ENGINEERING Artificial Intelligence
https://doi.org/10.2738/ENGAI.2026.0001

Original Research

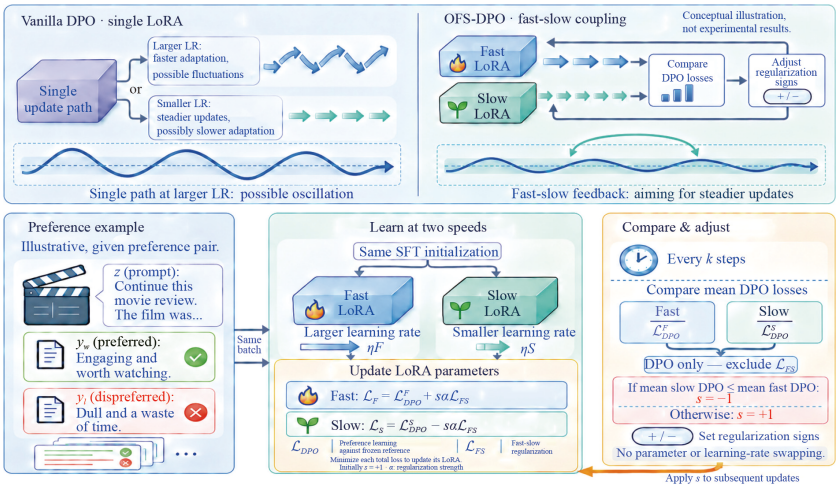
Online DPO: Online Direct Preference Optimization with Fast-Slow Chasing

Biqing Qi[1,2], Junqi Gao[3], Pengfei Li[3], Fangyuan Li[4], Zhixuan Su[5], Kaiyan Zhang[6], Bowen Zhou (✉)[1,6]

1 Shanghai AI Laboratory, Shanghai 201114, China
2 Institute of Data Science, The University of Hong Kong, Hong Kong 999077, China
3 School of Mathematics, Harbin Institute of Technology, Harbin 150001, China
4 Shanghai Innovation Institute, Shanghai 201114, China
5 School of Astronautics, Harbin Institute of Technology, Harbin 150001, China
6 Department of Electronic Engineering, Tsinghua University, Beijing 100084, China

© The Author(s) 2026. This article is published by Higher Education Press.

GRAPHICAL ABSTRACT



HIGHLIGHTS

- OFS-DPO uses fast and slow LoRA modules with a regularization term that guides their preference-probability gap during online alignment.
- COFS-DPO linearly combines task-specific fast LoRA modules to retain previously learned preferences during cross-domain continual alignment.
- Regret and gradient analyses characterize the fast-slow competition mechanism and its effect on online learning.
- Experiments show that OFS-DPO outperforms DPO in in-domain alignment and that COFS-DPO improves cross-domain continual learning.

ABSTRACT Direct preference optimization (DPO) improves the alignment of large language models (LLMs) with human values by training directly on human preference datasets, eliminating the need for reward models. However, due to the presence of cross-domain human preferences, direct continual training can lead to catastrophic forgetting, limiting DPO's performance and efficiency. Inspired by intraspecific competition driving species evolution, we propose an Online Fast-Slow Chasing DPO (OFS-DPO) for preference alignment, simulating competition through fast and slow chasing among models to facilitate rapid adaptation. Specifically, we first derive the regret upper bound for online learning, validating our motivation with a min-max optimization pattern. Based on this, we introduce two identical modules using Low-Rank Adaptation (LoRA) with different optimization speeds to simulate intraspecific competition, and propose a new regularization term to guide their learning. To further mitigate catastrophic forgetting in cross-domain scenarios, we extend OFS-DPO with a LoRA module

Received 1 Jun 2026; accepted 9 Sep 2026

E-mail: zhoubowen@tsinghua.edu.cn (Zhou B.).

combination strategy, resulting in the Cross-domain Online Fast-Slow Chasing DPO (COFS-DPO). This method leverages linear combinations of fast module parameters from different task domains, fully utilizing historical information to achieve continual value alignment. Experimental results show that OFS-DPO outperforms DPO in in-domain alignment, while COFS-DPO excels in cross-domain continual learning scenarios.

KEYWORDS Direct preference optimization; Large language models; Online learning; Catastrophic forgetting; Continual learning

1 Introduction

To better align large language models (LLMs) with human values and prevent harmful responses, reinforcement learning from human feedback (RLHF) is commonly used to fine-tune LLMs [1–3]. However, the complexity of the RLHF training process and its dependence on reward models limit its practicality and efficiency [3]. To address these issues, Direct Preference Optimization (DPO) [4] has emerged as an efficient alternative. DPO directly applies supervised training to preference data, eliminating the need for complex reinforcement-learning frameworks or explicit reward models [5], thereby simplifying preference-alignment optimization and improving training efficiency.

However, DPO is designed for supervised training on local offline data and does not adapt well to online preference data streams [5]. Additionally, DPO cannot leverage historical information, leading to catastrophic forgetting of the original task domain during cross-domain preference alignment and resulting in overall performance degradation [6,7]. The cycle of forgetting and relearning significantly increases resource consumption, including computational costs and the need for re-collecting annotated data, making DPO disadvantageous in resource-constrained scenarios.

Current online learning methods for human preference alignment either involve constructing reward models, which increases resource consumption (e.g., continual proximal policy optimization (CPPO) [8]), or rely on feedback generated by LLMs, which lacks a flexible modular design to ensure efficient learning and memory retention [5,9]. In this paper, inspired by the intraspecific competition theory [10,11], we design competitive components with consistent optimization objectives and integrate them into online learning to enhance the model's ability to adapt to continual changes. Our method retains the resource-efficient characteristics of DPO while improving its capability to handle continuously incoming streams of preference data.

To incorporate the concept of intraspecific competition into online preference alignment learning, we first derive the regret bounds for online learning methods [12,13]. We discover that these bounds include a min-max term similar to the objective function in Generative Adversarial Networks (GANs) [14]. The key difference is that, in our case, the min and max terms share the same optimization objective, closely reflecting the intraspecific competition observed in nature [10,11], thus validating our motivation. Specifically, the min-max terms in the upper-bound decomposition motivate the fast-slow design, whereas the subsequent lower-bound analysis characterizes its effect on the unavoidable regret component.

Furthermore, to maintain consistency with the original DPO method's adherence to human values and to prevent the policy model from significantly deviating from the reference model, we retain the objective function of the original DPO. Building on this foundation, we instantiate fast and slow modules using LoRA [15] and introduce a regularization term to measure the preference probability gap between the fast and slow modules, guiding the learning of these modules. Consequently, we propose the Online Fast-Slow Chasing DPO (OFS-DPO) for in-domain tasks. We further characterize the empirical-regret lower bound of OFS-DPO, analyze the gradient effect of the fast-slow regularization term, and experimentally validate the practical effectiveness of OFS-DPO.

To extend OFS-DPO to cross-domain preference alignment environments, we propose the Cross-domain Online Fast-Slow Chasing DPO (COFS-DPO). Specifically, using OFS-DPO, we derive the optimal fast modules for two task domains and maintain domain-specific memories. Drawing inspiration from the human brain's capacity [16–18] for continual learning via the interplay of modular memories, our method encapsulates this mechanism through a combination of LoRAs. Additionally, inspired by the conclusion of the equivalence between data shift and model parameter shift [19], we derive a cross-domain regret lower bound to clarify how the fast-slow mode affects the regret expression. In experimental evaluations, our proposed OFS-DPO outperforms DPO and other competitive methods in in-domain scenarios, including controlled sentiment generation, summarization, and single-turn dialogue tasks. In cross-domain scenarios, we demonstrate that our proposed COFS-DPO significantly surpasses competitive baselines in the summarization task. In summary, our contributions are as follows:

- We propose OFS-DPO, a simple and effective method based on fast-slow LoRA modules from

the novel perspective of intraspecific competition. This method introduces a regularization term to measure and guide the preference probability gap between the modules.

- To extend OFS-DPO to cross-domain scenarios, we propose COFS-DPO, which jointly optimizes the linear combination of the optimal fast modules from different tasks. This method achieves performance comparable to theoretically optimal model parameters across the entire task domain while maintaining strong memory retention in individual domains.
- We provide theoretical analyses of the regret expressions and the gradient effect of the fast-slow regularization term. The regret upper-bound decomposition motivates the fast-slow design, while the lower-bound results characterize the regret expressions under the stated assumptions. The effectiveness of OFS-DPO and COFS-DPO is validated through experiments.

