Se-DPO: Self-Evolving Token Credit for Direct Preference Optimization

Wenxiao Zhao[1,2] **Shu Wang**[1] **Ying Nian Wu**[1]
[1]University of California, Los Angeles
[2]Shanghai AI Laboratory

Abstract

Direct Preference Optimization (DPO) aggregates token-level log-probability ratios via uniform summation, implicitly treating all tokens as contributing equally to the preference signal. However, the contribution of individual tokens to the preference signal varies. We introduce token credit, which modulates each token's KL regularization based on its contribution to the preference outcome. We derive that effective token credit is proportional to the magnitude of each token's implicit reward, and observe that this quantity evolves substantially during training. This implies that static token credit becomes increasingly misaligned as training progresses. In this work, we propose Se-DPO (Self-Evolving Token Credit for DPO), a live mechanism that derives token credit from the model's own evolving internal signals during DPO training. Since the reward signal varies in reliability across positions, Se-DPO calibrates token credit based on both the strength and the confidence of each token's contribution. Se-DPO requires no external models, adding only a lightweight calibration network with minimal computational overhead. Experiments show that Se-DPO improves over DPO by up to 9.8 points on AlpacaEval 2 and 12.2 points on Arena-Hard.

1 Introduction

Direct Preference Optimization (Rafailov et al., 2023) has emerged as a widely adopted method for aligning large language models with human preferences. By reparameterizing the reward function in the reinforcement learning from human feedback (RLHF) objective (Christiano et al., 2017; Ziegler et al., 2019; Ouyang et al., 2022; Stiennon et al., 2020; Bai et al., 2022; Touvron et al., 2023; Schulman et al., 2017), DPO enables direct policy optimization from preference data without training a separate reward model. In the standard formulation, DPO aggregates token-level log-probability ratios via uniform summation, implicitly treating all tokens as equally important to the preference signal.

However, not all tokens contribute equally to the quality of a response. A factual error in a single token can flip a preference label (Rafailov et al., 2024), while filler words rarely affect human judgment. Recent methods have sought to address this mismatch by introducing token-level importance signals into the preference optimization objective, drawing on external sources such as pre-trained teacher models (Zhu et al., 2025; Chen et al., 2025b; Zhong et al., 2025), contrastive language model pairs (Liu et al., 2024), learned sparse masks (Christopoulou et al., 2024), optimal transport matching (Li et al., 2025), per-token KL constraints (Zeng et al., 2024), prompted self-evaluation (Zhou et al., 2025), or oracle-based token selection (Yang et al., 2025a). While these approaches differ in their signal sources, they share a common pattern: token importance is computed prior to or outside the training loop and treated as approximately fixed throughout optimization. A natural alternative is to derive token credit from DPO's own implicit reward (Rafailov et al., 2024), but recent work observed that this signal can assign disproportionate importance to non-critical tokens such as punctuation and line breaks (Chen et al., 2025b), leaving its direct use largely unexplored.

In this work, we study the implicit reward not at a single snapshot but across the full trajectory of DPO training. Our analysis reveals that the token-level implicit reward undergoes substantial evolution during training: the set of tokens deemed important shifts considerably between early and late stages, and the importance ranking stabilizes only gradually over the course of optimization. Crucially, this gradual stabilization suggests that the noisy importance estimates observed by prior work (Chen et al., 2025b) reflect the early training state rather than an inherent limitation of the implicit reward. Since token importance continues to shift until convergence, any static credit assignment becomes increasingly stale as optimization progresses. This temporal perspective distinguishes our approach from methods that adapt KL regularization at the sample level (Lee et al., 2025; Zhou et al., 2024; Ivison et al., 2024) or bootstrap implicit rewards at the sequence level (Chen et al., 2025a)—neither of which captures the within-training evolution of token-level importance.

Based on these observations, we show that relaxing DPO's uniform KL coefficient to be token-specific yields an objective where each token's contribution is scaled by a credit term. We derive that, under a variance-minimization criterion on the preference logit, the credit at each position should be proportional to the magnitude of the token's implicit reward. Intuitively, tokens with larger implicit rewards carry more preference-relevant information and should be allowed to deviate further from the reference policy, while tokens with near-zero implicit rewards contribute little to the preference signal and benefit from tighter regularization. Since the implicit reward evolves during training, the credit should also evolve—any static approach uses an increasingly stale signal by design.

We propose **Se-DPO** (Self-Evolving Token Credit for DPO), a live mechanism that periodically extracts implicit reward magnitudes as token credit during DPO training. Se-DPO adds no external models and only a lightweight calibration network with minimal computational overhead.

Our contributions are as follows:

- We observe that DPO's token-level implicit reward shifts considerably over the course of training, with limited overlap between top-$|\hat{r}_t|$ tokens at early and late stages.
- We introduce token credit to modulate per-token KL regularization strength, and analyze how it can be informed by implicit reward magnitude and estimation confidence.
- We propose Se-DPO, which derives token credit online from the model's own evolving internal signals via a lightweight calibration network, requiring no external models and achieving up to 50.6% win rate on AlpacaEval 2 and 43.3% on Arena-Hard.

2 Preliminaries

2.1 DPO and Implicit Reward

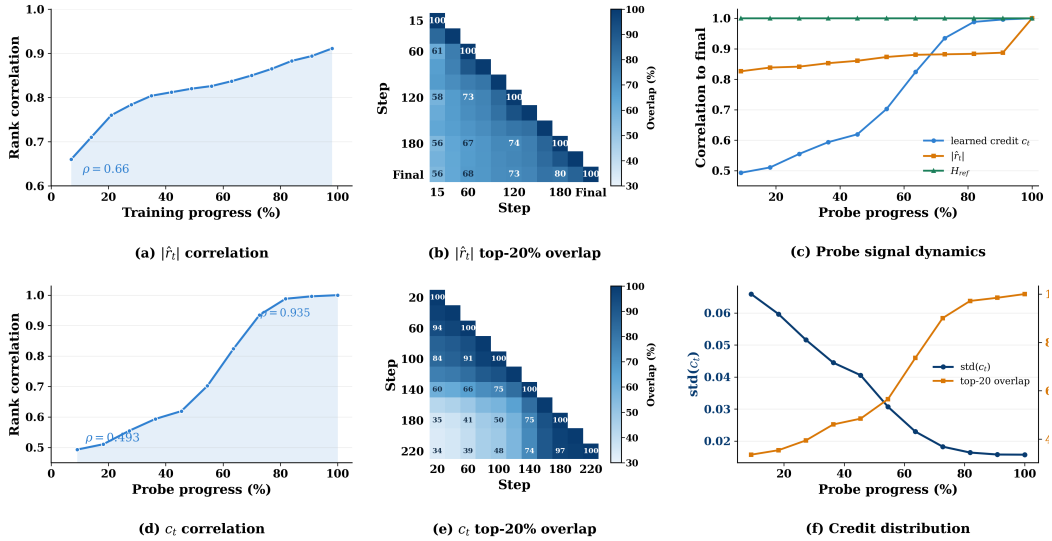
The standard RLHF objective maximizes expected reward subject to a KL divergence constraint against a reference policy π_{ref}:

