HH-RLHF, $N = 100$). Each value is the sequence-level log-probability averaged over the 100 pairs, with logps/chosen $= \log \pi_\theta(y^+ \mid x)$ and logps/rejected $= \log \pi_\theta(y^- \mid x)$.

As shown in table 7, both log-probabilities decrease, but the rejected log-probability decreases considerably faster, so the gap widens from 11.7 to 128.6. Since $\pi_{\theta_0} = \pi_{\mathrm{ref}}$, the margin satisfies $m_i(\theta) = \beta\,(\mathrm{gap}_i(\theta) - \mathrm{gap}_i(\theta_0))$, where gap_i denotes the per-pair log-probability gap; a widening gap thus directly increases the margin. As a result, $\sigma(-m_i) \to 0$ for easy pairs, which stops their contribution to A_i and produces the polarization of A_i observed in fig. 1.

Table 7: Average log-probabilities of preferred (chosen) and dispreferred (rejected) responses during DPO training (GPT-2 Medium, HH-RLHF, $N = 100$).

Epoch	logps/chosen	logps/rejected	Gap (chosen − rejected)
1	−247.2	−258.9	11.7
100	−280.5	−334.5	54.0
1000	−308.3	−436.9	128.6

D Model Merging Experiments

Setup. For each task, we fine-tune GPT-2 Medium with DPO or SFT until convergence and take the task vector $\tau = \theta^\star - \theta_0$ from the best checkpoint. For each target task, we form $\theta_{\text{merged}} = \theta_0 + \alpha\,(\tau_{\text{target}} + \tau_{\text{other}})$, where $\alpha \in \{0, 0.1, \ldots, 2.0\}$ is selected on a validation set, and report $\Delta = (\text{single-task}) - (\text{merged})$. We use the same two task pairs as in section 4.4.

Results. Table 8 reports the single-task and merged performance. On both task pairs, DPO degrades less than SFT under merging. The gap between the two methods is smaller than the order-of-magnitude reduction in the cross-task inner product (table 3), since merged performance also depends on factors other than parameter-space overlap.

Table 8: Single-task and merged performance of GPT-2 under task-vector merging. Δ is the single-task minus merged performance (lower is better).

		Single		Merged		Δ	
Task Pair	**Method**	Task 1	Task 2	Task 1	Task 2	Task 1	Task 2
HH help↔harm	SFT	0.631	0.625	0.521	0.523	0.110	0.102
	DPO	0.668	0.645	0.641	0.619	**0.027**	**0.026**
HH help↔UF	SFT	0.631	0.535	0.518	0.505	0.096	0.030
	DPO	0.668	0.572	0.632	0.558	**0.036**	**0.014**

E Licenses

Table 9 lists the existing assets used in this paper together with their licenses and the versions (Hugging Face identifiers) we use.

Table 9: Existing assets, versions, and licenses.

Asset	Version (Hugging Face)	License
GPT-2 Small	`openai-community/gpt2`	Modified MIT
GPT-2 Medium	`openai-community/gpt2-medium`	Modified MIT
Llama-3-8B-Instruct	`meta-llama/Meta-Llama-3-8B-Instruct`	Llama 3 Community License
Mistral-7B-Instruct-v0.2	`mistralai/Mistral-7B-Instruct-v0.2`	Apache 2.0
Gemma-2-9B-it	`google/gemma-2-9b-it`	Gemma Terms of Use
Anthropic HH-RLHF	`Anthropic/hh-rlhf`	MIT
UltraFeedback Binarized	`HuggingFaceH4/ultrafeedback_binarized`	MIT

