ARENA AS OFFLINE REWARD: EFFICIENT FINE-GRAINED PREFERENCE OPTIMIZATION FOR DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning from human feedback (RLHF) effectively promotes preference alignment of text-to-image (T2I) diffusion models. To improve computational efficiency, direct preference optimization (DPO), which avoids explicit reward modeling, has been widely studied. However, its reliance on binary feedback limits it to coarse-grained modeling on chosen–rejected pairs, resulting in suboptimal optimization. In this paper, we propose ArenaPO, which leverages Arena scores as offline rewards to provide refined feedback, thus achieving efficient and fine-grained optimization without a reward model. This enables ArenaPO to benefit from both the rich rewards of traditional RLHF and the efficiency of DPO. Specifically, we first construct a model Arena in which each model's capability is represented as a Gaussian distribution, and infer these capabilities by traversing the annotated pairwise preferences. Each output image is treated as a sample from the corresponding capability distribution. Then, for a image pair, conditioned on the two capability distributions and the observed pairwise preference, the absolute quality gap is estimated using latent-variable inference based on truncated normal distribution, which serves as fine-grained feedback during training. It does not require a reward model and can be computed offline, thus introducing no additional training overhead. We conduct ArenaPO training on Pick-a-Pic v2 and HPD v3 datasets, and the results on various T2I benchmarks demonstrate that ArenaPO consistently outperforms existing baselines. Code will be publicly available.

1 INTRODUCTION

Text-to-image (T2I) diffusion models, such as DALLE (Betker et al., 2023), Stable Diffusion (Rombach et al., 2022; Podell et al., 2023), FLUX (Black Forest Labs, 2024), and PixArt (Chen et al., 2023), have demonstrated strong capabilities in generating high-quality and highly realistic images (Yang et al., 2023). Although large-scale pretraining endows these models with powerful generative abilities, post-training optimization is typically required to better align them with human preferences for real-world applications (Wallace et al., 2024; Xue et al., 2025). Consequently, in recent years, preference alignment approaches that incorporate human feedback into the diffusion training have been widely investigated.

One notable alignment paradigm is reinforcement learning from human feedback (RLHF) (Christiano et al., 2017; Stiennon et al., 2020). However, its efficiency has been widely criticized, largely because the reward model requires substantial resources for training and its deployment necessitates online inference. Therefore, most works on diffusion model alignment focus on direct preference optimization (DPO) (Wallace et al., 2024; Li et al., 2024; Hong et al., 2024; Zhu et al., 2025), a more efficient paradigm that directly leverages static offline datasets and does not require explicit reward modeling. Despite its high efficiency, DPO is limited to binary rewards on chosen–rejected image pairs, unlike the rich reward signals used in traditional RLHF, as illustrated in Figure 1. In practice, however, the quality differences between image pairs vary substantially (Li et al., 2025b), thus this coarse-grained reward modeling and optimization can potentially lead to suboptimal results.

In this work, we aim to explore how fine-grained preference can be incorporated into DPO in an offline manner without relying on an explicit reward model. Following

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Thurstone's Case V theory (Thurstone, 1927), we model each model's capability as a Gaussian distribution characterized by a mean and uncertainty, rather than a fixed point, and regard a specific output image as a sample drawn from this distribution. The labels in the dataset represent discretized binary relative preferences between the output images. Under this formulation, the difference between the model capability means serves as a *prior of overall quality*, while the binary preference acts as an *observation induced by capability uncertainty*. Consequently, the absolute quality gap between two output images can be estimated by a function with respect to the source models' capability means and uncertainties, as shown in Figure 2.

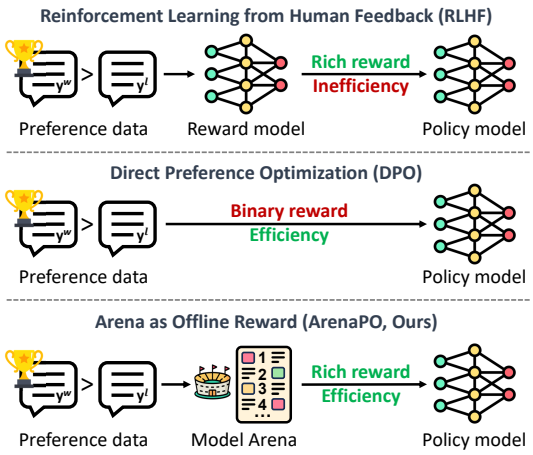


Figure 1: Comparison of RLHF, DPO, and ArenaPO. The proposed ArenaPO extracts and leverages rich offline rewards from model Arena, without requiring a reward model, enabling both fine-grained and efficient preference alignment.

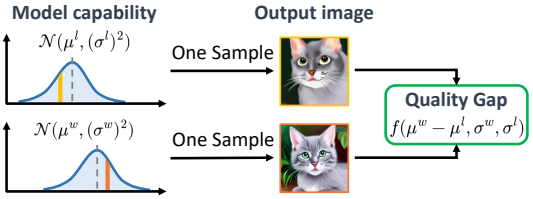


Figure 2: Motivation of the proposed ArenaPO. Each output image can be viewed as one sample drawn from the model's capability distribution $\mathcal{N}(\mu, \sigma^2)$. Thus, for two output images, their quality gap can be modeled as a function with respect to their source models' $\mu^w - \mu^l$, σ^w, and σ^l.