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

where $\beta > 0$ controls the strength of regularization. DPO (Rafailov et al., 2023) reparameterizes the reward in terms of the optimal policy, yielding a supervised loss over preference pairs (x, y^c, y^r):

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(x,y^c,y^r)}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y^c|x)}{\pi_{\text{ref}}(y^c|x)} - \beta\log\frac{\pi_\theta(y^r|x)}{\pi_{\text{ref}}(y^r|x)}\right)\right]. \tag{2}$$

Since language models generate tokens autoregressively, the sequence-level log-ratio decomposes as a sum of token-level terms. Rafailov et al. (2024) showed that DPO implicitly



(a) $|\hat{r}_t|$ correlation (b) $|\hat{r}_t|$ top-20% overlap (c) Probe signal dynamics

(d) c_t correlation (e) c_t top-20% overlap (f) Credit distribution

Figure 1: Token importance and learned credit evolve during training. **(a,b)** Implicit-reward magnitude $|\hat{r}_t|$ in the original training run: rank correlation to the final checkpoint increases over time, while top-20% token overlap remains limited. **(c–f)** Learned credit c_t in a separate Se-DPO LoRA credit-dynamics probe: credit correlation increases from 0.493 to 1.0, early-to-final top-20% overlap is 33.6%, and the credit distribution gradually stabilizes.

learns a token-level reward:

$$\hat{r}_t = \beta(\log \pi_\theta(y_t|x, y_{<t}) - \log \pi_{\text{ref}}(y_t|x, y_{<t})) \,. \tag{3}$$

2.2 Token-Specific KL Regularization

Standard DPO applies a uniform KL coefficient β to all token positions. Prior work has explored token-level KL decompositions (Zeng et al., 2024) and learned masks over token-level KL contributions (Christopoulou et al., 2024). In this work, we consider a generalization where the KL strength varies across positions:

$$\max_{\pi} \mathbb{E}_{y\sim\pi}\left[\sum_{t=1}^{T} r_t - \beta_t \log \frac{\pi(y_t|x, y_{<t})}{\pi_{\text{ref}}(y_t|x, y_{<t})}\right], \tag{4}$$

where β_t controls the KL penalty at position t. At each position, the optimization problem is:

$$\max_{\pi(\cdot|x,y_{<t})} \sum_{y_t} \pi(y_t|x, y_{<t})\left[r_t(y_t) - \beta_t \log \frac{\pi(y_t|x, y_{<t})}{\pi_{\text{ref}}(y_t|x, y_{<t})}\right]. \tag{5}$$

This is a KL-regularized optimization whose closed-form solution is:

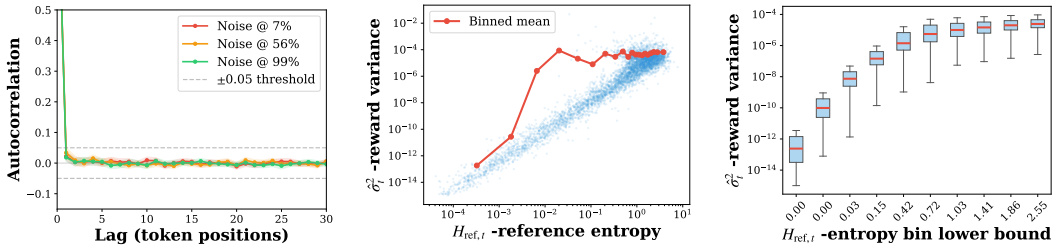
$$\pi^*(y_t|x, y_{<t}) = \frac{1}{Z_t}\pi_{\text{ref}}(y_t|x, y_{<t}) \exp\left(\frac{r_t(y_t)}{\beta_t}\right), \tag{6}$$

where $Z_t = \sum_{y'_t} \pi_{\text{ref}}(y'_t|x, y_{<t}) \exp(r_t(y'_t)/\beta_t)$ is a position-specific partition function. Rearranging yields the implicit reward:

$$r_t(y_t) = \beta_t(\log \pi^*(y_t|x, y_{<t}) - \log \pi_{\text{ref}}(y_t|x, y_{<t})) + \beta_t \log Z_t. \tag{7}$$

Since Z_t depends only on the context $(x, y_{<t})$ and not on the sampled token y_t, it does not affect the relative contribution of different tokens within a response.[1] Define $c_t = \beta/\beta_t$

[1]With position-dependent β_t, the per-position partition functions generally do not cancel between chosen and rejected responses, even when β_t is fixed: the two continuations induce different contexts



(a) Noise autocorrelation decay. (b) $H_{\text{ref},t}$ vs. $\hat{\sigma}_t^2$. (c) Variance by entropy decile.

Figure 2: Empirical validation of Proposition 1 assumptions. **(a)** Noise autocorrelation of $\hat{r}_t$ drops below 0.05 within lag 1 at all training stages ($m/T < 0.004$), confirming near-independence across positions. **(b)** Log-log scatter of reference entropy vs. empirical reward variance (Spearman $\rho = 0.86$); binned means (red) show a clear monotonic trend. **(c)** Boxplot of $\hat{\sigma}_t^2$ by entropy decile confirms that higher-entropy positions produce less reliable implicit rewards.

and recall the token-level implicit reward $\hat{r}_t^{(s)} = \beta(\log \pi_\theta(y_t^{(s)}|x, y_{<t}^{(s)}) - \log \pi_{\text{ref}}(y_t^{(s)}|x, y_{<t}^{(s)}))$ from Eq. 3. The resulting preference logit is:

$$\hat{\Delta} = \sum_{t=1}^{T_c} c_t^{(c)} \hat{r}_t^{(c)} - \sum_{t=1}^{T_r} c_t^{(r)} \hat{r}_t^{(r)}, \tag{8}$$

where standard DPO is recovered when $\beta_t = \beta$ (i.e., $c_t = 1$) for all t. A smaller β_t at position t yields a larger credit c_t, reducing the KL penalty and allowing the policy to deviate further from the reference at that position. The central question is how to set β_t—equivalently, how to allocate per-token KL budgets.

3 Method

3.1 Token Credit Dynamics During Training

Existing token-level methods compute importance prior to or outside the training loop, treating it as approximately fixed. However, it remains unclear how token importance behaves as training progresses. To investigate, we track the implicit reward (Eq. 3) across the trajectory of standard DPO training (details in Appendix C).

Token importance evolves continuously during training. Figure 1 provides empirical motivation for our self-evolving credit design. We first revisit the implicit reward magnitude $|\hat{r}_t|$, which reflects token-level preference signals. As shown in Figure 1(a,b), top-$|\hat{r}_t|$ tokens change continuously during the original training run: their rank correlation with the final checkpoint increases over time, while the early-to-final top-20% overlap remains limited.