2 Related Works

2.1 Direct preference optimization

DPO provides a resource-efficient alternative to reinforcement learning from human feedback and has been applied to a wide range of downstream tasks. For instance, in the multimodal domain, DPO has been beneficial for tasks such as text-to-image generation using diffusion models [20], text-to-action generation [21], text-to-audio conversion [22], video instruction following [23], and translation tasks leveraging LLMs [24]. However, DPO still has limitations that constrain its practical utility. Consequently, various improvement strategies rooted in DPO have emerged. For example, the multi-objective direct preference optimization (MODPO) [25] is proposed to address the requirements of multiple alignment objectives by balancing their weights. Additionally, the DPO-positive (DPOP) model [26] introduces an enhanced DPO objective function to mitigate accuracy degradation in preference datasets with smaller edit distances, thereby improving DPO's performance on specific tasks. Inspired by these advancements, we develop the online versions of the DPO, OFS-DPO, and COFS-DPO methods.

2.2 Continual learning

In continual learning within in-domain tasks, the need for swift adaptation to dynamic data streams has been emphasized [27]. In contrast, continual learning in cross-domain scenarios faces the significant challenge of preserving previously learned task features while accommodating new ones to prevent catastrophic forgetting [28]. Current strategies to address these challenges fall into several categories: regularization-based methods, replay-based techniques, and domain generalization methods [28,29]. Regularization-based methods [6,30–33] incorporate regularization terms to balance the integration of new and old knowledge, assessing the significance of various features. Replay-based methods [34–36] mitigate forgetting in cross-domain scenarios by retaining past data or experiences, but consequently require additional memory. In addition, domain-generalization research [37–39] aims to identify feature representations that generalize beyond the training distribution while maintaining satisfactory performance on current tasks. However, these methods typically struggle when faced with substantial shifts in cross-domain distributions [38,39].

3 Methodology

3.1 Preliminaries

In a standard online setting, we consider the distribution of data corresponding to specific human preference tasks as $\mathcal{D}$, updated within T time steps. In cross-domain scenarios, we differentiate different tasks as $\mathcal{D}_1, T_1$ and $\mathcal{D}_2, T_2$. The sequence $(x_1, x_2, x_3, \ldots, x_T)$ represents samples within T time steps. Each data point in our task setup consists of three parts: $x_i = (z, y_w, y_l)$, where z is the prompt statement for the task, and y_w and y_l represent the desired and undesired model preferences given z, respectively. To ensure a fair comparison, we adhere to the settings established in [8], concentrating on two domain configurations. Let $\mathcal{H}$ be the hypothesis class defined on distribution $\mathcal{D}$, with $h_i \in \mathcal{H}$ denoting a hypothesis function belonging to the hypothesis class at the i-th time step. The model parameter is denoted by θ, with θ_i representing the model parameters at the i-th time step. The objective function of the DPO is denoted by $l(\theta, x)$. The main notation used in the theoretical analysis and the proposed methods is summarized in Table 1.

3.2 Motivation and theoretical analysis

To better understand the feasibility of the intraspecific competition motivation, we first conduct a theoretical analysis of the difference between online learning methods and the offline optimal decision by the regret definition [40]. We begin by precisely defining online

Table 1 Summary of the main notation used in the theoretical analysis and the proposed methods.

Symbol	Description
$\mathcal{D}$	Distribution of the online preference-data sequence
$\mathcal{D}_k$	Preference-data distribution of the k-th task domain
x_i and $s_i^{(k)}$	Sample observed at time i in the in-domain setting and sample observed at time i from domain k, respectively
$\mathcal{H}$	Original hypothesis class used to define the offline optimal decision
$\mathcal{H}'$ and $\mathcal{H}''$	Intermediate hypothesis classes used in the expected-regret upper-bound decomposition
θ_i	Trainable model or LoRA parameters at online step i
θ^F and θ^S	Trainable parameters of the F-module and S-module, respectively
$l(\theta, x)$	DPO-related loss evaluated using parameters θ and sample x
$\mathcal{L}_{DPO}$	Original DPO objective function
$\mathcal{L}_{DPO-FS}$	Fast-slow regularization term measuring the relative preference difference between the F-module and S-module
α	Coefficient of the fast-slow regularization term
β	Scaling coefficient used in the DPO objective
η_F and η_S	Learning rates of the F-module and S-module, respectively
ρ	Fast-slow learning-rate multiplier satisfying $\eta_F = \rho\eta_S$
k	Period used to compare the two module losses and determine whether their roles should be switched
M_1 and M_2	Retained domain-specific sample sets for Task 1 and Task 2
β_1 and β_2	Coefficients used to combine the task-specific LoRA modules in COFS-DPO

expected regret to prevent any conceptual ambiguities.

Definition 3.2.1 (Expected regret)

$$R(T) = \mathbb{E}_{x\sim\mathcal{D}}\frac{1}{T}\Big[\sum_{t=1}^{T} l(h_t, x_t) - \min_{h\in\mathcal{H}}\sum_{t=1}^{T} l(h, x_t)\Big]. \tag{1}$$

Here, $\mathcal{D}$ represents the distribution of task sequences in online learning, namely, $\mathcal{D} = (\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_T)$, and $x = (x_1, x_2, \ldots, x_T) \sim \mathcal{D}$. Similar to the regret upper-bound discussion in [41], we present the following lemma and derive the regret upper bound.

Lemma 3.2.1

In online learning methods, there exists a regret upper bound that includes a minimax term:

$$\begin{aligned} R(T) \leqslant\ & O\Big(\sqrt{\ln(|\mathcal{H}'|)/T}\Big) \\ & + \mathbb{E}_{x\sim\mathcal{D}}\frac{1}{T}\left[\min_{h'\in\mathcal{H}'}\max_{h\in\mathcal{H}}\left(\sum_{t=1}^{T} l(h'_t, x_t) - \sum_{t=1}^{T} l(h, x_t)\right)\right], \end{aligned} \tag{2}$$

where the first term is the regret against the best $h' \in \mathcal{H}'$ and $\mathcal{H}'$ is an infinite hypothesis class to approximate $\mathcal{H}$, so the second term captures how well $\mathcal{H}'$ approximates $\mathcal{H}$.

The detailed proof of Lemma 3.2.1 is provided in Section A.1 of the Supplementary Material. Let $h \in \mathcal{H}$ denote the optimal hypothesis in the hypothesis space. Introducing another class $\mathcal{H}''$ to approximate $\mathcal{H}$ yields the following inequality:

$$\begin{aligned} R(T) \leqslant\ & O\Big(\sqrt{\ln(|\mathcal{H}'|)/T}\Big) \\ & + \mathbb{E}_{x\sim\mathcal{D}}\frac{1}{T}\left[\min_{h'\in\mathcal{H}'}\max_{h''\in\mathcal{H}''}\left(\sum_{t=1}^{T} l(h'_t, x_t) - \sum_{t=1}^{T} l(h'', x_t)\right)\right] \\ & + \mathbb{E}_{x\sim\mathcal{D}}\frac{1}{T}\left[\min_{h''\in\mathcal{H}''}\max_{h\in\mathcal{H}}\left(\sum_{t=1}^{T} l(h''_t, x_t) - \sum_{t=1}^{T} l(h, x_t)\right)\right]. \end{aligned} \tag{3}$$

According to Eq. (3), the first term depends only weakly on the learning method. We therefore further constrain the expected regret by minimizing the two min-max terms on the right-hand side. The min-max structure resembles competitive interaction within natural populations, where the two hypotheses interact through their cumulative losses. To approximate h' and h'', respectively, we introduce a fast module and a slow module. The two modules pursue each other while approximating the best offline decision h [40], thereby

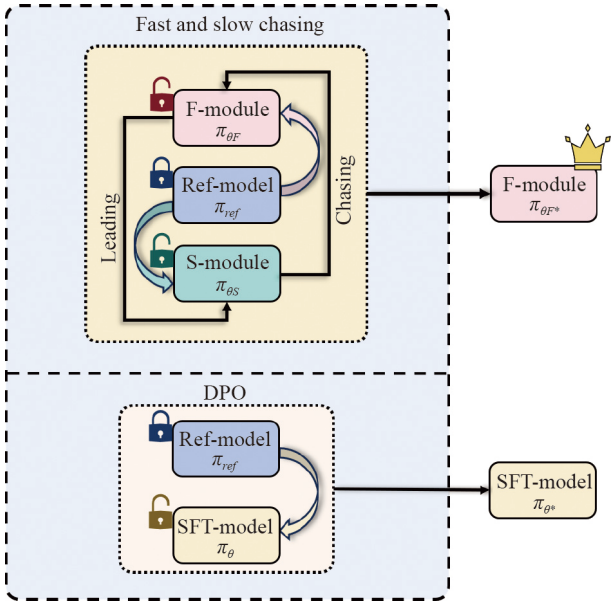


Fig. 1 Framework of OFS-DPO. The upper panel shows the F-module and S-module adaptively adjusting during training while the reference model remains fixed. The lower panel shows the original DPO framework.

targeting the min-max terms, as illustrated in Fig. 1.