With the above insight, we propose ArenaPO, which leverages Arena information to provide fine-grained preference in an offline manner. ArenaPO can benefit from both the rich reward signals of traditional RLHF and the efficiency of DPO. Specifically, as in K-Sort Arena (Li et al., 2025b), we construct a model Arena and represent each model's capability as a Gaussian distribution. By traversing the entire dataset and performing pairwise comparisons, we iteratively update the capability distribution of each model. Then, for each image pair, given the corresponding capability distributions of the two models and the annotated binary preference, we introduce a latent-variable inference procedure based on truncated normal distribution to efficiently estimate the absolute quality difference between the two images. This estimated difference serves as a fine-grained preference during training. Notably, the entire pipeline does not rely on a reward model and can be computed fully offline, thereby incurring no additional training overhead. An overview of the proposed method is illustrated in Figure 3.

Our main contributions are summarized as follows:

- We propose ArenaPO, which extracts fine-grained preference from model Arena and uses it as efficient feedback for DPO, thereby effectively improving alignment performance for T2I diffusion models. ArenaPO does not rely on a reward model and can be computed offline, thus delivering both accuracy and efficiency.
- We introduce a latent-variable inference method based on truncated normal distribution, which enables efficient estimation of the absolute quality difference between an image pair given the models' capability distributions and the annotated binary preference.
- ArenaPO training is conducted on Pick-a-Pic v2 and HPD v3 datasets, and the results across multiple T2I benchmarks demonstrate ArenaPO's superiority. For instance, with ArenaPO training, the win-rate reaches 79.0% compared to original Stable Diffusion 1.5.

2 Related Work

T2I Diffusion Models T2I diffusion models have become the dominant paradigm due to their strong generation quality (Croitoru et al., 2023; Rombach et al., 2022). Recent works focus on improving model capacity and conditioning quality, including more powerful backbones (Peebles & Xie, 2023; Si et al., 2024), stronger text encoders (Chen et al., 2023), and more detailed captions (Chatterjee

et al., 2024). Advances in diffusion formulations and training objectives, such as improved parameterizations (Song & Ermon, 2019; Karras et al., 2022) and flow-based variants (Lipman et al., 2022; Liu et al., 2022), further enhance performance. Despite this, these pre-trained models typically require post-training to perform alignment optimization for real-world applications (Wallace et al., 2024).

Human Preference Alignment Preference optimization is initially developed for large language models (LLMs) to align the outputs with human preferences (Christiano et al., 2017; Stiennon et al., 2020). In the RLHF paradigm, human preferences are first used to train a reward model, and the target model is then optimized via reinforcement learning by maximizing the reward scores (Schulman et al., 2017). However, this pipeline is criticized for its inefficiency. DPO (Rafailov et al., 2023) provides a more efficient alternative by eliminating the need for a reward model and relying solely on offline preference dataset, and has thus been widely studied (Azar et al., 2024; Yuan et al., 2024).

Alignment of Diffusion Models Preference alignment for diffusion models has recently attracted increasing attention (Xue et al., 2025; Karthik et al., 2025). Most existing methods are built upon the DPO paradigm. For instance, Diffusion-DPO (Wallace et al., 2024) reformulates likelihood optimization as error modeling, resulting in a tractable training objective. DSPO (Zhu et al., 2025) leverages score matching to align the post-training objective with the pre-training objective. DMPO (Li et al., 2025a) performs preference alignment by optimizing the reverse KL divergence. MaPO (Hong et al., 2024) directly optimizes the likelihood margin between image pairs. Despite its efficiency advantages, DPO is limited by discrete binary preferences and cannot benefit from the rich reward signals in RLHF, thus leading to suboptimal optimization. In this work, we introduce fine-grained preferences extracted from Arena into DPO, enabling both efficient and more effective alignment.

3 PRELIMINARIES

Diffusion Models Diffusion models generate samples by learning to invert a progressive noise corruption process (Ho et al., 2020; Song et al., 2020). Let p_{data} denote the data distribution. Starting from a clean sample $\mathbf{x}_0 \sim p_{\text{data}}$, the forward diffusion process gradually injects Gaussian noise over T steps according to a predefined noise schedule $\beta_1, \ldots, \beta_T$. At an arbitrary timestep t, the resulting sample $\mathbf{x}_t$ follows:

$$\begin{aligned} q\left(\mathbf{x}_t \mid \mathbf{x}_0\right) &= \mathcal{N}\left(\sqrt{\bar{\alpha}_t}\mathbf{x}_0, \left(1-\bar{\alpha}_t\right) I\right), \\ \bar{\alpha}_t &= \prod_{s=1}^{t} \alpha_s, \quad \alpha_t = 1-\beta_t. \end{aligned} \tag{1}$$

To synthesize data, the reverse (denoising) dynamics are modeled using a neural network ϵ_θ, which is trained to predict the injected noise ϵ_t from a corrupted sample $\mathbf{x}_t$ at timestep t. Learning is carried out by minimizing a weighted denoising score matching objective (Ho et al., 2020):

$$\mathcal{L}_{\text{DM}}(\theta) = \mathbb{E}_{\mathbf{x}_0, \epsilon, t, \mathbf{x}_t}\Big[\omega(\lambda_t)\left\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\right\|_2^2\Big], \tag{2}$$

where $\lambda_t = \log(\alpha_t^2/\sigma_t^2)$ denotes the signal-to-noise ratio, and $\omega(\lambda_t)$ is a predefined weighting function.