To verify that this phenomenon also appears in the actual credit used by Se-DPO, we run a separate Se-DPO LoRA credit-dynamics probe on fixed evaluation samples. As shown in Figure 1(c–f), the learned credit c_t used for per-token KL-budget allocation is also dynamic. Its correlation with the final checkpoint starts from 0.493 and gradually reaches 1.0, while the early-to-final top-20% overlap is only 33.6%. In contrast, the reference entropy H_{ref} remains static. These observations suggest that token credit is shaped by the evolving model state, motivating our online credit update rather than a static per-token KL-budget allocation. We formalize this intuition in Section 3.2 and propose the complete Se-DPO algorithm in Section 3.4.

after their first divergence, and therefore different Z_t values. Eq. 8 should thus be understood as a principled approximation to the token-specific KL objective. The uncancelled term acts as a context-dependent value offset; because Se-DPO mean-normalizes credits within each response and keeps them bounded, this residual cannot grow by uniformly scaling token-credit values. Similar approximations are also used by other token-level objectives that modify the per-token KL structure.

Algorithm 1 Se-DPO: Self-Evolving Token Credit Direct Preference Optimization

Require: Data $\mathcal{D}$, policy π_θ, reference π_{ref}, calibration network f_ϕ, warmup steps W

for each training step with batch $(x, y^c, y^r) \sim \mathcal{D}$ **do**
  // Token-level implicit rewards
  for $s \in \{c, r\}$, $t = 1, \ldots, T_s$ **do**
    $\hat{r}_t^{(s)} \leftarrow \beta\Big(\log \pi_\theta(y_t^{(s)} \mid x, y_{<t}^{(s)}) - \log \pi_{\text{ref}}(y_t^{(s)} \mid x, y_{<t}^{(s)})\Big)$
  end for
  // Token credit assignment
  if step $\leq W$ **then**
    $c_t^{(s)} \leftarrow 1$ for all t, s
  else
    $H_{\text{ref},t}^{(s)} \leftarrow -\sum_v \pi_{\text{ref}}(v \mid x, y_{<t}^{(s)}) \log \pi_{\text{ref}}(v \mid x, y_{<t}^{(s)})$
    $\tilde{c}_t^{(s)} \leftarrow f_\phi\Big(|\hat{r}_t^{(s)}|,\ H_{\text{ref},t}^{(s)}\Big)$
    $c_t^{(s)} \leftarrow \tilde{c}_t^{(s)} \,/\, \text{mean}_{t'}(\tilde{c}_{t'}^{(s)})$
  end if
  // Token-specific KL-regularized preference loss
  $\mathcal{L} \leftarrow -\log \sigma\Big(\sum_t c_t^{(c)}\, \hat{r}_t^{(c)} - \sum_t c_t^{(r)}\, \hat{r}_t^{(r)}\Big)$
  Update θ and ϕ via $\nabla_{\theta,\phi}\mathcal{L}$
end for

3.2 Two Complementary Signals for Token Credit

The analysis above establishes that token credit should be extracted online. We now ask: what signals should inform the credit assignment?

The implicit reward $\hat{r}_t$ (Eq. 3) provides a natural candidate. However, a high $|\hat{r}_t|$ can arise from multiple causes: the model may have learned a meaningful preference at that position, or the reference model may simply be uncertain there, making the log-ratio noisy. Using $|\hat{r}_t|$ alone conflates signal and noise.

We observe that the reference model's token-level entropy provides a complementary signal:

$$H_{\text{ref},t} = -\sum_{v \in \mathcal{V}} \pi_{\text{ref}}(v \mid x, y_{<t}) \log \pi_{\text{ref}}(v \mid x, y_{<t}), \tag{9}$$

where $\mathcal{V}$ is the vocabulary. At positions where $H_{\text{ref},t}$ is high, the reference model is uncertain, and the implicit reward is inherently noisier; at positions where $H_{\text{ref},t}$ is low, the reference model is confident, and deviations from it carry stronger signal.

This decomposition connects to the following result on optimal credit:

Proposition 1. *Consider the preference logit $\hat{\Delta} = \sum_t c_t \hat{r}_t$, where each $\hat{r}_t = r_t^* + \epsilon_t$ is a noisy estimate of the true token reward with $\text{Var}[\epsilon_t] = \sigma_t^2$. Since $c_t = \beta/\beta_t > 0$ by construction, we require $c_t \geq 0$. Assume the noise is m-dependent: $\text{Cov}[\epsilon_t, \epsilon_{t'}] = 0$ for $|t - t'| > m$.*

(a) *Under independent noise ($m = 0$), the non-negative credits that minimize $\text{Var}[\hat{\Delta}]$ subject to $\sum_t c_t |r_t^*| = C$ satisfy:*

$$c_t^* \;\propto\; \frac{|r_t^*|}{\sigma_t^2}. \tag{10}$$

(b) *For m-dependent noise with bounded credits $c_t \in [c_{\min}, c_{\max}]$, the cross-covariance contribution to $\text{Var}[\hat{\Delta}]$ is at most $O(m)$ times the diagonal term $\sum_t c_t^2 \sigma_t^2$. The solution from **(a)** thus remains a valid first-order approximation when $m \ll T$.*

Proof sketch. For Part (a), we minimize $\text{Var}[\hat{\Delta}] = \sum_t c_t^2 \sigma_t^2$ subject to the signal-preservation constraint $\sum_t c_t |r_t^*| = C$ with $c_t \geq 0$. The Lagrangian first-order condition $\partial\mathcal{L}/\partial c_t =$

$2c_t\sigma_t^2 - \lambda|r_t^*| = 0$ directly yields $c_t^* = \lambda|r_t^*|/(2\sigma_t^2) \propto |r_t^*|/\sigma_t^2$, which is automatically non-negative. For Part (b), with bounded credits and m-dependent noise, the cross-covariance contribution to $\mathrm{Var}[\hat{\Delta}]$ scales as $O(m \cdot c_{\max}^2 \rho_{\max})$ relative to the diagonal term, remaining a bounded perturbation when $m \ll T$. The full proof is in Appendix A. □

Part (b) ensures that the conclusion is robust to the local correlations inherent in autoregressive generation, provided the credit values remain bounded—a property guaranteed by the mean normalization. We verify empirically that the noise autocorrelation drops below 0.05 within lag 1, yielding $m/T < 0.004$ (Figure 2a), confirming that the independent-noise solution from Part (a) is itself a close approximation.