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state three contributions (C1: a dual sparsity theorem, C2: a framework relating dual sparsity to weight disentanglement, C3: empirical validation), each of which is supported by theorem 3 in section 3.2, the framework in section 3.3, and the experiments in section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: A dedicated "Limitations" paragraph in section 6 discusses (i) the NTK linearization underlying the analysis and (ii) the empirical (rather than derived) status of assumption 2, including its failure mode for preference pairs whose responses share no tokens.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All theorems, lemmas, and assumptions are numbered and cross-referenced. Assumption 1 (linear separability) and assumption 2 (gradient-feature sparsity) are stated explicitly. Lemma 1 follows from the implicit-bias result of Soudry et al. [2018], which is properly cited. Lemma 2 is stated with its assumptions ($\pi_{\theta_0} = \pi_{\mathrm{ref}}$, assumption 1, and the NTK linearization) in section 3.2 and proved in full in section A.1, and theorem 3 is obtained by combining lemmas 1 and 2 with assumption 2.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 4.1 and section B specify the models (GPT-2 Small/Medium, Llama-3-8B-Instruct, Mistral-7B-Instruct-v0.2, Gemma-2-9B-it), the datasets (Anthropic HH-RLHF, UltraFeedback Binarized), the empirical NTK gradient-feature construction, the number of preference pairs and training budget used in each experiment (table 6), the discrete-time approximation of $A_i(T)$, and the cross-task interference metrics (Top-1% Jaccard, $\langle u_1^\star, V_2 \rangle$). Anonymized code reproducing all experiments is released as supplementary material.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Anonymized code that reproduces the main experiments is provided as supplementary material with a README describing dependencies and run commands. Both datasets used (Anthropic HH-RLHF, UltraFeedback Binarized) and all base models (GPT-2 Small/Medium, Llama-3-8B-Instruct, Mistral-7B-Instruct-v0.2, Gemma-2-9B-it) are publicly available.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4.1 and section B report the training-relevant information needed to interpret the figures and tables, namely the models, datasets, the number of preference pairs and epoch budgets used in each experiment (table 6), and the metric definitions in section 4.4. Remaining low-level details (optimizer, learning rate, batch size, gradient accumulation, random seeds) are fully specified in the released code.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Error bars are not reported. Computing each gradient feature vector $\Phi_i \in \mathbb{R}^p$ with $p > 10^8$ is memory-intensive (section 4.1), making multi-seed evaluation prohibitive at the scale of these experiments. The qualitative trends are nonetheless consistent across five models from 124M to 9B parameters (table 2), four models in the data-level analysis (table 1), multiple training durations (fig. 1), and two task pairs (table 3), which provides indirect evidence of robustness.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: The GPU type is reported in section B: the full fine-tuning DPO runs for the 7–8B models use up to eight NVIDIA A100 GPUs, while the GPT-2 Small (124M) and Medium (355M) experiments are runnable on a single modern GPU. Memory footprint and wall-clock time are not reported.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: The work uses publicly available datasets and pre-trained models, involves no human subjects or scraped data, and releases no model with elevated misuse risk. The submission preserves anonymity and conforms to the NeurIPS Code of Ethics.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: The paper presents a theoretical analysis of the optimization dynamics of DPO under an NTK linearization, together with controlled experiments on publicly available models (GPT-2 and 7–9B instruction-tuned models). As foundational research not tied to any specific application or deployment, there is no direct path from these results to societal impact.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper releases no pre-trained model, scraped dataset, or other asset with elevated misuse risk. Only analysis code accompanying the experiments is released.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All existing assets used in the paper are cited at first mention: GPT-2 [Radford et al., 2019], Llama-3-8B-Instruct [Grattafiori et al., 2024], Mistral-7B-Instruct-v0.2 [Jiang et al., 2023], Gemma-2-9B-it [Gemma Team, 2024], Anthropic HH-RLHF [Bai et al., 2022], and UltraFeedback Binarized [Cui et al., 2024]. Their licenses and versions are documented in section E, and all assets are used in accordance with their terms of use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The only new asset is anonymized analysis code released as supplementary material, accompanied by a README that documents dependencies, data preparation, and the commands needed to reproduce each figure and table.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The research does not involve crowdsourcing or human subjects. Human preference annotations enter only via the publicly released HH-RLHF dataset, which is used as-is.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The research does not involve human subjects, so IRB approval (or equivalent) is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs appear in the paper only as the object of study (GPT-2, Llama-3-8B-Instruct, Mistral-7B-Instruct-v0.2, and Gemma-2-9B-it as base policies π_{θ_0}) and as part of an existing dataset (GPT-4 as annotator for UltraFeedback, used as released). They are not an original or non-standard component of the proposed method, so no declaration is required under the NeurIPS LLM policy.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.