DPO for Diffusion Models DPO (Rafailov et al., 2023) provides a simple framework for optimizing language models directly from preference data without relying on an explicit reward model. Given a prompt $\mathbf{c}$ and a preference-labeled pair $(\mathbf{x}_0^w, \mathbf{x}_0^l)$, DPO models preferences with the Bradley-Terry (BT) model (Bradley & Terry, 1952) as follow:

$$p_{\text{BT}}\left(\mathbf{x}_0^w \succ \mathbf{x}_0^l \mid \mathbf{c}\right) = \sigma\left(r\left(\mathbf{c}, \mathbf{x}_0^w\right) - r\left(\mathbf{c}, \mathbf{x}_0^l\right)\right), \tag{3}$$

where $r(\mathbf{c}, \mathbf{x}_0)$ is the latent reward, and $\sigma(\cdot)$ is the sigmoid function. DPO optimizes the target policy against a fixed reference policy, with the reward implicitly given by their log-likelihood ratio as follows:

$$\mathcal{L}_{\text{DPO}}(\theta) = -\mathbb{E}_{\mathbf{c}, \mathbf{x}_0^w, \mathbf{x}_0^l}\Big[\log\sigma\Big(\beta\log\frac{\pi_\theta(\mathbf{x}_0^w|c)}{\pi_{\text{ref}}(\mathbf{x}_0^w|c)} - \beta\log\frac{\pi_\theta(\mathbf{x}_0^l|c)}{\pi_{\text{ref}}(\mathbf{x}_0^l|c)}\Big)\Big], \tag{4}$$

where π_θ is the target policy to be optimized, and π_{ref} is a fixed reference policy (typically the pre-trained model). Here, $\log\sigma(\cdot)$ is the log-sigmoid function, and the hyperparameter β controls the optimization strength.

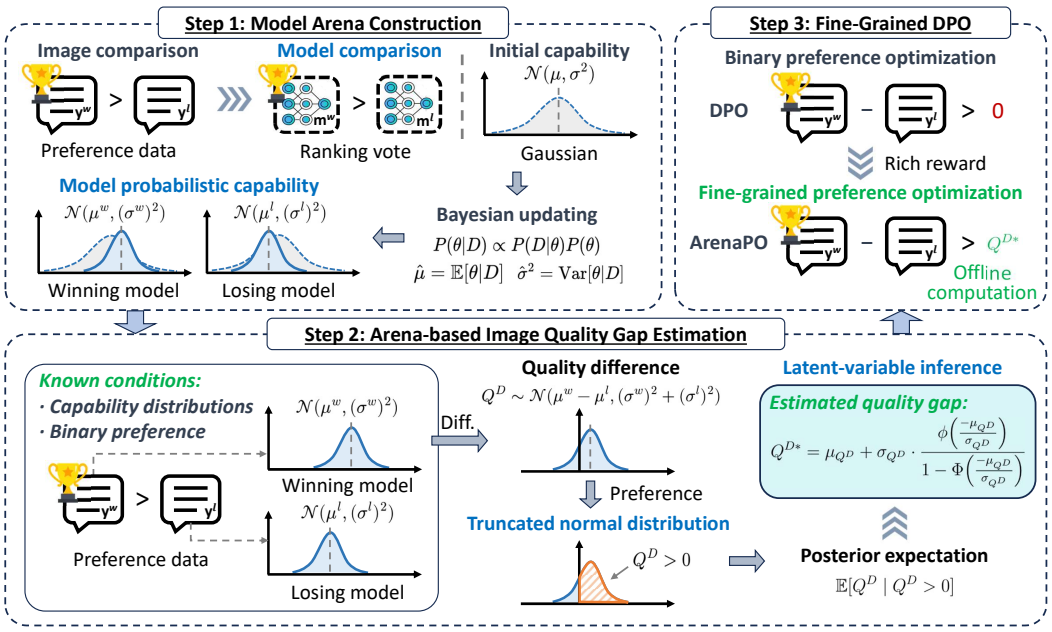


Figure 3: Overview of the proposed ArenaPO. First, we build a model Arena that converts image-level pairwise comparisons in the dataset into model-level comparisons, and updates each model's capability distribution using Bayesian inference. Second, for each data instance, the two images are regarded as samples from their corresponding capability distributions. Conditioned on the two distributions and the observed binary preference, we estimate the quality gap using latent-variable inference via truncated normal distribution. Finally, the estimated quality gap serves as an offline, fine-grained reward to enable more effective preference optimization.

Diffusion-DPO (Wallace et al., 2024) extends the DPO framework to diffusion models by interpreting the diffusion denoising objective $\mathcal{L}_{\text{DM}}$ as an implicit reward signal. The key idea is to quantify, at each timestep t, how much the fine-tuned model ϵ_θ improves over a reference model ϵ_{ref} in terms of denoising quality. Accordingly, a timestep-dependent reward for a sample x is defined as:

$$\delta_\theta(\mathbf{c}, \mathbf{x}_t, t) = -\Big(\|\epsilon - \epsilon_\theta(\mathbf{x}_t, \mathbf{c}, t)\|_2^2 - \|\epsilon - \epsilon_{\text{ref}}(\mathbf{x}_t, \mathbf{c}, t)\|_2^2 \Big). \tag{5}$$

Building on this per-timestep reward formulation, the resulting preference learning objective is obtained by aggregating contributions across the entire diffusion trajectory:

$$\mathcal{L}_{\text{Diff-DPO}}(\theta) = -\mathbb{E}_{\mathbf{c},\mathbf{x}_0^w,\mathbf{x}_0^l,t}\Big[\log \sigma\Big(\beta T \omega(\lambda_t)(\delta_\theta(\mathbf{c}, \mathbf{x}^w, t) - \delta_\theta(\mathbf{c}, \mathbf{x}^l, t)) \Big) \Big], \tag{6}$$

where T denotes the total number of diffusion steps, and $\omega(\lambda_t)$ is a weighting function that controls the relative importance of different timesteps.