We use $|\hat{r}_t|$ as a proxy for $|r_t^*|$. For σ_t^2, we observe that reference entropy $H_{\mathrm{ref},t}$ provides an informative proxy: at high-entropy positions the reference model distributes probability mass across many tokens, so the log-ratio $\log \pi_\theta/\pi_{\mathrm{ref}}$ is inherently more sensitive to small changes in π_θ, increasing the estimation noise of $\hat{r}_t$. Empirically, we estimate σ_t^2 as the variance of $\hat{r}_t$ across the latter half of training checkpoints and find Spearman $\rho = 0.86$ between $H_{\mathrm{ref},t}$ and $\hat{\sigma}_t^2$ (Figure 2b–c), confirming this relationship. However, a binned analysis controlling for $|\hat{r}_t|$ reveals that the correlation is strongest at low-reward positions and diminishes at high-reward positions, indicating a nonlinear interaction between the two signals. This motivates a calibration network that learns the mapping jointly (Section 3.3), rather than fixing the functional form $c_t \propto |\hat{r}_t|/H_{\mathrm{ref},t}$.

On potential-based shaping. The token-level implicit reward is not unique under arbitrary potential-based shaping (Rafailov et al., 2024). Se-DPO does not claim invariance to all such decompositions. Instead, it operates under the canonical token decomposition induced by the DPO log-ratio in Eq. 3, which is reproducible for a fixed policy–reference pair and provides a practical credit signal. Our ablations in Section 4.3 support this convention empirically: using $|\hat{r}_t|$ alone already improves over uniform DPO, while the large gap between Static-Credit and Se-DPO shows that tracking the evolution of this canonical signal is more important than merely using a fixed token-level decomposition.

3.3 Credit Calibration Network

Proposition 1 suggests credit of the form $c_t \propto |\hat{r}_t|/H_{\mathrm{ref},t}$, but the proxy relationships $|r_t^*| \approx |\hat{r}_t|$ and $\sigma_t^2 \approx H_{\mathrm{ref},t}$ are approximate and interact nonlinearly (Section 3.2). Moreover, the closed-form ratio is numerically unstable at low-entropy positions where $\sigma_t^2 \to 0$, producing arbitrarily large credits for tokens with negligible reward. To account for these issues, we parameterize the credit function as a lightweight MLP f_ϕ that maps the two signals to a scalar credit:

$$\tilde{c}_t^{(s)} = f_\phi\left(|\hat{r}_t^{(s)}|,\ H_{\mathrm{ref},t}^{(s)}\right), \quad s \in \{c, r\}, \tag{11}$$

where ϕ is shared across all token positions and both response sides (architecture details in Appendix C).

Without constraint, the credit function could degenerately shrink all credits toward zero, which would trivially reduce the DPO loss. To prevent this, we normalize credits to have unit mean within each response, matching the total credit of standard DPO while allowing redistribution across tokens.

3.4 Se-DPO: Complete Algorithm

We now describe the complete Se-DPO algorithm, which integrates the components above. During the first W steps (the *warmup phase*), Se-DPO uses standard DPO with uniform credit ($c_t = 1$), avoiding amplification of early noise when $|\hat{r}_t| \approx 0$ for all tokens. From step $W+1$ onward (the *guided phase*), at each training step Se-DPO computes $|\hat{r}_t^{(s)}|$ and $H_{\mathrm{ref},t}^{(s)}$ from the current forward pass through the policy and reference models. Since DPO already evaluates both models on each batch, these quantities can be derived from the existing forward pass

Table 1: Experiment results on AlpacaEval 2 (Li et al., 2023), Arena-Hard (Li et al., 2024), and MT-Bench benchmarks. Best results in each setting are **bolded**.

Method	Llama-3-8B-Instruct (PairRM)				Llama-3-8B-Instruct (ArmoRM)			
	AlpacaEval 2	Arena-Hard	MT-Bench		AlpacaEval 2	Arena-Hard	MT-Bench	
	WR (%)	WR (%)	Score	WR (%)	WR (%)	WR (%)	Score	WR (%)
SFT	30.6	21.4	7.9	27.5	30.6	21.4	7.9	27.5
DPO	41.7	30.4	8.0	37.5	40.8	36.2	**8.2**	**46.3**
SimPO	39.8	28.7	7.8	32.5	37.0	28.1	7.8	42.5
Static-Credit	42.3	31.2	8.0	37.4	41.2	38.3	8.1	45.6
TGDPO	43.9	34.3	8.0	**41.9**	42.5	40.5	7.9	45.0
Se-DPO	**47.2**	**42.6**	7.4	**41.9**	**50.6**	**43.3**	6.9	40.0

Method	Llama-3.2-3B-Instruct (ArmoRM)				Gemma-2-2B-it (ArmoRM)			
	AlpacaEval 2	Arena-Hard	MT-Bench		AlpacaEval 2	Arena-Hard	MT-Bench	
	WR (%)	WR (%)	Score	WR (%)	WR (%)	WR (%)	Score	WR (%)
SFT	23.8	17.1	7.0	16.3	32.8	20.1	7.9	37.5
DPO	29.6	23.2	7.9	29.4	40.8	26.4	8.0	43.1
SimPO	26.2	22.6	7.4	15.7	34.8	21.1	7.8	40.0
Static-Credit	30.1	23.8	7.8	28.8	41.2	27.3	7.9	42.5
TGDPO	**35.8**	25.4	8.1	36.9	43.0	**30.7**	**8.1**	**46.9**
Se-DPO(LoRA)	34.2	**28.8**	**8.3**	**42.5**	**44.4**	29.6	7.9	42.3

with minimal overhead. The calibration network f_ϕ then maps $|\hat{r}_t^{(s)}|$ and $H_{\text{ref},t}^{(s)}$ to token credit via Eqs. 11.

Loss. The token credit modulates each token's contribution to the preference loss:

$$\mathcal{L}_{\text{Se-DPO}} = -\log \sigma \left(\sum_t c_t^{(c)} \, \hat{r}_t^{(c)} - \sum_t c_t^{(r)} \, \hat{r}_t^{(r)} \right), \tag{12}$$

Se-DPO adds only the forward and backward pass of the calibration network per training step. Both input signals—$|\hat{r}_t|$ and $H_{\text{ref},t}$—are derived from the standard DPO forward pass, requiring no additional model evaluations.

4 Experiments

4.1 Setup

Models and data. We conduct experiments on three models: Llama-3.2-3B-Instruct, Llama-3-8B-Instruct (Grattafiori et al., 2024), and Gemma-2-2B-it (Gemma Team, 2024). Following Zhu et al. (2025) and Meng et al. (2024), we use prompts from the UltraFeedback dataset (Cui et al., 2023) and let each model generate 5 responses with a temperature of 0.8. These responses are then ranked using ArmoRM (Wang et al., 2024), with the highest and lowest-ranked responses selected as the chosen and rejected samples, respectively. For Llama-3-8B-Instruct, we additionally use PairRM (Jiang et al., 2023) for preference annotation to evaluate robustness across different annotators. The data construction protocol is summarized in Appendix Table 7. Each dataset contains approximately 60K preference pairs. We compare against DPO (Rafailov et al., 2023), SimPO (Meng et al., 2024), TGDPO (Zhu et al., 2025), and Static-Credit, using the original method-specific settings whenever applicable. The SFT row refers to the original instruction-tuned model evaluated directly, without additional supervised fine-tuning. All methods are trained for 1 epoch with the AdamW optimizer and a cosine learning rate schedule with 10% warmup. Hyperparameter settings are presented in Appendix C.