3.3 In-domain: Online Fast-Slow Chasing DPO

(1) OFS-DPO objective function. Based on the above analysis, we introduce fast and slow modules to better adapt to the evolving data stream. However, the original DPO objective [4] does not describe the interaction between these modules, so additional module-specific objectives are required. In essence, our goal remains consistent with DPO: aligning the policy with human preferences while limiting deviation from the reference model. We therefore retain the DPO objective as the principal component of the proposed objectives. The DPO objective function measures the optimization gap between the learning model and the reference model. To adapt it to our new setup, we replace π_θ in the DPO objective function with π_{θ^F} and π_{θ^S} for the fast and slow modules, respectively, and denote these modules as F-module and S-module for convenience. By constructing the regularization term that measures the preference probability gap between the fast and slow modules, we induce a chasing interaction between them to promote efficient optimization. Specifically, we first define the fast-slow regularization term as follows:

$$\mathcal{L}_{DPO-FS} = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_{\theta^F}(y_w|x)}{\pi_{\theta^S}(y_w|x)}-\beta\log\frac{\pi_{\theta^F}(y_l|x)}{\pi_{\theta^S}(y_l|x)}\right)\right]. \quad (4)$$

Using this regularization term, we formulate the objectives of the fast and slow modules as follows:

$$\mathcal{L}_{DPO-new}(\theta^F) = \mathcal{L}_{DPO}(\theta^F) + \alpha\mathcal{L}_{DPO-FS}, \quad (5)$$

$$\mathcal{L}_{DPO-new}(\theta^S) = \mathcal{L}_{DPO}(\theta^S) - \alpha\mathcal{L}_{DPO-FS}. \quad (6)$$

In the training phase, we optimize Eqs. (5) and (6) using distinct learning rates for the fast and slow modules. Here,

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

where π denotes the preference policy probability function, and $\sigma(\cdot)$ represents the logistic function. We have $\mathcal{L}_{DPO} \in (a,b) = (0, \ln 2)$. The coefficient $\alpha \in (0,1)$ is the regularization term coefficient, and θ^F and θ^S represent the parameters of the F-module and S-module, respectively. From the perspective of relative preference optimization, $\mathcal{L}_{DPO-FS}$ compares the preference strengths of the F-module and S-module. Assigning opposite signs to this term encourages the F-module to enlarge the difference and the S-module to reduce it, creating a chasing interaction while both modules remain guided by their respective standard DPO objectives. Combined with different learning rates and periodic sign adjustment based on the DPO losses, this interaction produces the proposed Fast-Slow Chasing mechanism.

The corresponding gradients become:

$$g_t^F = \nabla_{\theta^F}\mathcal{L}_{DPO-new}(\theta^F); \quad g_t^S = \nabla_{\theta^S}\mathcal{L}_{DPO-new}(\theta^S). \quad (8)$$

(2) OFS-DPO. Based on the above analysis, we proceed to formally construct OFS-DPO. Specifically, we incorporate two LoRA modules into the reference model and initialize the F-module and S-module, respectively. Using the update gradients defined in Eq. (8), we update their parameters at each online training step. Every k time steps, we compare their average DPO losses. If $\mathcal{L}_{\text{DPO}}(\theta^F) > \mathcal{L}_{\text{DPO}}(\theta^S)$, we reverse the signs of the $\mathcal{L}_{\text{DPO-FS}}$ terms so that the better-performing module assumes the F-module role in the next cycle. Otherwise, the current assignment is maintained. This role adjustment, together with the different learning rates, enables the proposed Fast-Slow Chasing mechanism. The method is summarized as Algorithm 1 in the Supplementary Material.

For the instantiation of our method, we implement the two modules using LoRA [15], a widely used parameter-efficient fine-tuning strategy for LLMs. Its modular structure allows the two modules to be optimized independently while sharing the same frozen backbone, without duplicating the full base model. LoRA therefore satisfies our requirements while mitigating the additional cost of introducing two modules.

(3) Theoretical analysis of OFS-DPO. We next derive an empirical-regret lower bound to characterize how the fast-slow mode enters the regret expression in in-domain tasks. We first present the empirical definition of regret.

Definition 3.3.1 (Empirical regret)

$$R(T) = \frac{1}{T}\sum_{i=1}^{T}[l(\theta_T, x_i) - l(\theta^*, x_i)], \quad (9)$$

where $\theta^* \triangleq \operatorname{argmin}_\theta \frac{1}{T}\sum_{i=1}^{T} l(\theta, x_i)$, $x_i \sim \mathcal{D}$, $i \in \{1,2,\ldots,T\}$, and $\mathcal{D}$ is the data distribution.

At regular update intervals, we compare $\mathcal{L}_{DPO}(\theta^F)$ and $\mathcal{L}_{DPO}(\theta^S)$ so that the better-performing module assumes the F-module role. For convenience, let θ and w denote the F-module and S-module parameters, respectively. Accordingly, the optimal module parameters at each time step satisfy

$$\hat{\theta}_i = \operatorname*{argmin}_{v\in\{\theta_i, w_i\}} l(v, x_i). \quad (10)$$

Continuing, we can express the empirical regret of the OFS-DPO in the following form:

$$\hat{R}(T)=\sum_{i=1}^{T}\frac{1}{T}[l(\hat{\theta}_T,x_i)-l(\theta^*,x_i)]. \tag{11}$$

Before analyzing the empirical regret in Eq. (11), we introduce boundedness assumptions for the gradients and task-specific module parameters.

Assumption 3.3.1 (Gradient boundedness)
Let θ_i denote the trainable parameters at the i-th step, while the parameters of the backbone model remain frozen. Denote $g_i=\nabla_\theta l(\theta_i,x_i)$, $i=1,2,\ldots,T$. Then $\|g_i\|_2\leqslant G$, where G is a positive constant.

Assumption 3.3.2 (Trainable parameter boundedness)
Let θ_i and w_i denote the trainable parameters of the F-module and S-module, respectively. Suppose $\|\theta_n-\theta_m\|_2\leqslant d$ and $\|w_n-w_m\|_2\leqslant d$ for all $n,m\in\{1,\ldots,T\}$, where d is a positive constant.

These assumptions are restricted to the trainable parameters over the finite training horizon considered in this work. This setting is consistent with the bounded-gradient and bounded-iterate conditions commonly used in optimization analysis [42,43], as well as with the local-update nature of LoRA fine-tuning [15,44]. Their validity under our experimental setting is further examined in Section 4.4.

Under Assumptions 3.3.1 and 3.3.2, we derive the following theorem.

Theorem 3.3.1 Within the proposed OFS-DPO, a lower empirical regret bound can be attained, with probability $1-\delta$, where $\delta=2(T-1)\delta_0-(T-1)(2T-3)(\delta_0)^2[1-\delta_0]^{2T-4}$, and $\delta_0\in(0,1)$.

$$\begin{aligned}R(T)\geqslant\ & l(\theta_1,x_1)-\frac{1}{T}\sum_{i=1}^{T}l(\theta^*,x_i)\\ &-\left[2-\frac{1}{T}+\left(1-\frac{1}{T}\right)\mathbb{1}_{\{\text{mode}=FS\}}\right]Gd\\ &-2\left(1-\frac{1}{T}\right)\ln 2\sqrt{-\frac{\ln\delta_0}{2}},\end{aligned} \tag{12}$$

where $\mathbb{1}_{\{\text{mode}=FS\}}$ represents whether to introduce fast and slow modules.

From Theorem 3.3.1, we observe that, when the fast-slow mode is introduced, i.e., $\mathbb{1}_{\{\text{mode}=FS\}}=1$, the right-hand side of Inequality 12 decreases by $(1-\frac{1}{T})Gd$. This reduction indicates that the fast-slow mode can further reduce the inherent regret component induced by the conventional update scheme. More detailed derivations can be found in Section A.2 of the Supplementary Material.

(4) More stable gradient. We further analyze the effect of the $\mathcal{L}_{DPO-FS}$ regularization term on gradient behavior through the following proposition.

Proposition 3.3.1
As training progresses, for any $\epsilon>0$, the original DPO gradient can satisfy $\|\nabla_\theta\mathcal{L}_{DPO}(\theta)\|_2<\epsilon$, while the fast-module objective retains $\|\nabla_{\theta^F}\mathcal{L}_{DPO-new}(\theta^F)\|_2>\epsilon$ under the conditions stated in the proof. In other words, the original DPO update signal can diminish during training, whereas the $\mathcal{L}_{DPO-FS}$ regularization term provides an additional gradient component.

The above proposition provides a mechanism-level explanation of how the $\mathcal{L}_{DPO-FS}$ term provides an additional gradient update signal. This gradient analysis complements the preceding regret analysis.

3.4 Cross-domain: Online Fast-Slow Chasing DPO

(1) COFS-DPO. We further extend OFS-DPO to cross-domain scenarios. Unlike the in-domain setting, COFS-DPO must balance information obtained from different task domains. It therefore preserves and integrates domain-specific information to mitigate catastrophic forgetting. Inspired by the human brain's ability [16–18] to achieve continual learning through the interaction of modular memories, we model this process using the combination of LoRAs. This design enables COFS-DPO to retain important information across tasks, as illustrated in Fig. 2.