4 Methodology

4.1 Model Arena Construction

In the preference dataset, each entry consists of a pair of images and a binary label. In addition, the model source of each image is recorded. Therefore, the dataset can also be viewed as the result of large-scale pairwise comparisons among multiple models. Based on this observation, we construct an Arena that includes these models, and, as in the K-Sort Arena (Li et al., 2025b), the capability of each model is probabilistically modeled as follows:

$$\theta \sim \mathcal{N}(\mu, \sigma^2), \tag{7}$$

where μ and σ represent the mean and uncertainty of model capability, respectively, and $\mathcal{N}(\cdot)$ denotes the normal distribution. During each round of pairwise comparison, the current capability distribution

$P(\theta)$ of a model is treated as the prior, whereas the binary label defines the likelihood function $P(D|\theta)$ of the observation D given θ. Based on Bayes' theorem, the posterior distribution of the capability, denoted as $P(\theta|D)$, is then updated as follows:

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{\int_{-\infty}^{\infty} P\left(D|\theta'\right) P\left(\theta'\right) d\theta'} = \frac{P(D|\theta)P(\theta)}{C}. \tag{8}$$

Given the posterior distribution, the mean and variance of the model capability are calculated as follows:

$$\begin{aligned} \hat{\mu} &= \mathbb{E}[\theta|D] = \textstyle\int_{-\infty}^{\infty} \theta P(\theta|D) d\theta, \\ \hat{\sigma}^2 &= \mathrm{Var}[\theta|D] = \textstyle\int_{-\infty}^{\infty} (\theta - \mathbb{E}[\theta|D])^2 P(\theta|D) d\theta. \end{aligned} \tag{9}$$

The detailed derivations of $\hat{\mu}$ and $\hat{\sigma}^2$ are provided in Appendix A. Based on the above equations, by iterating over the entire dataset, the capability distribution of each model is continuously updated. As proved in K-Sort Arena, μ and σ can eventually converge to values that accurately characterize the true capability distribution of the model.

4.2 Quality Gap Estimation from Arena

Our goal is to estimate the absolute quality gap between two images in each dataset instance based on the available information, without relying on a reward model. Specifically, the available information includes: (i) the source models' capability distributions extracted by Arena, parameterized as $\theta^w \sim \mathcal{N}(\mu^w, (\sigma^w)^2)$ and $\theta^l \sim \mathcal{N}(\mu^l, (\sigma^l)^2)$, and (ii) a human-annotated binary preference label $\mathbf{x}_0^w \succ \mathbf{x}_0^l$. The quality of each individual image, denoted by Q^w and Q^l, is obtained by a single draw from θ^w and θ^l, respectively. In this case, the problem can be formalized as follows:

Problem 1 (Arena-based Image Quality Gap Estimation)**.** *Consider two models whose capability distributions are $\theta^w \sim \mathcal{N}(\mu^w, (\sigma^w)^2)$ and $\theta^l \sim \mathcal{N}(\mu^l, (\sigma^l)^2)$. Let Q^w and Q^l be the output qualities obtained by drawing once from θ^w and θ^l, respectively. Given the observed event $Q^w > Q^l$, calculate the posterior expectation of the quality gap $\mathbb{E}[Q^w - Q^l \mid Q^w > Q^l]$.*

To solve the above problem, we proceed with the following derivation. Since Q^w and Q^l are independent Gaussian random variables, their difference $Q^D = Q^w - Q^l$ also follows a Gaussian distribution:

$$\begin{aligned} Q^D &\sim \mathcal{N}(\mu_{Q^D}, \sigma_{Q^D}^2), \\ \text{where} \quad \mu_{Q^D} &= \mu^w - \mu^l, \quad \sigma_{Q^D}^2 = (\sigma^w)^2 + (\sigma^l)^2. \end{aligned} \tag{10}$$

In each instance of preference datasets, we have the observed outcome $Q^w > Q^l$. This observation is equivalent to conditioning on the event $Q^D > 0$. Consequently, the estimation of the quality gap reduces to computing the conditional expectation of a Gaussian random variable under a one-sided truncation, i.e., $\mathbb{E}[Q^D \mid Q^D > 0]$.

Theorem 1 (Expectation the Truncated Normal Distribution)**.** *By the properties of the truncated normal distribution (Johnson et al., 1995), for a random variable $X \sim \mathcal{N}(\mu, \sigma^2)$, the conditional expectation under the constraint $X > a$ admits the following closed-form expression:*

$$\begin{aligned} \mathbb{E}[X \mid X > a] &= \mu + \sigma\, \lambda(\alpha), \\ \textit{where} \quad \alpha &= \frac{a - \mu}{\sigma}, \quad \lambda(\alpha) = \frac{\phi(\alpha)}{1 - \Phi(\alpha)}. \end{aligned} \tag{11}$$

Here, $\lambda(\alpha)$ is the inverse Mills ratio, and $\phi(\cdot)$ and $\Phi(\cdot)$ denote the probability density function and cumulative distribution function of the standard normal distribution, respectively. The derivation is presented in Appendix B.