Evaluation benchmarks. We use three prominent instruction-following benchmarks for evaluation: AlpacaEval 2 (Li et al., 2023), Arena-Hard (Li et al., 2024), and MT-Bench. For AlpacaEval 2, we report the win rate against GPT-4 Turbo. For Arena-Hard, we report the win rate against GPT-4-0314. For MT-Bench, we report the score and win rate against GPT-4.

Table 2: Warmup sensitivity on Llama-3.2-3B-Instruct with ArmoRM annotation.

Variant	AlpacaEval 2 WR (%)	AlpacaEval 2 LC WR (%)	Arena-Hard WR (%)
No warmup	32.57	33.03	22.1
20% warmup	31.15	31.50	22.8
50% warmup	32.08	32.53	25.2
4% warmup (default)	34.20	32.60	28.8

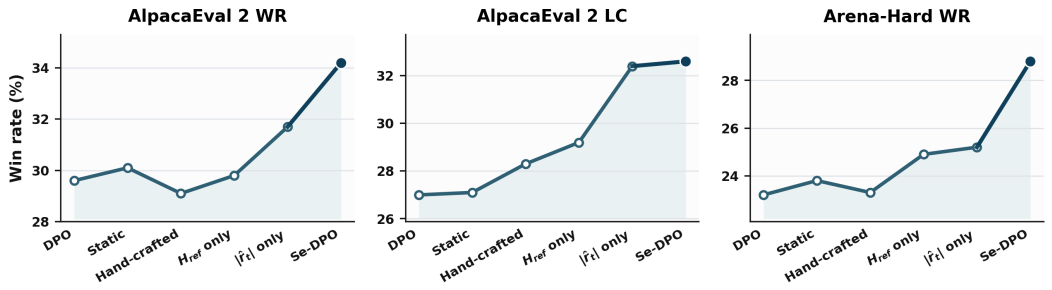


Figure 3: Ablation results on Llama-3.2-3B-Instruct. We compare DPO, Static-Credit, hand-crafted credit, single-signal variants, and Se-DPO across AlpacaEval 2 raw win rate, AlpacaEval 2 length-controlled win rate, and Arena-Hard win rate.

4.2 Main Results

Table 1 summarizes the results across three base models and two preference annotators. Se-DPO achieves the strongest or near-strongest performance on AlpacaEval 2 and Arena-Hard across most settings. On Llama-3-8B-Instruct with ArmoRM annotation, Se-DPO achieves 50.6% win rate on AlpacaEval 2 and 43.3% on Arena-Hard, surpassing the strongest prior baseline in Table 1 by +8.1 and +2.8 points; with PairRM annotation, Se-DPO similarly leads with 47.2% and 42.6%, improving over the strongest prior baseline in Table 1 by +3.3 and +8.3 points. Notably, these gains are achieved without any external model. Se-DPO also shows consistent improvements on AlpacaEval 2 and Arena-Hard under both PairRM and ArmoRM annotations, suggesting that the self-evolving credit mechanism is robust to the choice of preference annotator, as it derives importance signals entirely from the model's own internal states.

On smaller models (Llama-3.2-3B-Instruct and Gemma-2-2B-it), Se-DPO is trained with LoRA in the main table. Se-DPO (LoRA) consistently improves over standard DPO on AlpacaEval 2 and Arena-Hard, while remaining competitive with TGDPO across smaller-model settings. The margins are narrower than in the Llama-3-8B full fine-tuning setting, which suggests that parameter-efficient training may limit how precisely the policy can realize fine-grained, position-dependent deviations from the reference policy.

4.3 Ablation Studies

We conduct component ablations on Llama-3.2-3B-Instruct with ArmoRM annotation to isolate the contribution of online updating, the two input signals, warmup, and learned calibration (Figure 3). Static-Credit computes token credit once at the end of warmup and then keeps it frozen. It improves only modestly over DPO, confirming that a fixed per-token credit signal is useful but insufficient. In contrast, Se-DPO substantially outperforms Static-Credit, directly supporting our central claim that token credit should evolve with the policy during training.

The input ablations show that both implicit reward magnitude and reference entropy carry useful information. The hand-crafted ratio does not consistently improve over the single-signal variants, indicating that a fixed functional form is insufficient for capturing the relationship between reward strength and noise. This is consistent with our analysis in

Section 3.2: reference entropy is most informative at low-reward positions, while high-reward positions require a more regime-dependent mapping. We also find that high-credit tokens at convergence have near-zero rank correlation with the variance of $\hat{r}_t$ across the final checkpoints, suggesting that the calibration network does not simply amplify noisy tokens.

Warmup sensitivity. The Se-DPO credit warmup W is independent of the optimizer learning-rate warmup. During the first W steps, training uses the standard DPO loss and the calibration network is inactive; after W steps, credit-based KL allocation is enabled and f_ϕ is updated jointly with the policy. Table 2 shows that the default short warmup performs best on AlpacaEval 2 raw win rate and Arena-Hard, while remaining comparable on length-controlled AlpacaEval 2. Removing warmup or delaying credit activation reduces performance on these primary win-rate benchmarks, suggesting that Se-DPO benefits from a brief stabilization phase before online credit updates begin, while overly long warmups reduce the benefit of online credit adaptation.

Because LLM-as-a-judge benchmarks can be sensitive to response length, we report length-controlled AlpacaEval 2 scores and mean generation lengths in Appendix Table 5. On Llama-3-8B-Instruct, Se-DPO's average length is higher than DPO, but it retains a large length-controlled advantage over DPO and TGDPO, indicating that the gains are not explained by verbosity alone. We also observe that Se-DPO's MT-Bench score is lower than DPO in the Llama-3-8B full fine-tuning settings, and its MT-Bench win rate varies across annotators. Since MT-Bench score averages absolute ratings while win rate counts pairwise victories, improvements on pairwise instruction-following benchmarks may not always translate directly to MT-Bench scores; we leave this trade-off for future investigation.