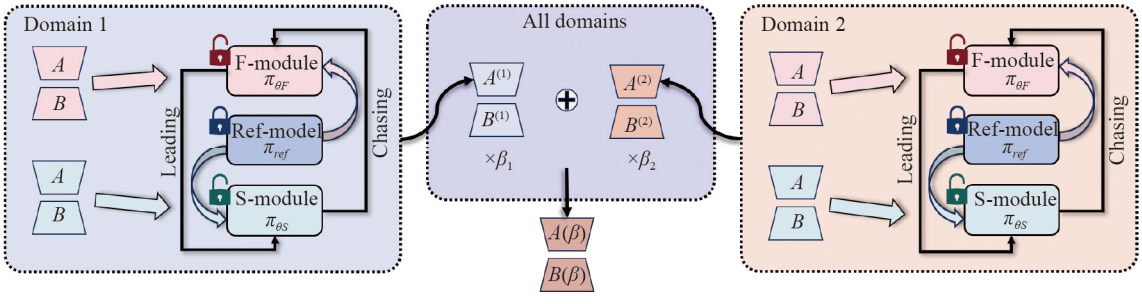


Fig. 2 Framework of COFS-DPO. OFS-DPO is applied separately to each task domain to obtain a task-specific F-module. Subsequently, we seek the optimal linear combination (β_1,β_2) across all task domains to construct the final model.

Specifically, in the cross domain scenario, we first use the OFS-DPO to obtain the final F-modules for two task domains, retaining a random subset of domain-specific memories M_1 and M_2. Subsequently, we compute the optimal combination of F-modules over the joint memory distribution (M_1, M_2) to achieve the best performance. The detailed procedure is outlined in Algorithm 2 in the Supplementary Material. After each task, COFS-DPO stores the task-specific LoRA module together with a small memory set. The retained samples are used to determine the module-combination weights, while the LoRA modules preserve the information learned from individual tasks. Because these modules share the same frozen backbone, they can be stored and combined independently without duplicating the full base model. Other parameter-efficient fine-tuning methods could also be used if they provide similar module independence and composability.

Although the current formulation and experiments focus on two task domains, the modular procedure can be extended to K domains. Specifically, OFS-DPO can be applied to each domain $\mathcal{D}_k$ to obtain a task-specific F-module θ_k^F and retain a small domain-specific memory set M_k. The final model can then be constructed by selecting the combination coefficients over the union of the retained samples and combining the task-specific modules as $\theta(\beta) = \sum_{k=1}^{K} \beta_k \theta_k^F$. This extension follows the same principle as the two-domain formulation. However, as the number of domains increases, the storage of task-specific modules, the allocation of retained samples, and the selection of combination coefficients become more complex. The current experiments validate only the two-domain setting, and empirical evaluation with more domains is left for future work.

(2) Theoretical analysis of COFS-DPO. Next, we derive a cross-domain regret lower bound for COFS-DPO to examine how the fast-slow mode affects its regret expression.

To ensure clarity, we first standardize the use of symbols: let $s_i^{(k)}$ represent the sample at the i-th moment from distribution $\mathcal{D}_k$, where $k = 1, 2$ and $i = 0, 1, 2, \ldots, T_k$.

Building on these definitions and the previously established single-task regret lower bound in Theorem 3.3.1, we derive Theorem 3.4.1 for the dual-task setting.

Definition 3.4.1 (Empirical regret of cross-domain tasks: dual-task regret)

$$R(T_1, T_2) = \frac{1}{T_1}\sum_{i=1}^{T_1} l(\theta_{T_1}, s_i^{(1)}) + \frac{1}{T_2}\sum_{j=1}^{T_2} l(\theta_{T_2}, s_j^{(2)}) - \frac{1}{T_1}\sum_{i=1}^{T_1} l(\theta^*, s_i^{(1)}) - \frac{1}{T_2}\sum_{j=1}^{T_2} l(\theta^*, s_j^{(2)}). \tag{13}$$

Given the definition of dual-task regret, we derive a lower regret bound similar to the in-domain case.

Theorem 3.4.1

Under the settings of this paper, the dual-task regret also has a lower empirical regret bound.

$$R(T_1, T_2) \geqslant l_1(s_1^{(1)}, s_1^{(2)}) - B(T_1, T_2) - 2\left(1 - \frac{T_1 + T_2}{T_1 T_2}\right)c - [6 - \frac{T_1 + T_2}{T_1 T_2} + \left(2 - \frac{T_1 + T_2}{T_1 T_2}\right)\mathbb{1}_{\{mode=FS\}}]Gd, \tag{14}$$

where $B(T_1, T_2) = \frac{1}{T_1}\sum_{i=1}^{T_1} l(\theta^{(1)}, s_i^{(1)}) + \frac{1}{T_2}\sum_{j=1}^{T_2} l(\theta^{(2)}, s_j^{(2)})$, $l_1(s_1^{(1)}, s_1^{(2)}) = l(\theta_1, s_1^{(1)}) + l(\theta_2, s_1^{(2)})$, $c = \max\{\ln 2\sqrt{-\frac{\ln \delta_1}{2}}, \ln 2\sqrt{-\frac{\ln \delta_2}{2}}\}$, where $\delta_1, \delta_2 \in (0, 1)$, and $\mathbb{1}_{\{mode=FS\}}$ indicates whether fast-slow modules are used.

Theorem 3.4.1 shows that, when the fast-slow mode is introduced, the right-hand side of Inequality 14 is reduced by $\left(2 - \frac{T_1 + T_2}{T_1 T_2}\right)Gd$. This indicates that COFS-DPO, like OFS-DPO, also achieves a lower empirical regret bound. Under the stated assumptions, this reduction reflects a decrease in the intrinsic problem difficulty captured by the derived cross-domain lower-bound expression.

4 Experiments

4.1 Experimental settings

For the online optimization protocol, the F-module and S-module are initialized with identical parameters from the same SFT model. At each online training step, both modules use the current batch of preference data and are updated according to the gradients defined in Eq. (8). Their different optimization speeds are implemented through distinct learning rates: the F-module uses the larger learning rate and responds more rapidly to new preference data, while the S-module provides a more slowly changing reference. In the main experiments, we set $\eta_F = \rho\eta_S$, with $\rho = 2$ and $\eta_S = 5 \times 10^{-7}$.

The experiments comprise two parts. The first evaluates the ability of OFS-DPO to adapt to online preference streams within a single domain, whereas the second evaluates the ability of COFS-DPO to retain previously acquired knowledge across domains.

(1) In-domain task setting. For in-domain online preference alignment, we follow the DPO experimental setting [4] and evaluate our method on three tasks: controlled sentiment generation, summarization, and single-turn dialogue. For the controlled sentiment

generation task, we use the large movie review dataset (IMDB) [45]. Initially, we fine-tune GPT-2 [46] on the IMDB training dataset to obtain the supervised fine-tuning (SFT) model [47]. Subsequently, using positive and negative reviews from the dataset as preference data, we further fine-tune the SFT model using OFS-DPO to align it with human value preferences.

In the summarization task, we use the TL;DR Summarize dataset [1] with human preferences. We fine-tune GPT-J [48] on the summarization dataset as the SFT model and then conduct online value preference alignment using our method.

For single-turn dialogue, we fine-tune the Pythia-2.8B model Helpful and Harmless dialogue dataset as the SFT model. Subsequently, we train the SFT model with OFS-DPO.

We compare our method with proximal policy optimization (PPO), DPO, and the corresponding SFT model for each task under identical training settings. Specifically, consistent with previous studies [4,5,8,49], we use GPT-4 as a surrogate human evaluator to assess the quality of model-generated content and compare it with preferences extracted from authentic datasets. The resulting win rate serves as a metric to quantify the effectiveness of model alignment. More detailed experimental configurations are provided in the Supplementary Material.

(2) Cross-domain task setting. Based on the experimental setup in CPPO [8], we design our cross-domain experiment named "Summary". We utilize two datasets: the Reddit TL; DR dataset for SFT and the human preference dataset provided by CarperAI for RLHF. To validate our method's continual learning capability, we partition these datasets into two domains based on post types: "relationships" and "others". We denote the "relationships" domain as Task-1, comprising a single category, while the "others" domain is denoted as Task-2, consisting of the remaining 28 categories. To ensure a fair comparison with CPPO, we adopt the same experimental settings. We train a GPT-2 Small (GPT2-s) model with 124M parameters using Task-1 data from the Reddit TL;DR dataset for 5 epochs as the SFT model. Additionally, we train the LLaMA3 (8B) model [50] for further testing. We fine-tune the SFT model combined with LoRA on the Task-1 data of the human preference dataset to obtain $\theta^{(1)}$. Then, we fine-tune it on the Task-2 data after initializing the model to obtain $\theta^{(2)}$. We retain a small amount of data from different tasks to provide COFS-DPO for the combined optimization of $\Delta\theta^{(1)}$ and $\Delta\theta^{(2)}$.