Latent-Variable Inference via Truncated Normal Distribution With Theorem 1, by applying the result to the present case with $X = Q^D$ and $a = 0$, we can obtain the posterior expectation of the

quality gap conditioned on the observed comparison as follows:

$$
\begin{aligned}
Q^{D*} = \mathbb{E}[Q^D \mid Q^D > 0] &= \mu_{Q^D} + \sigma_{Q^D} \cdot \frac{\phi\left(\frac{-\mu_{Q^D}}{\sigma_{Q^D}}\right)}{1 - \Phi\left(\frac{-\mu_{Q^D}}{\sigma_{Q^D}}\right)} \\
&= (\mu^w - \mu^l) + \sqrt{(\sigma^w)^2 + (\sigma^l)^2} \cdot \frac{\phi\left(\frac{-(\mu^w - \mu^l)}{\sqrt{(\sigma^w)^2 + (\sigma^l)^2}}\right)}{1 - \Phi\left(\frac{-(\mu^w - \mu^l)}{\sqrt{(\sigma^w)^2 + (\sigma^l)^2}}\right)}.
\end{aligned} \tag{12}
$$

This result shows that the posterior expected quality gap consists of the prior mean difference augmented by a correction term induced by the selection event $Q^w > Q^l$. This quantity exhibits the following properties, making it suitable as a surrogate for the reward model. When the observed outcome is consistent with the prior ordering (i.e., $\mu^w \gg \mu^l$), the correction term becomes negligible and the posterior expectation is dominated by the prior mean difference. In contrast, when the outcome is unexpected (i.e., $\mu^w < \mu^l$), the correction term becomes dominant, enforcing a strictly positive expected gap consistent with the observed comparison.

4.3 DPO WITH FINE-GRAINED PREFERENCE

For each instance, the estimated Q^{D*} is treated as a fine-grained reward signal, enabling DPO to learn more fine-grained preferences. Specifically, instead of only using the ordering between the chosen and rejected responses, we incorporate Q^{D*} as a instance-specific margin that potentially reflects the preference intensity. The resulting BT model with a margin is formulated as follows:

$$
p_{\mathrm{BT}}\left(\mathbf{x}_0^w \succ \mathbf{x}_0^l \mid \mathbf{c}, Q^D\right) = \sigma\left(r\left(\mathbf{c}, \mathbf{x}_0^w\right) - r\left(\mathbf{c}, \mathbf{x}_0^l\right) - Q^{D*}\right). \tag{13}
$$

Then, we integrate this margin-based BT formulation into the Diffusion-DPO (Wallace et al., 2024) training objective, thereby extending standard DPO from enforcing a binary ordering constraint to matching a fine-grained preference signal. The resulting ArenaPO loss is as follows:

$$
\mathcal{L}_{\text{ArenaPO}}(\theta) = -\mathbb{E}_{\mathbf{c}, \mathbf{x}_0^w, \mathbf{x}_0^l, t}\left[\log \sigma\left(\beta T \omega(\lambda_t)(\delta_\theta(\mathbf{c}, \mathbf{x}^w, t) - \delta_\theta(\mathbf{c}, \mathbf{x}^l, t) - \gamma Q^{D*})\right)\right], \tag{14}
$$

where γ is a coefficient to align the scale of Q^{D*} with δ_θ. Compared to standard DPO, which only enforces a positive ordering between image pairs, ArenaPO further encourages the policy to match the magnitude of the preference gap. As a result, the model not only learns which response is better, but also how much better it is, thus achieving more fine-grained and effective preference optimization.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Models and Datasets We perform preference optimization training on Stable Diffusion 1.5 (Rombach et al., 2022) and Stable Diffusion XL (Podell et al., 2023). In addition to the commonly used Pick-a-Pic v2 (Kirstain et al., 2023) dataset, we also perform training on the recent HPD v3 (Ma et al., 2025) dataset, which includes a more diverse set of models and can further leverage the strengths of Arena.

Baseline Methods We thoroughly compare ArenaPO with existing representative baseline methods for diffusion preference optimization, including Diffusion-DPO (Wallace et al., 2024), MaPO (Hong et al., 2024), DSPO (Zhu et al., 2025), and SDPO (Fu et al., 2025). All results on HPD v3 dataset are reproduced using the official open-source implementations with the original hyperparameters.

Evaluations We adopt multiple preference metrics to verify model performance, including PickScore (Kirstain et al., 2023), HPS v2 (Wu et al., 2023), Aesthetic (Schuhmann et al., 2022), CLIP (Radford et al., 2021), and ImageReward (Xu et al., 2023), where prompts are from Pick-a-Pic v2 (Kirstain et al., 2023), HPS v2 (Wu et al., 2023), and PartiPrompts (Yu et al., 2022), respectively. We also conduct evaluations on human preference alignment benchmark K-Sort Eval (Li et al.).

Implementation Details For the Arena construction, we follow the implementation of K-Sort Arena. During training, we incorporate fine-grained preferences into Diffusion-DPO. To ensure a fair

comparison, all hyperparameters are kept identical to those used in Diffusion-DPO. Specifically, all models are fine-tuned for 2,000 steps with a global batch size of 2,048, and the learning rates are set to 1×10^{-8} and 1×10^{-9} for Stable Diffusion 1.5 and Stable Diffusion XL, respectively. After the grid search in section 5.5, the hyperparameter γ is set to 1×10^{-3} for all experiments.

5.2 Arena Results

We first construct a model Arena by treating each image comparison as a competition between the corresponding models. By iterating over the dataset, we obtain the capability distribution (μ, σ^2) of each model. The results on Pick-a-Pic v2 and HPD v3 datasets are in Appendix C.

Table 1 presents examples of estimated quality gaps. When the winning model is intrinsically stronger than the losing model, i.e., its capability mean μ is higher, the estimated gap is greater than 1 and increases with their difference. Conversely, when the winning model has a lower μ, the gap falls between 0 and 1 and decreases as the difference grows.

Table 1: Estimated quality gaps between two images based on model capabilities in Arena.