5 Related Work

Reinforcement learning from human feedback. RLHF has become the dominant paradigm for aligning large language models with human preferences (Ouyang et al., 2022; Stiennon et al., 2020; Schulman et al., 2017). The standard pipeline first trains a reward model from human comparisons, then optimizes the policy via PPO. DPO (Rafailov et al., 2023) simplifies this pipeline by reparameterizing the reward into a supervised loss over preference pairs, eliminating the need for a separate reward model. This formulation has inspired a family of variants that modify the optimization objective along different axes: removing the reference model in favor of length-normalized implicit rewards (Meng et al., 2024), replacing pairwise preferences with prospect-theoretic pointwise feedback (Ethayarajh et al., 2024), adopting alternative divergence constraints (Azar et al., 2024), decomposing optimization to the reasoning-step level (Wang et al., 2025; Lai et al., 2024), or refining the optimization under autoregressive or small-margin settings (Oi et al., 2026; Pang et al., 2026). Orthogonally, the KL penalty coefficient β has been made adaptive at the sample level—via per-pair perturbation (Lee et al., 2025), batch-level adjustment based on data quality (Wu et al., 2024), instance-level weighting of preference pairs (Zhou et al., 2024), or adaptive batch-wise scheduling (Huang et al., 2025)—but these methods modulate regularization per sample rather than per token. Se-DPO extends adaptive regularization to token granularity by deriving per-token KL budgets from the model's own evolving signals.

Token-level rewards for preference optimization. Standard DPO aggregates token-level log-ratios via uniform summation, yet human judgments often hinge on a small subset of tokens. A growing body of work addresses this mismatch by introducing token-level importance signals into the preference loss. Several approaches estimate token importance from external models such as pre-trained teachers, contrastive LLM pairs, or discriminative Q-function models (Zhu et al., 2025; Liu et al., 2024; Zhou et al., 2025; Chen et al., 2025b). Others avoid external models but introduce auxiliary mechanisms—learned sparse masks, optimal transport, per-token KL constraints, gradient attribution, or token-adaptive barriers (Christopoulou et al., 2024; Li et al., 2025; Zeng et al., 2024; Yang et al., 2025b; Fodeh et al., 2026). Separately, DPO's implicit reward has been exploited at the sequence level for iterative self-alignment, token selection, PPO-based fine-tuning, process rewards, and token-level distillation (Chen et al., 2025a; Yang et al., 2025a; Zhong et al., 2025; Cui et al., 2025;

Table 3: Comparison of fine-grained preference optimization methods.

Method	Granularity	Signal Source	Extra Models?	Extra Cost
TGDPO	Token	Teacher model	Yes	Teacher training
TIS-DPO	Token	Contrastive LLMs	Yes	2 extra LLMs
SparsePO	Token	Loss-driven masks	No	$O(T)$ mask params
OTPO	Token	OT on embeddings	No	OT solver
Q-RM	Token	Discriminative model	Yes	Full model training
ε-DPO	Instance	Logit perturbation	No	Negligible
DICE	Sequence	Implicit reward	No	Multi-round training
Se-DPO (ours)	Token	Implicit reward	No	Minimal

Zhang et al., 2025). More broadly, iterative and feedback-driven optimization has been studied in both language-model alignment and scientific discovery settings (Chen et al., 2025a; Cui et al., 2025; Zhao et al., 2026; Li et al., 2026). Learning from ranking or preference signals also connects to broader work on robust optimization under noisy or imbalanced supervision (Azar et al., 2024; Ethayarajh et al., 2024; Yu et al., 2026). Across these directions, token importance is generally computed outside the training loop or extracted from a fixed checkpoint. Table 3 summarizes this comparison. Se-DPO differs in that it tracks the implicit reward online as it evolves during training and calibrates it with reference entropy for noise robustness, without requiring any external models.

6 Conclusion

We have identified a previously overlooked temporal dimension of token-level preference optimization: the implicit reward learned by DPO evolves substantially during training, with only 56% overlap between top-$|\hat{r}_t|$ tokens at early and late stages. This finding challenges the static importance assumption shared by existing token-level methods, which compute token-level importance signals prior to or outside the training loop. We formalized the notion of token credit as per-token KL budget allocation, and showed via a variance-minimization analysis that optimal credit should be proportional to implicit reward magnitude and inversely related to estimation noise. Since both quantities change as the model trains, effective credit must evolve with the model.

Based on this analysis, we proposed Se-DPO, which derives token credit online from two complementary signals—implicit reward magnitude and reference entropy—through a lightweight calibration network. Se-DPO requires no external models, no pre-trained teachers, no contrastive LLM pairs, and no complex solvers, adding only minimal computational overhead to standard DPO training. Experiments across three base models and two preference annotators demonstrate consistent improvements on AlpacaEval 2 and Arena-Hard, with Se-DPO achieving up to 50.6% win rate on AlpacaEval 2 and 43.3% on Arena-Hard, surpassing most baselines without external models.

Our analysis builds on two proxy relationships: $|\hat{r}_t|$ as a stand-in for the true reward magnitude and $H_{\text{ref},t}$ as an indicator of estimation noise. The two design choices in Se-DPO—the warmup phase and the calibration network—are specifically introduced to account for their approximate nature: the warmup phase ensures that $|\hat{r}_t|$ has stabilized before being used as credit, while the calibration network learns to handle the nonlinear interaction between the two signals rather than relying on a fixed functional form. Incorporating richer representations from the model's hidden states could potentially improve credit estimation.

References

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.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, 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.

Changyu Chen, Zichen Liu, Chao Du, Tianyu Pang, Qian Liu, Arunesh Sinha, Pradeep Varakantham, and Min Lin. Bootstrapping language models with DPO implicit rewards. In *ICLR*, 2025a.

Hongzhan Chen, Tao Yang, Shiping Gao, Ruijun Chen, Xiaojun Quan, Hongtao Tian, and Ting Yao. Discriminative policy optimization for token-level reward models. *arXiv preprint arXiv:2505.23363*, 2025b.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *NeurIPS*, 2017.

Fenia Christopoulou, Ronald Cardenas, Gerasimos Lampouras, Haitham Bou-Ammar, and Jun Wang. Sparsepo: Controlling preference alignment of llms via sparse token masks. *arXiv preprint arXiv:2410.05102*, 2024.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. Ultrafeedback: Boosting language models with high-quality feedback. 2023.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Yuchen Zhang, Jiacheng Chen, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, Jiarui Yuan, Huayu Chen, Kaiyan Zhang, Xingtai Lv, Shuo Wang, Yuan Yao, Xu Han, Hao Peng, Yu Cheng, Zhiyuan Liu, Maosong Sun, Bowen Zhou, and Ning Ding. Process reinforcement through implicit rewards, 2025. URL https://arxiv.org/abs/2502.01456.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

Samah Fodeh, Linhai Ma, Ganesh Puthiaraju, Srivani Talakokkul, Afshan Khan, Ashley Hagaman, Sarah R Lowe, and Aimee Kendall Roundtree. Tab-po: Preference optimization with a token-level adaptive barrier for token-critical structured generation. *arXiv preprint arXiv:2603.00025*, 2026.

Gemma Team. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Zixuan Huang, Yikun Ban, Lean Fu, Xiaojie Li, Zhongxiang Dai, Jianxin Li, and Deqing Wang. Adaptive batch-wise sample scheduling for direct preference optimization. *arXiv preprint arXiv:2506.17252*, 2025.