To evaluate alignment performance, consistent with previous work [8,51], we fine-tune GPT-J (6.7B) on the complete human-preference dataset as a reference preference model (rPM). We use recall-oriented understudy for gisting evaluation (ROUGE) [52] and rPM scores (rPMS) [8] to measure alignment with the current data, and the score forgettig ratio (SFR) metric [8] to measure forgetting on the previous task. The evaluation-metrics table in the Supplementary Material presents the evaluation metrics for each task.

4.2 Evaluation results on in-domain tasks

Table 2 presents the performance of OFS-DPO compared with traditional DPO, PPO, and SFT models under an online in-domain data stream. Using GPT-4 as a surrogate human evaluator [4,5,8,49], we evaluate the model-generated content against the corresponding preference data. The results show that OFS-DPO consistently outperforms DPO and PPO. Specifically, in the controlled sentiment generation task, OFS-DPO improves the win rate by approximately 8% across the evaluated LoRA ranks. In the single-turn dialogue task, it improves the win rate by approximately 5%. These results demonstrate the alignment effectiveness of OFS-DPO across different tasks.

We evaluate LoRA ranks 16 and 256 to compare a relatively lightweight adapter setting with a higher-capacity setting. OFS-DPO achieves consistent improvements at both ranks, showing that the observed gains are not limited to a single tested adapter capacity.

4.3 Evaluation results on cross-domain tasks

Table 3 presents the continual-learning results for the

Table 2 GPT-4 win rates for three in-domain tasks.

LoRA rank	Method	Controlled sentiment generation	Summarization	Single-turn Dialogue
16	SFT	0.307 ± 0.042	0.128 ± 0.036	0.132 ± 0.033
	DPO	0.488 ± 0.037	0.194 ± 0.014	0.192 ± 0.012
	PPO	0.358 ± 0.056	0.136 ± 0.048	0.147 ± 0.052
	OFS-DPO (Ours)	**0.568 ± 0.034**	**0.223 ± 0.033**	**0.231 ± 0.014**
256	SFT	0.313 ± 0.035	0.138 ± 0.036	0.138 ± 0.032
	DPO	0.494 ± 0.032	0.211 ± 0.028	0.203 ± 0.024
	PPO	0.364 ± 0.039	0.148 ± 0.044	0.155 ± 0.042
	OFS-DPO (Ours)	**0.571 ± 0.043**	**0.234 ± 0.021**	**0.253 ± 0.017**

Note: we use LoRA ranks 16 and 256 to examine the effect of adapter rank on fine-tuning performance. Results are reported over five trials. Additional implementation details are provided in the Supplementary Material.

Table 3 Main results for the cross-domain tasks.

Model	LoRA rank	Method	Task-1		Final				
			$rPMS_1(\uparrow)$	$ROUGE_1(\uparrow)$	$rPMS_1(\uparrow)$	$ROUGE_1(\uparrow)$	$SFR(\downarrow)$	$rPMS_2(\uparrow)$	$ROUGE_2(\uparrow)$
GPT2-s	16	DPO [4]	5.750 ± 0.124	0.228 ± 0.009	5.773 ± 0.132	0.230 ± 0.008	-0.023 ± 0.012	4.785 ± 0.121	0.167 ± 0.019
		PPO+EWC [6]	4.932 ± 0.117	0.203 ± 0.011	4.983 ± 0.108	0.207 ± 0.014	-0.051 ± 0.014	4.505 ± 0.118	0.137 ± 0.014
		PPO+LwF [32]	4.890 ± 0.122	0.199 ± 0.021	4.953 ± 0.118	0.202 ± 0.010	-0.063 ± 0.004	4.533 ± 0.114	0.128 ± 0.008
		PPO+TFCL [53]	4.934 ± 0.148	0.217 ± 0.018	4.988 ± 0.220	0.217 ± 0.013	-0.054 ± 0.010	4.524 ± 0.113	0.135 ± 0.012
		PC[54]	4.811 ± 0.210	0.204 ± 0.031	4.845 ± 0.164	0.216 ± 0.008	-0.034 ± 0.013	4.574 ± 0.120	0.149 ± 0.011
		HN-PPO [55]	4.945 ± 0.151	0.218 ± 0.013	4.992 ± 0.211	0.204 ± 0.011	-0.047 ± 0.012	4.531 ± 0.147	0.136 ± 0.012
		NLPO [56]	4.931 ± 0.121	0.203 ± 0.022	4.987 ± 0.136	0.208 ± 0.033	-0.056 ± 0.007	4.482 ± 0.124	0.136 ± 0.019
		CPPO-H [8]	5.059 ± 0.212	0.211 ± 0.012	5.410 ± 0.189	0.213 ± 0.010	-0.351 ± 0.015	4.629 ± 0.127	0.165 ± 0.017
		COFS-DPO (Ours)	**5.756 ± 0.212**	**0.230 ± 0.010**	**6.398 ± 0.224**	**0.234 ± 0.007**	**-0.642 ± 0.014**	**5.641 ± 0.151**	**0.174 ± 0.021**
	256	DPO [4]	5.754 ± 0.183	0.230 ± 0.010	5.766 ± 0.215	0.230 ± 0.013	-0.012 ± 0.009	4.793 ± 0.131	0.168 ± 0.019
		PPO+EWC [6]	4.944 ± 0.161	0.208 ± 0.011	4.997 ± 0.172	0.211 ± 0.015	-0.053 ± 0.007	4.505 ± 0.249	0.137 ± 0.010
		PPO+LwF [32]	4.907 ± 0.198	0.203 ± 0.015	4.969 ± 0.167	0.210 ± 0.013	-0.062 ± 0.007	4.562 ± 0.154	0.151 ± 0.007
		PPO+TFCL [53]	5.068 ± 0.185	0.213 ± 0.012	5.142 ± 0.231	0.204 ± 0.013	-0.074 ± 0.011	4.563 ± 0.214	0.145 ± 0.014
		PC[54]	4.923 ± 0.245	0.207 ± 0.023	4.981 ± 0.178	0.237 ± 0.016	-0.058 ± 0.011	4.558 ± 0.214	0.119 ± 0.008
		HN-PPO [55]	5.072 ± 0.235	0.214 ± 0.011	5.153 ± 0.234	0.208 ± 0.019	-0.081 ± 0.009	4.575 ± 0.212	0.157 ± 0.012
		NLPO [56]	5.113 ± 0.261	0.209 ± 0.013	5.201 ± 0.246	0.206 ± 0.020	-0.088 ± 0.011	4.501 ± 0.196	0.146 ± 0.014
		CPPO-H [8]	5.270 ± 0.284	0.211 ± 0.016	5.618 ± 0.258	0.217 ± 0.021	-0.348 ± 0.014	4.664 ± 0.188	0.134 ± 0.016
		COFS-DPO (Ours)	**5.816 ± 0.248**	**0.234 ± 0.012**	**6.430 ± 0.245**	**0.241 ± 0.013**	**-0.614 ± 0.023**	**5.664 ± 0.213**	**0.178 ± 0.017**
LLaMA3	16	DPO [4]	5.809 ± 0.215	0.110 ± 0.013	5.816 ± 0.236	0.113 ± 0.015	-0.007 ± 0.014	5.473 ± 0.267	0.099 ± 0.016
		PPO+EWC [6]	5.076 ± 0.221	0.102 ± 0.016	5.160 ± 0.186	0.113 ± 0.027	-0.084 ± 0.012	5.050 ± 0.271	0.102 ± 0.011
		PPO+LwF [32]	5.007 ± 0.232	0.103 ± 0.012	5.109 ± 0.245	0.110 ± 0.012	-0.102 ± 0.016	4.601 ± 0.169	0.111 ± 0.014
		PPO+TFCL [53]	5.082 ± 0.231	0.108 ± 0.017	5.171 ± 0.235	0.113 ± 0.009	-0.089 ± 0.009	4.624 ± 0.123	0.112 ± 0.011
		PC[54]	5.012 ± 0.271	0.094 ± 0.016	5.104 ± 0.244	0.112 ± 0.013	-0.092 ± 0.010	4.573 ± 0.239	0.082 ± 0.014
		HN-PPO [55]	5.098 ± 0.258	0.098 ± 0.014	5.168 ± 0.198	0.106 ± 0.019	-0.070 ± 0.015	4.627 ± 0.211	0.127 ± 0.010
		NLPO [56]	5.018 ± 0.219	0.087 ± 0.013	5.101 ± 0.272	0.108 ± 0.012	-0.083 ± 0.008	4.594 ± 0.222	0.110 ± 0.016
		CPPO-H [8]	5.121 ± 0.214	0.096 ± 0.011	5.449 ± 0.261	0.101 ± 0.009	-0.328 ± 0.021	5.318 ± 0.264	0.089 ± 0.013
		COFS-DPO (Ours)	**5.867 ± 0.220**	**0.115 ± 0.014**	**7.285 ± 0.288**	**0.159 ± 0.012**	**-1.418 ± 0.026**	**7.094 ± 0.301**	**0.139 ± 0.010**
	256	DPO [4]	5.801 ± 0.271	0.114 ± 0.023	5.805 ± 0.243	0.114 ± 0.012	-0.004 ± 0.007	5.477 ± 0.264	0.099 ± 0.010
		PPO+EWC [6]	5.121 ± 0.215	0.101 ± 0.016	5.220 ± 0.275	0.105 ± 0.014	-0.099 ± 0.017	5.216 ± 0.248	0.115 ± 0.010
		PPO+LwF [32]	5.107 ± 0.214	0.098 ± 0.013	5.201 ± 0.237	0.110 ± 0.010	-0.094 ± 0.014	5.203 ± 0.235	0.108 ± 0.011
		PPO+TFCL [53]	5.172 ± 0.233	0.109 ± 0.012	5.263 ± 0.269	0.116 ± 0.035	-0.091 ± 0.015	5.278 ± 0.249	0.094 ± 0.031
		PC[54]	4.893 ± 0.198	0.101 ± 0.023	4.980 ± 0.251	0.107 ± 0.012	-0.087 ± 0.022	4.995 ± 0.276	0.056 ± 0.005
		HN-PPO [55]	5.168 ± 0.314	0.111 ± 0.018	5.235 ± 0.341	0.109 ± 0.014	-0.067 ± 0.022	5.280 ± 0.361	0.096 ± 0.021
		NLPO [56]	5.096 ± 0.277	0.092 ± 0.019	5.167 ± 0.301	0.108 ± 0.024	-0.071 ± 0.014	5.236 ± 0.267	0.038 ± 0.012
		CPPO-H [8]	5.322 ± 0.255	0.102 ± 0.011	5.657 ± 0.248	0.097 ± 0.014	-0.335 ± 0.022	5.351 ± 0.257	0.060 ± 0.007
		COFS-DPO (Ours)	**5.895 ± 0.269**	**0.116 ± 0.011**	**7.508 ± 0.285**	**0.163 ± 0.014**	**-1.613 ± 0.022**	**7.317 ± 0.275**	**0.143 ± 0.014**