Winning Model (μ/σ)		Losing Model (μ/σ)		Est. Gap
FLUX.1-dev	(28.21/0.80)	HunyuanDiT	(23.41/0.77)	4.8038
Kolors	(28.54/0.81)	Real-Images	(24.55/0.78)	3.9928
HunyuanDiT	(23.41/0.77)	SDXL	(21.55/0.78)	1.9662
SD 1-1	(22.80/0.80)	SD 2	(22.67/0.78)	0.9406
SD 1-4	(22.97/0.83)	VQ-Diffusion	(26.59/0.82)	0.3211

5.3 Main Results of Training on Pick-a-Pic v2 Dataset

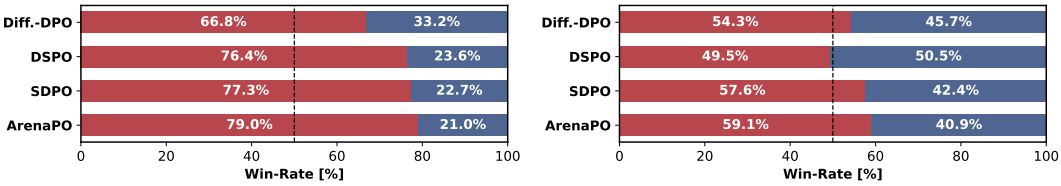
The training results on Pick-a-Pic v2 dataset are reported in Table 2. In most cases, ArenaPO achieves the best overall performance across different datasets and backbones. On Pick-a-Pic v2 benchmark with Stable Diffusion 1.5, ArenaPO outperforms all baselines on all five metrics, improving PickScore from 0.2143 (SDPO) to 0.2151 and ImageReward from 0.5546 to 0.6063. Moreover, Figure 4 presents the win-rate results, and ArenaPO consistently demonstrates an advantage.

Table 2: Comparison of preference optimization performance using different methods on Pick-a-Pic v2 dataset. The pretrained backbones include Stable Diffusion 1.5 and Stable Diffusion XL.

Dataset	Method	Stable Diffusion 1.5					Stable Diffusion XL				
		PickScore	HPS	Aes.	CLIP	IR	PickScore	HPS	Aes.	CLIP	IR
Pick v2	Pre-Trained	0.2073	0.2651	5.3907	0.3299	-0.1376	0.2242	0.2846	5.9970	0.3684	0.7382
	SFT	0.2128	0.2765	5.6888	0.3408	0.5767	0.2183	0.2809	5.7922	0.3658	0.5974
	MaPO	0.2097	0.2702	5.5572	0.3365	0.2435	0.2242	0.2871	**6.0979**	0.3684	0.8359
	Diff.-DPO	0.2109	0.2690	5.4958	0.3357	0.1020	0.2251	0.2868	6.0115	0.3732	0.8357
	DSPO	0.2131	0.2769	5.6825	0.3428	0.5642	0.2228	0.2834	5.8797	0.3756	0.8818
	SDPO	0.2143	0.2772	5.7172	0.3458	0.5546	0.2257	0.2876	5.9812	0.3746	0.8840
	ArenaPO (ours)	**0.2151**	**0.2781**	**5.7317**	**0.3480**	**0.6063**	**0.2268**	**0.2883**	6.0260	**0.3764**	**0.9172**
HPS v2	Pre-Trained	0.2088	0.2697	5.4933	0.3480	-0.0469	0.2290	0.2900	6.1271	0.3847	0.9047
	SFT	0.2168	0.2838	5.7851	0.3591	0.6619	0.2228	0.2883	5.9689	0.3806	0.8528
	MaPO	0.2124	0.2760	5.6890	0.3528	0.3308	0.2293	0.2934	**6.1882**	0.3840	0.9703
	Diff.-DPO	0.2131	0.2743	5.6639	0.3552	0.1705	0.2288	0.2927	6.1380	0.3840	1.0159
	DSPO	0.2168	0.2837	5.8346	0.3598	0.6483	0.2273	0.2916	6.0424	0.3894	1.0054
	SDPO	0.2174	0.2827	**5.8744**	0.3600	0.6211	0.2308	0.2938	6.1284	0.3879	1.0326
	ArenaPO (ours)	**0.2184**	**0.2842**	5.8591	**0.3619**	**0.6881**	**0.2319**	**0.2945**	6.1685	**0.3899**	**1.0438**
PartiPrompts	Pre-Trained	0.2144	0.2724	5.3466	0.3343	0.0637	0.2277	0.2880	5.7901	0.3591	0.8573
	SFT	0.2181	0.2821	5.5981	0.3389	0.5830	0.2221	0.2834	5.6496	0.3559	0.7515
	MaPO	0.2152	0.2754	5.4754	0.3366	0.3358	0.2278	0.2902	5.8921	0.3580	0.9324
	Diff.-DPO	0.2167	0.2755	5.4045	0.3391	0.2560	0.2279	0.2900	5.8294	0.3629	1.0638
	DSPO	0.2178	0.2819	5.5997	0.3385	0.5640	0.2261	0.2871	5.6947	0.3664	1.0514
	SDPO	0.2187	0.2815	5.5880	0.3423	0.5425	0.2290	**0.2907**	5.7882	0.3645	1.0654
	ArenaPO (ours)	**0.2198**	**0.2833**	**5.6164**	**0.3445**	**0.6291**	**0.2299**	0.2906	**5.9003**	**0.3671**	**1.0672**

5.4 Main Results of Training on HPD v3 Dataset

To further exploit the strengths of ArenaPO, we also train it on the more recent HPD v3 dataset. The results are presented in Table 3. Since this is a newly released dataset, the performance of all baselines is reproduced using the official implementations and hyperparameter settings. These results demonstrate that ArenaPO consistently achieves the best performance across all three benchmarks.



(a) Stable Diffusion 1.5 (b) Stable Diffusion XL

Figure 4: Win-rates of models trained using different methods on Pick-a-Pic v2 dataset versus the original model. The test prompts are from HPS v2. The results are the mean values across PickScore, HPS, Aesthetic, CLIP, and ImageReward.