Hamish Ivison, Yizhong Wang, Jiacheng Liu, Zeqiu Wu, Valentina Pyatkin, Nathan Lambert, Noah A Smith, Yejin Choi, and Hannaneh Hajishirzi. Unpacking dpo and ppo: Disentangling best practices for learning from preference feedback. *Advances in neural information processing systems*, 37:36602–36633, 2024.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. Llm-blender: Ensembling large language models with pairwise ranking and generative fusion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14165–14178, 2023.

Xin Lai, Zhuotao Tian, Yukang Chen, Senqiao Yang, Xiangru Peng, and Jiaya Jia. Step-dpo: Step-wise preference optimization for long-chain reasoning of llms. *arXiv preprint arXiv:2406.18629*, 2024.

Sangkyu Lee, Janghoon Han, Hosung Song, Stanley Jungkyu Choi, Honglak Lee, and Youngjae Yu. Kl penalty control via perturbation for direct preference optimization. *arXiv preprint arXiv:2502.13177*, 2025.

Haobo Li, Eunseo Jung, Wenxiao Zhao, Feng Liu, Jiong Wang, Kaiyi Xu, Zijie Guo, Zixin Chen, Ben Fei, Fenghua Ling, and Lei Bai. Autosupervision: Closing the feedback loop in scientific workflows with grounded revision verification, 2026. URL https://arxiv.org/abs/2607.27845.

Meng Li, Guangda Huzhang, Haibo Zhang, Xiting Wang, and Anxiang Zeng. Optimal transport-based token weighting scheme for enhanced preference optimization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 21311–21334, 2025.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E Gonzalez, and Ion Stoica. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. *arXiv preprint arXiv:2406.11939*, 2024.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models, 2023.

Aiwei Liu, Haoping Bai, Zhiyun Lu, Yanchao Sun, Xiang Kong, Simon Wang, Jiulong Shan, Albin Madappally Jose, Xiaojiang Liu, Lijie Wen, et al. Tis-dpo: Token-level importance sampling for direct preference optimization with estimated weights. *arXiv preprint arXiv:2410.04350*, 2024.

Yu Meng, Mengzhou Xia, and Danqi Chen. Simpo: Simple preference optimization with a reference-free reward. In *Advances in Neural Information Processing Systems*, 2024.

Masanari Oi, Mahiro Ukai, Masahiro Kaneko, Naoaki Okazaki, and Nakamasa Inoue. Autoregressive direct preference optimization. *arXiv preprint arXiv:2602.09533*, 2026.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *NeurIPS*, 2022.

Jinlong Pang, Zhaowei Zhu, Na Di, Yichi Zhang, Yaxuan Wang, Chen Qian, and Yang Liu. Small-margin preferences still matter—if you train them right. *arXiv preprint arXiv:2602.00954*, 2026.

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 *NeurIPS*, 2023.

Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to q^*: Your language model is secretly a q-function. *arXiv preprint arXiv:2404.12358*, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback. *NeurIPS*, 2020.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 10582–10592, 2024.

Shu Wang, Lei Ji, Renxi Wang, Wenxiao Zhao, Haokun Liu, Yifan Hou, and Ying Nian Wu. Explore the reasoning capability of LLMs in the chess testbed. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, April 2025. doi: 10.18653/v1/2025.naacl-short.52. URL https://aclanthology.org/2025.naacl-short.52/.

Junkang Wu, Yuexiang Xie, Zhengyi Yang, Jiancan Wu, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. β-dpo: Direct preference optimization with dynamic β. 2024.

Kailai Yang, Zhiwei Liu, Qianqian Xie, Jimin Huang, Erxue Min, and Sophia Ananiadou. Selective preference optimization via token-level reward function estimation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 7043–7067, 2025a.

Ning Yang, Hai Lin, Yibo Liu, Baoliang Tian, Guoqing Liu, and Haijun Zhang. Token-importance guided direct preference optimization. *arXiv preprint arXiv:2505.19653*, 2025b.

Yanxuan Yu, Dong Liu, Eric Jiang, Shu Wang, Wenxiao Zhao, Jinxi Yu, Shaoyi Lu, Hui Pan, Renata Borovica-Gajic, and Ying Nian Wu. Rubric: Realism–utility balanced ranking for imbalanced classification, 2026. URL https://arxiv.org/abs/2607.09816.

Yongcheng Zeng, Guoqing Liu, Weiyu Ma, Ning Yang, Haifeng Zhang, and Jun Wang. Token-level direct preference optimization. In *ICML*, 2024.

Songming Zhang, Xue Zhang, Tong Zhang, Bojie Hu, Yufeng Chen, and Jinan Xu. Alignditstil: Token-level language model alignment as adaptive policy distillation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19791–19807, 2025.

Wenxiao Zhao, Dong Liu, Kaiyi Xu, Feng Liu, Zhen Zhao, Fei Ben, Shu Wang, Wenhao Li, Ying Nian Wu, Fenghua Ling, Haobo Li, and Lei Bai. A-sr: Self-evolving agentic llms for symbolic regression via hierarchical coordination, 2026. URL https://arxiv.org/abs/2608.04872.

Han Zhong, Zikang Shan, Guhao Feng, Wei Xiong, Xinle Cheng, Li Zhao, Di He, Jiang Bian, and Liwei Wang. Dpo meets ppo: Reinforced token optimization for rlhf. In *International Conference on Machine Learning*, pp. 78498–78521. PMLR, 2025.

Wenxuan Zhou, Ravi Agrawal, Shujian Zhang, Sathish Reddy Indurthi, Sanqiang Zhao, Kaiqiang Song, Silei Xu, and Chenguang Zhu. Wpo: Enhancing rlhf with weighted preference optimization. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8328–8340, 2024.

Wenxuan Zhou, Shujian Zhang, Lingxiao Zhao, and Tao Meng. T-reg: Preference optimization with token-level reward regularization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27876–27889, 2025.

Mingkang Zhu, Xi Chen, Zhongdao Wang, Bei Yu, Hengshuang Zhao, and Jiaya Jia. Tgdpo: Harnessing token-level reward guidance for enhancing direct preference optimization. *arXiv preprint arXiv:2506.14574*, 2025.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

A Proof of Proposition 1

Part (a). Since $c_t = \beta/\beta_t > 0$, we restrict to non-negative credits. Under independent noise, $\mathrm{Var}[\hat{\Delta}] = \sum_t c_t^2 \sigma_t^2$. We minimize this subject to $\sum_t c_t |r_t^*| = C$ with $c_t \geq 0$. The Lagrangian is

$$\mathcal{L} = \sum_t c_t^2 \sigma_t^2 - \lambda \left(\sum_t c_t |r_t^*| - C \right).$$

Setting $\partial \mathcal{L}/\partial c_t = 0$ yields $2c_t\sigma_t^2 = \lambda |r_t^*|$, so

$$c_t^* = \frac{\lambda\, |r_t^*|}{2\sigma_t^2} \;\geq\; 0,$$

which is automatically non-negative. Since credits are normalized to unit mean, the proportionality constant cancels: $c_t^* \propto |r_t^*|/\sigma_t^2$.