Note: CPPO supports heuristic and learnable hyperparameter configurations.We use the heuristic variant (CPPO-H) for comparison with COFS-DPO. Results are reported for GPT2-s and LLaMA3 with LoRA ranks 16 and 256 over five trials. Additional implementation details are provided in the Supplementary Material. EWC: elastic weight consolidation; LwF: learning without forgetting; TFCL: task-free continual learning; PC: policy consolidation; HN-PPO: hypernetwork proximal policy optimization; NLPO: natural language policy optimization; CPPO-H: the heuristic variant of continual proximal policy optimization.

summarization task using human-preference datasets. We use GPT2-s and LLaMA3 as the base models. After training on Task 1, we evaluate the model on Task 1 using the rPMS and ROUGE metrics. We then continue training on Task 2 and evaluate the resulting model on both tasks. Changes in the Task-1 rPMS before and after Task-2 training are used to assess catastrophic forgetting. With GPT2-s, COFS-DPO achieves an SFR of approximately −0.6, outperforming all baselines in memory retention. With LLaMA3, it achieves an SFR of approximately −1.5, nearly twice that of the best-performing PPO variant. These results demonstrate the strong memory-retention capability of COFS-DPO.

We further compare COFS-DPO with CPPO-H, a representative continual preference-optimization baseline included in Table 3. With GPT2-s and LoRA rank 16, COFS-DPO improves the SFR from −0.351 to −0.642 and the final Task-2 rPMS from 4.629 to 5.641. With LLaMA3 and LoRA rank 16, the corresponding SFR improves from −0.328 to −1.418, while the final Task-2 rPMS increases from 5.318 to 7.094. Similar improvements are observed with LoRA rank 256. These results show that COFS-DPO better balances the retention of Task-1 preferences and alignment with Task 2.

4.4 Ablation studies

To validate the impact of the coefficient α in the regularization term of our objective function on the win rate of models in the controlled sentiment generation

task, we designed experiments with α ranging from 0 to 0.9. The results, illustrated in the left panel of Fig. 3, indicate that the model's win rate remains stable around 50%, even in the least favorable scenario ($\alpha = 0$), demonstrating the stability of our method. The second panel from the left in Fig. 3 investigates the effect of varying the learning-rate multiples between fast-slow modules on training effectiveness. Across various LoRA rank settings, OFS-DPO significantly outperforms the baseline PPO, maintaining a lead of at least 10% even in the worst-case scenario. This suggests that our method is robust across different learning-rate configurations. As illustrated in the second panel from the right in Fig. 3, increasing the batch size and the contrastive update period between fast-slow modules leads to a more stable win rate for our models. This demonstrates the positive impact of these adjustments on model performance. The rightmost panel of Fig. 3 shows the empirical distribution of the gradient norms of DPO and OFS-DPO during training. The results indicate that OFS-DPO maintains a more sustained gradient signal than the original DPO. This observation is consistent with the mechanism described in Proposition 3.3.1: when the original DPO gradient becomes small, the difference in preference learning between the F-module and S-module allows the $\mathcal{L}_{DPO-FS}$ term to provide an additional gradient component. Consequently, OFS-DPO maintains a more sustained gradient signal during the evaluated training process.

To examine Assumptions 3.3.1 and 3.3.2, we record the LoRA gradient norms and parameter distances from initialization during three complete OFS-DPO training runs with GPT-2 Large on IMDB. The runs use random seeds 42, 43, and 44, with 6250 optimizer updates in each run. As shown in Fig. 4, the median gradient norms range from 2.46 to 2.53, the 95th percentiles range from 3.01 to 3.10, and the largest observed gradient norm is 19.65. The maximum distances from initialization are 0.0813 for the F-module and 0.0407 for the S-module, corresponding to approximately 0.59% and 0.30% of their parameter norms, respectively. Thus, the gradient norms remain finite, and both modules remain close to their initialization throughout the evaluated training trajectories. These results provide empirical support for Assumptions 3.3.1 and 3.3.2 under the evaluated setting.

To investigate the impact of retaining specific domain samples on the memory retention capability of models in cross-domain tasks, we designed validation experiments

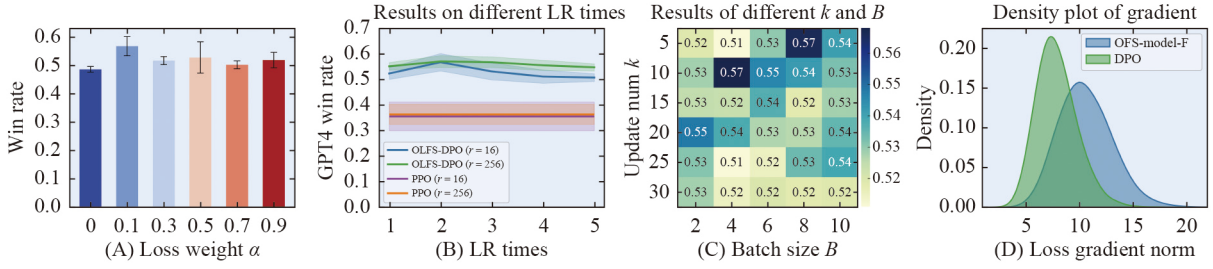


Fig. 3 Ablation results on IMDB. From left to right, the panels show: (A) win rates for different regularization coefficients α; (B) win rates of OFS-DPO and PPO under different fast-slow learning-rate multipliers; (C) the effects of batch size and role-comparison period k on win rate; and (D) kernel density estimates of the loss-gradient norms of DPO and OFS-DPO during training.

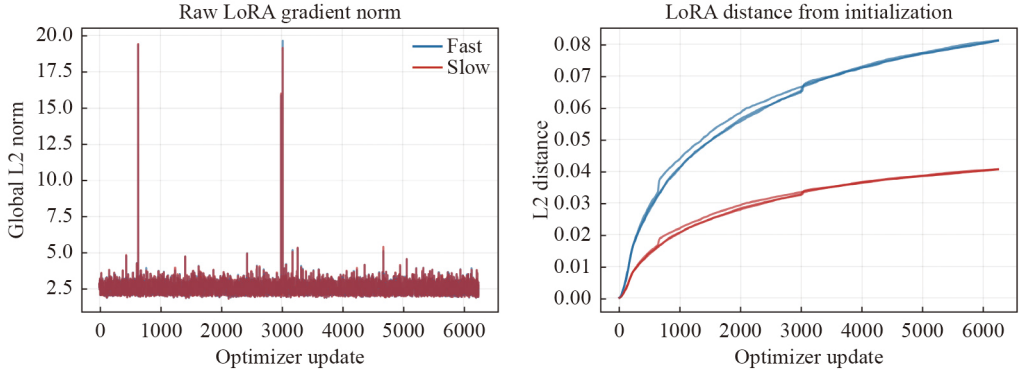


Fig. 4 Empirical boundedness measurements on IMDB. The left panel shows the raw LoRA gradient norms of the F-module and S-module over one complete training epoch. The right panel shows their distances from initialization. Each trajectory corresponds to one of three random seeds (42, 43 and 44).