Table 3: Comparison of preference optimization performance using different methods on HPD v3 dataset. The pretrained backbones include Stable Diffusion 1.5 and Stable Diffusion XL.

Dataset	Method	Stable Diffusion 1.5					Stable Diffusion XL				
		PickScore	HPS	Aes.	CLIP	IR	PickScore	HPS	Aes.	CLIP	IR
Pick v2	Pre-Trained	0.2073	0.2651	5.3907	0.3299	-0.1376	0.2242	0.2846	5.9970	0.3684	0.7382
	SFT	0.2119	0.2773	5.6673	0.3419	0.5852	0.2203	0.2828	5.9062	0.3668	0.6427
	MaPO	0.2099	0.2760	5.6523	0.3397	0.4894	0.2239	0.2859	5.9862	0.3679	0.7181
	Diff.-DPO	0.2127	0.2777	5.6628	0.3427	0.5089	0.2255	0.2874	6.0273	0.3739	0.8413
	DSPO	0.2140	0.2782	5.7231	0.3487	0.5825	0.2248	0.2873	6.0288	0.3763	0.9032
	SDPO	0.2145	0.2789	5.7356	0.3482	0.5796	0.2261	0.2879	6.0295	0.3760	0.8983
	ArenaPO (ours)	**0.2164**	**0.2815**	**5.7716**	**0.3522**	**0.6337**	**0.2280**	**0.2889**	**6.0311**	**0.3796**	**0.9257**
HPS v2	Pre-Trained	0.2088	0.2697	5.4933	0.3480	-0.0469	0.2290	0.2900	6.1271	0.3847	0.9047
	SFT	0.2189	0.2829	5.7636	0.3573	0.6078	0.2267	0.2879	6.1084	0.3823	0.8749
	MaPO	0.2180	0.2792	5.7521	0.3557	0.5111	0.2286	0.2891	6.1197	0.3840	0.8934
	Diff.-DPO	0.2188	0.2817	5.7837	0.3613	0.5837	0.2298	0.2941	6.1418	0.3871	1.0192
	DSPO	0.2194	0.2844	5.8728	0.3602	0.6591	0.2309	0.2930	6.1397	0.3867	1.0148
	SDPO	0.2199	0.2851	5.8706	0.3629	0.6628	0.2315	0.2943	6.1572	0.3886	1.0363
	ArenaPO (ours)	**0.2215**	**0.2882**	**5.8993**	**0.3659**	**0.6974**	**0.2328**	**0.2954**	**6.1696**	**0.3894**	**1.0463**
PartiPrompts	Pre-Trained	0.2144	0.2724	5.3466	0.3343	0.0637	0.2277	0.2880	5.7901	0.3591	0.8573
	SFT	0.2191	0.2833	5.5862	0.3387	0.5668	0.2263	0.2880	5.7386	0.3538	0.8012
	MaPO	0.2175	0.2798	5.5638	0.3375	0.4729	0.2271	0.2875	5.7876	0.3583	0.8394
	Diff.-DPO	0.2182	0.2813	5.6210	0.3398	0.5132	0.2290	0.2905	5.8476	0.3628	0.9927
	DSPO	0.2190	0.2831	5.6237	0.3494	0.5825	0.2286	0.2902	5.8521	0.3626	1.0023
	SDPO	0.2194	0.2826	5.6244	0.3513	0.5986	0.2297	0.2910	5.8829	0.3649	1.0274
	ArenaPO (ours)	**0.2211**	**0.2846**	**5.6417**	**0.3528**	**0.6363**	**0.2308**	**0.2915**	**5.9062**	**0.3664**	**1.0429**

For instance, on HPS v2 with Stable Diffusion XL, ArenaPO pushes PickScore to 0.2328 and ImageReward to 1.0463, which outperforms the previous best SDPO.

5.5 Ablation Studies

Image Visualization of Different Methods Figure 5 presents a qualitative comparison of different methods. Compared to baseline methods, ArenaPO produces images with higher visual quality, better realism, and stronger alignment with human preferences. Please see Appendix D for more results.

Image Visualization of Different Training Steps Figure 6 presents a qualitative comparison of different training steps. We observe that as training proceeds, the generated images exhibit progressively more accurate semantic content and continuously improved visual quality, which verifies the effectiveness of the training process.

ArenaPO Combined with Advanced Methods Since the offline fine-grained rewards of ArenaPO is decoupled from the DPO optimization strategy, it can be seamlessly integrated with more advanced methods. As shown in Table 4, the combination of ArenaPO and DSPO further yields performance gains, which validates the strong scalability and general applicability of our approach.

ArenaPO vs. Uniform Margin The reward in ArenaPO is incorporated in the form of a margin. Therefore, we also compare it with fixed-margin baselines, as shown in Table 5, where the fixed margin is the replacement of Q^D. It can be seen that the uniform margin yields only limited



Figure 5: Qualitative comparison of different methods for training Stable Diffusion 1.5 on Pick-a-Pic v2 dataset. Prompt: *1) Cute grey cat, digital oil painting by Monet. 2) A pale half-elf in a dark, silver-trimmed robe, with long hair partly tied back. 3) An astronaut floating in space, with Earth dazzling starlight shining in the background.*



Figure 6: Qualitative comparison of different training steps for training Stable Diffusion 1.5 on Pick-a-Pic v2 dataset. Prompt: *1) Photograph of medieval warrior, with giant hammer, posing in battlefield. 2) A person stands on the edge of a cliff, overlooking majestic mountains and a deep valley. 3) The iron man in action, candid shot, capturing the movement.*

Table 4: Preference optimization performance of ArenaPO combined with advanced methods. The reported results are obtained by training Stable Diffusion 1.5 on Pick-a-Pic v2 dataset.