Part (b). For m-dependent noise, the full variance is

$$\mathrm{Var}[\hat{\Delta}] = \underbrace{\sum_t c_t^2 \sigma_t^2}_{V_{\text{diag}}} + \underbrace{\sum_t \sum_{t' : 0 < |t-t'| \leq m} c_t c_{t'} \mathrm{Cov}[\epsilon_t, \epsilon_{t'}]}_{V_{\text{cross}}}.$$

With bounded credits $c_t \in [c_{\min}, c_{\max}]$ and writing $\rho_{\max} = \max_{|t-t'| \leq m} |\mathrm{Corr}[\epsilon_t, \epsilon_{t'}]|$, the cross term satisfies

$$|V_{\text{cross}}| \leq 2m \cdot c_{\max}^2 \cdot \rho_{\max} \sum_t \sigma_t^2.$$

Meanwhile $V_{\text{diag}} \geq c_{\min}^2 \sum_t \sigma_t^2$. Hence $|V_{\text{cross}}|/V_{\text{diag}} \leq 2m\, \rho_{\max}\, (c_{\max}/c_{\min})^2$. When $m \ll T$ and the credit ratio $c_{\max}/c_{\min}$ is moderate (as enforced by mean normalization), the cross term is a bounded perturbation and the solution from Part (a) remains a valid first-order approximation.

B Additional Experimental Results

B.1 Training Cost

Se-DPO adds only a lightweight calibration network and reference-entropy computation on top of the standard DPO forward pass. Table 4 reports the wall-clock training time on Llama-3.2-3B-Instruct with ArmoRM annotation.

Table 4: Wall-clock training time on Llama-3.2-3B-Instruct with ArmoRM annotation.

Method	Training Time	Relative Overhead
DPO	97.1 min	–
Se-DPO	103.7 min	+6.85%

The overhead comes from the lightweight calibration network and reference entropy computation. Since both $|\hat{r}_t|$ and $H_{\text{ref},t}$ are derived from the standard DPO forward pass without additional model evaluations, this cost is substantially lower than methods requiring teacher, contrastive, or separate discriminative reward models.

B.2 MT-Bench Category Breakdown

Table 6 reports the category-level MT-Bench scores for Se-DPO on Llama-3-8B-Instruct with PairRM annotation. Se-DPO performs strongly on open-ended generation categories, while the lower overall mean is driven by Reasoning, Coding, and Math. This explains why MT-Bench score and MT-Bench win rate can diverge: the score averages absolute ratings, whereas win rate counts pairwise victories across questions.

Table 5: Length analysis across Llama-3-8B-Instruct and Llama-3.2-3B-Instruct with ArmoRM annotation. AE2 LC is the length-controlled AlpacaEval 2 win rate.

Model	Method	AE2 WR (%)	AE2 LC (%)	AE2 Avg. L	AH Avg. L
Llama-3-8B-Instruct	DPO	40.8	38.2	1837	621
Llama-3-8B-Instruct	SimPO	37.0	41.2	1825	583
Llama-3-8B-Instruct	TGDPO	42.5	40.1	1882	638
Llama-3-8B-Instruct	Se-DPO	50.6	47.4	2215	661
Llama-3.2-3B-Instruct	Se-DPO LoRA	34.2	32.6	2005	414

Table 6: MT-Bench category breakdown for Se-DPO on Llama-3-8B-Instruct with PairRM annotation.

Category	Score
Writing	9.62
Roleplay	9.40
STEM	9.47
Humanities	9.57
Extraction	8.10
Reasoning	4.50
Coding	4.60
Math	3.90

B.3 Token-Credit Behavior

We examine representative evaluation samples to understand the behavior of the learned credit. The resulting per-token KL budgets are context-dependent: in code-related responses, instruction tokens and variable names tend to receive larger credit than comment markers and section labels; in creative writing, character names and narrative-driving tokens tend to receive larger credit; in poetry, line breaks can receive elevated credit because they carry structural information. Quantitatively, the learned credit distribution is highly non-uniform: top-credited positions account for a larger share of the implicit-reward contribution and receive correspondingly larger KL budgets, while low-credit positions contribute less to the preference signal and are kept under tighter regularization. This supports the view that the calibration network learns to allocate token-level KL budgets according to informative implicit-reward patterns rather than applying uniform token-level regularization.

C Additional Experimental Details

Preliminary analysis (Section 3.1). The token importance evolution analysis in Section 3.1 is conducted on Qwen2.5-1.5B-Instruct with $\beta = 0.1$, learning rate 10^{-5}, trained for 1 epoch on 2,000 samples from UltraFeedback. We track 200 diagnostic samples across 15 evenly spaced checkpoints. The Se-DPO LoRA credit-dynamics probe in Figure 1(c–f) is conducted separately on fixed evaluation samples and is used only as a mechanism analysis of learned credit dynamics. The noise autocorrelation in Figure 2(a) is computed at three training stages (7%, 56%, 99% progress). The empirical reward variance $\hat{\sigma}_t^2$ in Figure 2(b–c) is estimated as the variance of $\hat{r}_t$ across the latter half of training checkpoints.

Credit calibration network. The calibration network f_ϕ is a two-layer MLP: Linear(2, 16) $\rightarrow$ ReLU $\rightarrow$ Linear(16, 1) $\rightarrow$ Softplus. The Softplus activation ensures non-negative output. The learning rate for ϕ is set to 10^{-3}, higher than the policy learning rate, to enable fast adaptation to the evolving implicit reward signal.

Training hyperparameters. All methods are trained for 1 epoch with the AdamW optimizer and a cosine learning rate schedule with 10% warmup ratio. For Llama-3-8B-Instruct,

Table 7: Preference-data construction and annotation protocol.

Component	Setting
Prompt source	UltraFeedback prompts
Candidate responses	5 responses per prompt generated by the target instruction-tuned model
Sampling temperature	0.8
Preference annotator	ArmoRM; PairRM additionally for Llama-3-8B-Instruct
Chosen / rejected pair	Highest-ranked response vs. lowest-ranked response
Training size	Approximately 60K preference pairs per setting

we use a learning rate of 5×10^{-7}, $\beta = 0.01$. For Llama-3.2-3B-Instruct and Gemma-2-2B-it, Se-DPO is trained with LoRA (rank 64, $\alpha = 128$, applied to all linear layers), with a learning rate of 5×10^{-6} and $\beta = 0.1$. The Se-DPO credit warmup W is set to approximately 4% of total training steps in the main experiments. DPO, SimPO, and TGDPO follow the hyper-parameters reported in Zhu et al. (2025); additional token-level baselines use their original method-specific settings whenever applicable.