Table 4 COFS-DPO results with different numbers of retained samples.

Samples	$rPMS_1$	SFR	$rPMS_2$
100	5.988	−0.242	5.156
200	6.180	−0.434	5.340
500	6.398	−0.652	5.641
1000	6.367	−0.621	5.590
2000	6.312	−0.641	5.605

with total sample sizes of 100, 200, 500, 1000 and 2000 for two tasks, as shown in Table 4.

The results indicate that when the total sample size increases to 500, COFS-DPO achieves optimal retention of historical preference information. Beyond this sample size, further increases do not enhance the method's performance. This finding suggests that within our proposed framework, it is unnecessary to retain a large number of specific domain samples to achieve excellent results.

5 Conclusions

In this work, inspired by intraspecific competition theory, we propose a simple and effective OFS-DPO, which leverages the competition between fast and slow modules under the same objective to achieve continual preference learning, supported by theoretical analyses of regret and gradient behavior. Furthermore, to extend OFS-DPO to cross-domain settings, we introduce COFS-DPO, which achieves preference alignment through a linear combination of task-specific LoRA modules from different domains, as validated both experimentally and theoretically. Our work demonstrates that the proposed methods based on intraspecific competition provide new insights and solutions for online human preference alignment tasks and have the potential for broad applicability across multiple domains.

The current experiments cover a limited range of model scales and preference-alignment tasks, while the cross-domain evaluation considers two sequential task domains. Future work will evaluate the proposed framework on larger language models, a broader range of preference- alignment tasks, and cross-domain settings involving three or more task domains and different domain-arrival orders.

Supporting Information Supporting Information is available in the online version of this article at https://doi.org/10.2738/ENGAI.2026.0001.

Data availability The IMDB data supporting this study are available in the Stanford Natural Language Processing (NLP) data repository at accession Large Movie Review Dataset v1.0.The TL; DR Summarize data supporting this study are available in GitHub at repository identifier openai/summarize-from-feedback. The Anthropic Helpful and Harmless data supporting this study are available in the Hugging Face Hub at repository identifier Anthropic/hh-rlhf. The CarperAI human-preference data supporting this study are available in the Hugging Face Hub at repository identifier CarperAI/openai_summarize_comparisons.

Acknowledgments This work was supported in part by the National Natural Science Foundation of China (No. 6250076080), and in part by the China Postdoctoral Science Foundation (No. 2025M771537). The authors used OpenAI Codex (powered by GPT-5) as a language-editing tool to improve grammar, clarity, and readability. The AI tool was not used for data analysis, data generation, image processing, orscientific content creation. All outputs were carefully reviewed and verifiedby the authors, who take full responsibility for the final content. The artificial intelligence model described in the Experimental Settings section was used solely as a research tool (GPT-4, online version). Specifically, GPT-4 was used as a surrogate human evaluator to compare model-generated outputs with the corresponding reference responses in the in-domain experiments, and theresulting preferences were used to calculate win rates. The AI system was not used for data fabrication, data manipulation, image alteration, or undisclosed scientific content generation. All analyses, interpretations, and conclusions were independently conducted, critically evaluated, and validated by the authors, who assume full responsibility for the integrity and accuracy of the manuscript.

Competing interests The authors declare no conflicts of interest.

References

1. Stiennon N, Ouyang L, Wu J, et al. Learning to summarize from human feedback. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. Vancouver, Canada, 2020, 253
2. Christiano P F, Leike J, Brown T B, et al. Deep reinforcement learning from human preferences. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach, CA, USA, 2017, 4302−4310
3. Bai Y T, Jones A, Ndousse K, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. arXiv preprint: arXiv: 2204.05862, 2022
4. Rafailov R, Sharma A, Mitchell E, et al. Direct preference optimization: your language model is secretly a reward model. In: Proceedings of the 37th International Conference on Neural Information Processing Systems. New Orleans, LA, USA, 2023, 2338
5. Guo S M, Zhang B, Liu T L, et al. Direct language model alignment from online AI feedback. arXiv preprint: arXiv: 2402.04792, 2024
6. Kirkpatrick J, Pascanu R, Rabinowitz N, et al. Overcoming catastrophic forgetting in neural networks. Proceedings of the National Academy of Sciences of the United States of America, 2017, 114(13): 3521−3526
7. Rolnick D, Ahuja A, Schwarz J, et al. Experience replay for continual learning. In: Proceedings of the 33rd International Conference on Neural Information Processing Systems. Vancouver, Canada, 2019, 32
8. Zhang H, Lei Y, Gui L, et al. CPPO: continual learning for

reinforcement learning with human feedback. In: Proceedings of the 12th International Conference on Learning Representations. Vienna, Austria, 2024
9. Lee H, Phatale S, Mansoor H, et al. RLAIF vs. RLHF: scaling reinforcement learning from human feedback with AI feedback. In: Proceedings of the 41st International Conference on Machine Learning. Vienna, Austria, 2024, 26874–26901
10. Bolnick D I. Intraspecific competition favours niche width expansion in *Drosophila melanogaster*. Nature, 2001, 410(6827): 463−466
11. Thorne B L, Breisch N L, Muscedere M L. Evolution of eusociality and the soldier caste in termites: influence of intraspecific competition and accelerated inheritance. Proceedings of the National Academy of Sciences of the United States of America, 2003, 100(22): 12808−12813
12. Agarwal N, Bullins B, Hazan E, et al. Online control with adversarial disturbances. In: Proceedings of the 36th International Conference on Machine Learning. Long Beach, CA, USA, 2019, 111–119
13. Hazan E, Kakade S, Singh K. The nonstochastic control problem. In: Proceedings of the 31st International Conference on Algorithmic Learning Theory. San Diego, CA, USA, 2020, 408–421
14. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks. Communications of the ACM, 2020, 63(11): 139−144
15. Hu E J, Shen Y L, Wallis P, et al. LoRA: low-rank adaptation of large language models. In: Proceedings of the 10th International Conference on Learning Representations. Online, 2022.
16. Lake B M, Baroni M. Human-like systematic generalization through a meta-learning neural network. Nature, 2023, 623(7985): 115−121
17. van de Ven G M, Siegelmann H T, Tolias A S. Brain-inspired replay for continual learning with artificial neural networks. Nature Communications, 2020, 11(1): 4069
18. Qi B Q, Chen X Q, Gao J Q, et al. Interactive continual learning: fast and slow thinking. In: Proceedings of 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA, USA, 2024, 12882–12892
19. Jiang Z M, Han X T, Jin H Y, et al. Chasing fairness under distribution shift: a model weight perturbation approach. In: Proceedings of the 37th International Conference on Neural Information Processing Systems. New Orleans, LA, USA, 2023, 2793
20. Wallace B, Dang M H, Rafailov R, et al. Diffusion model alignment using direct preference optimization. In: Proceedings of 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA, USA, 2024, 8228–8238
21. Pappa M, Collorone L, Ficarra G, et al. MoDiPO: text-to-motion alignment via AI-feedback-driven direct preference optimization. Frontiers in Computer Science, 2026, 8: 1707808,
22. Majumder N, Hung C Y, Ghosal D, et al. Tango 2: aligning diffusion-based text-to-audio generations through direct preference optimization. In: Proceedings of the 32nd ACM International Conference on Multimedia. Melbourne, Australia, 2024, 564–572
23. Zhang R H, Gui L K, Sun Z Q, et al. Direct preference optimization of video large multimodal models from language model reward. In: Proceedings of 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies. Albuquerque, New Mexico, 2025, 694–717
24. Yang G Y, Chen J H, Lin W Z, et al. Direct preference optimization for neural machine translation with Minimum Bayes Risk decoding. In: Proceedings of 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Mexico City, Mexico, 2024, 391–398
25. Zhou Z H, Liu J, Yang C, et al. Beyond one-preference-fits-all alignment: multi-objective direct preference optimization. In: Proceedings of Findings of the Association for Computational Linguistics: ACL 2024. Bangkok, Thailand, 2024, 10586–10613
26. Pal A, Karkhanis D, Dooley S, et al. Smaug: fixing failure modes of preference optimisation with DPO-positive. arXiv preprint: arXiv: 2402.13228, 2024
27. Xie J W, Yan S P, He X M. General incremental learning with domain-aware categorical representations. In: Proceedings of 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans, LA, USA, 2022, 14331–14340
28. Simon C, Faraki M, Tsai Y H, et al. On generalizing beyond domains in cross-domain continual learning. In: Proceedings of 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans, LA, USA, 2022, 9255–9264
29. Wang L Y, Zhang X X, Su H, et al. A comprehensive survey of continual learning: theory, method and application. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024, 46(8): 5362−5383
30. Aljundi R, Babiloni F, Elhoseiny M, et al. Memory aware synapses: learning what (not) to forget. In: Proceedings of the 15th European Conference on Computer Vision. Munich, Germany, 2018, 144–161
31. Chaudhry A, Dokania P K, Ajanthan T, et al. Riemannian walk for incremental learning: understanding forgetting and intransigence. In: Proceedings of the 15th European Conference on Computer Vision. Munich, Germany, 2018, 556–572
32. Li Z Z, Hoiem D. Learning without forgetting. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(12): 2935−2947
33. Castro F M, Marín-Jiménez M J, Guil N, et al. End-to-end incremental learning. In: Proceedings of the 15th European Conference on Computer Vision. Munich, Germany, 2018, 241–257

34. Sun F K, Ho C H, Lee H Y. LAMOL: language modeling for lifelong language learning. In: Proceedings of the 8th International Conference on Learning Representations. Addis Ababa, Ethiopia, 2020
35. Liu X L, Wu C S, Menta M, et al. Generative feature replay for class-incremental learning. In: Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. Seattle, WA, USA, 2020, 915–924 1109/C
36. Qi B Q, Gao J Q, Chen X Q, et al. Contrastive augmented Graph2Graph memory interaction for few shot continual learning. IEEE Transactions on Circuits and Systems for Video Technology, 2025, 35(6): 5805−5820
37. Rosenfeld E, Ravikumar P, Risteski A. An online learning approach to interpolation and extrapolation in domain generalization. In: Proceedings of the 25th International Conference on Artificial Intelligence and Statistics. Valencia, Spain, 2022, 2641–2657
38. Volpi R, Larlus D, Rogez G. Continual adaptation of visual representations via domain randomization and meta-learning. In: Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville, TN, USA, 2021, 4441–4451
39. Kundu J N, Venkatesh R M, Venkat N, et al. Class-incremental domain adaptation. In: Proceedings of the 16th European Conference on Computer Vision. Glasgow, UK, 2020, 53–69
40. Zhao P, Zhang L. Improved analysis for dynamic regret of strongly convex and smooth functions. In: Proceedings of the 3rd Conference on Learning for Dynamics and Control. Zurich, Switzerland, 2021, 48–59
41. Haghtalab N, Roughgarden T, Shetty A. Smoothed analysis of online and differentially private learning. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. Vancouver, Canada, 2020, 772
42. Nguyen L M, Nguyen P H, van Dijk M, et al. SGD and Hogwild! Convergence without the bounded gradients assumption. In: Proceedings of the 35th International Conference on Machine Learning. Stockholm, Sweden, 2018, 3750–3758
43. Gorbunov E, Hanzely F, Richtárik P. A unified theory of SGD: variance reduction, sampling, quantization and coordinate descent. In: Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics. Palermo, Italy, 2020, 680–690
44. Jang U, Lee J D, Ryu E K. LoRA training in the NTK regime has no spurious local minima. In: Proceedings of the 41st International Conference on Machine Learning. Vienna, Austria, 2024, 21306–21328
45. Maas A L, Daly R E, Pham P T, et al. Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland, OR, USA, 2011, 142–150
46. Radford A, Wu J, Child R, et al. Language models are unsupervised multitask learners. Available online (accessed 17 Sep 2026)
47. Ziegler D M, Stiennon N, Wu J, et al. Fine-tuning language models from human preferences. arXiv preprint: arXiv: 1909.08593, 2019
48. Wang B, Komatsuzaki A. GPT-J-6B: a 6 billion parameter autoregressive language model. Available online (accessed 17 Sep 2026)
49. Chen Y, Wang R, Jiang H Y, et al. Exploring the use of large language models for reference-free text quality evaluation: an empirical study. In: Proceedings of Findings of the Association for Computational Linguistics: IJCNLP-AACL 2023. Nusa Dua, Bali, 2023, 361–374
50. AI@Meta. Llama 3 model card. Available online (accessed 17 Sep 2026)
51. Gao L, Schulman J, Hilton J. Scaling laws for reward model overoptimization. In: Proceedings of the 40th International Conference on Machine Learning. Honolulu, HI, USA, 2023, 10835–10866
52. Lin C Y. ROUGE: a package for automatic evaluation of summaries. In: Proceedings of Text Summarization Branches Out. Barcelona, Spain, 2004, 74–81
53. Aljundi R, Kelchtermans K, Tuytelaars T. Task-free continual learning. In: Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA, 2019, 11246–11255
54. Kaplanis C, Shanahan M, Clopath C. Policy consolidation for continual reinforcement learning. In: Proceedings of the 36th International Conference on Machine Learning. Long Beach, California, USA, 2019, 3242–3251
55. Schöpf P, Auddy S, Hollenstein J, et al. Hypernetwork-PPO for continual reinforcement learning. In: Proceedings of the Deep Reinforcement Learning Workshop at NeurIPS 2022. New Orleans, USA, 2022
56. Ramamurthy R, Ammanabrolu P, Brantley K, et al. Is reinforcement learning (not) for natural language processing: benchmarks, baselines, and building blocks for natural language policy optimization. In: Proceedings of the 11th International Conference on Learning Representations. Kigali, Rwanda, 2023



Biqing Qi is currently a Researcher at Shanghai AI Lab and a Postdoctoral Fellow at the University of Hong Kong, China. He received his Ph.D. from Harbin Institute of Technology, jointly supervised by Tsinghua University. His research interests include machine learning and natural language processing. He has published over 70 papers in top-tier conferences and journals, including the Conference on Neural Information Processing Systems (NeurIPS), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the Annual Meeting of the Association for Computational Linguistics (ACL), and IEEE Transactions on Pattern Analysis and Machine Intelligence

(TPAMI), and has led four projects, including national major projects and Shanghai major projects.



Junqi Gao is currently a third-year Ph.D. student at the School of Mathematics, Harbin Institute of Technology, China. His primary research focus is efficient continual learning, while also exploring topics such as the fusion and merging of heterogeneous or homogeneous deep learning models. His relevant work has been published in conferences including the AAAI Conference on Artificial Intelligence (AAAI), ACL, CVPR and NeurIPS. His research interest lies in developing efficient continual deep learning techniques and bridging the gap between theoretical research and practical real-world applications.



Pengfei Li received his B.S. degree in Mathematics and Applied Mathematics from Shanghai Jiao Tong University. He is currently pursuing his Ph.D. through a joint doctoral program between Harbin Institute of Technology and Shanghai Innovation Institute. His research interests include multi-agent reinforcement learning, reasoning and alignment of large language models, code generation, and model robustness. His work has appeared in leading international conferences on artificial intelligence, including NeurIPS, ACL and AAAI.



Fangyuan Li is currently a Ph.D. student at the School of Astronautics, Harbin Institute of Technology, jointly trained with Shanghai Innovation Institute. He is advised by Prof. Jianxing Liu and Researcher Yuqiang Li. His research interests include large language models, autonomous agents, self-improving reasoning, and artificial intelligence for scientific discovery. His research focuses on improving the reasoning, learning, and adaptation capabilities of intelligent systems, with related work published in ACL, CVPR and AAAI.



Zhixuan Su received the B.S. degree in Computer Science and Technology from Harbin Institute of Technology, Shenzhen, China, in 2026. She is currently pursuing doctoral study through the HIT–Pujiang joint training program. Her research interests include agentic reinforcement learning (Agentic RL), large language models, model architecture and learning methods, and online model fusion.



Kaiyan Zhang received the B.S. degree in Computer Science and Technology from the Harbin Institute of Technology, in 2022. He is currently working toward the third-year Ph.D. degree with the Department of Electronic Engineering, Tsinghua University. His research interests include the alignment and collaboration of large language models (LLMs), focusing on developing trustworthy and scalable collaborative intelligence systems.



Bowen Zhou is the Director and Chief Scientist with Shanghai AI Lab, Huiyan Chair Professor with the Department of Electrical Engineering, Tsinghua University, CAAI Fellow, the Chief Scientist of the scientific and technological innovation 2030 "New Generation Artificial Intelligence" major project. Prior to that, he was the director of AI Foundations of IBM T. J. Watson Research Center, Chief Scientist of IBM Watson Group, a distinguished engineer of IBM, SVP of JD.Com, the Chairman of JD. Com's Technology Committee, and the President of JD Cloud & AI BU. He won various influential awards including "Wu Wenjun Artificial Intelligence Science and Technology Award—Outstanding Contribution Award".