Method	PickScore	HPS	Aes.	CLIP	IR
Pre-Trained	0.2088	0.2697	5.4933	0.3480	-0.0469
SFT	0.2168	0.2838	5.7851	0.3591	0.6619
MaPO	0.2124	0.2760	5.6890	0.3528	0.3308
Diff.-DPO	0.2131	0.2743	5.6639	0.3552	0.1705
DSPO	0.2168	0.2837	5.8346	0.3598	0.6483
SDPO	0.2174	0.2827	5.8744	0.3600	0.6211
ArenaPO	0.2184	0.2842	5.8591	0.3619	0.6881
ArenaPO+DSPO	**0.2191**	**0.2851**	**5.8818**	**0.3627**	**0.6914**

Table 5: Comparison of preference optimization performance between ArenaPO and methods with uniform margins. The reported results are obtained by training Stable Diffusion 1.5 on Pick-a-Pic v2 dataset.

Method	PickScore	HPS	Aes.	CLIP	IR
Pre-Trained	0.2088	0.2697	5.4933	0.3480	-0.0469
Margin=0.5	0.2147	0.2782	5.6976	0.3580	0.3727
Margin=1.0	0.2168	0.2830	5.8218	0.3587	0.6287
Margin=2.0	0.2174	0.2836	5.8476	0.3602	0.6375
Margin=3.0	0.2171	0.2833	5.8525	0.3599	0.6469
Margin=4.0	0.2158	0.2811	5.7235	0.3587	0.5724
ArenaPO	**0.2184**	**0.2842**	**5.8591**	**0.3619**	**0.6881**

improvements, while ArenaPO provides fine-grained and adaptive awards, resulting in significantly better performance. To improve the discriminability of the results, we also evaluate on the new benchmark K-Sort Eval (Li et al.), and the results are presented in Appendix E.

Effect of Scaling Coefficient γ To align the scale when incorporating Q^D, we introduce a scaling coefficient γ. We perform a grid search over γ, as shown in Figure 7. The final performance is insensitive to the choice of γ, and achieves the best overall result at 1×10^{-3}. Thus, for simplicity and general applicability, we fix this parameter to 1×10^{-3} across all experiments.

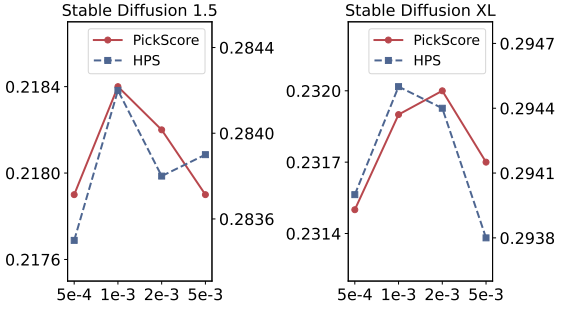


Figure 7: Grid search of the hyperparameter γ.

6 Conclusion

In this paper, we propose ArenaPO, an efficient preference optimization framework for diffusion models that extracts fine-grained preference from Arena without relying on an explicit reward model. By modeling each source model's capability as a Gaussian distribution, and interpreting binary annotations as observations induced by uncertainty, ArenaPO can estimate quality gaps between image pairs using latent-variable inference via truncated normal distribution. These fine-grained preferences are then seamlessly integrated into training, combining the efficiency of offline optimization with richer preference signals. Extensive experiments on Pick-a-Pic v2 and HPD v3 demonstrate consistent and significant improvements over existing DPO-based methods, validating the effectiveness and practicality of ArenaPO for diffusion model alignment.

AI USE STATEMENT

The authors did not use generative AI tools in the research or preparation of this work.

REFERENCES

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pp. 4447–4455. PMLR, 2024.

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

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

Agneet Chatterjee, Gabriela Ben Melech Stan, Estelle Aflalo, Sayak Paul, Dhruba Ghosh, Tejas Gokhale, Ludwig Schmidt, Hannaneh Hajishirzi, Vasudev Lal, Chitta Baral, et al. Getting it right: Improving spatial consistency in text-to-image models. In *European Conference on Computer Vision*, pp. 204–222. Springer, 2024.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-alpha: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(9): 10850–10869, 2023.

Minghao Fu, Guo-Hua Wang, Tianyu Cui, Qing-Guo Chen, Zhao Xu, Weihua Luo, and Kaifu Zhang. Diffusion-sdpo: Safeguarded direct preference optimization for diffusion models. *arXiv preprint arXiv:2511.03317*, 2025.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Jiwoo Hong, Sayak Paul, Noah Lee, Kashif Rasul, James Thorne, and Jongheon Jeong. Margin-aware preference optimization for aligning diffusion models without reference. In *First Workshop on Scalable Optimization for Efficient and Adaptive Foundation Models*, 2024.

Norman L Johnson, Samuel Kotz, and Narayanaswamy Balakrishnan. *Continuous univariate distributions, volume 2*, volume 2. John wiley & sons, 1995.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Shyamgopal Karthik, Huseyin Coskun, Zeynep Akata, Sergey Tulyakov, Jian Ren, and Anil Kag. Scalable ranked preference optimization for text-to-image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18399–18410, 2025.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Binxu Li, Minkai Xu, Jiaqi Han, Meihua Dang, and Stefano Ermon. Divergence minimization preference optimization for diffusion model alignment. *arXiv preprint arXiv:2507.07510*, 2025a.

Shufan Li, Konstantinos Kallidromitis, Akash Gokul, Yusuke Kato, and Kazuki Kozuka. Aligning diffusion models by optimizing human utility. *Advances in Neural Information Processing Systems*, 37:24897–24925, 2024.

Zhikai Li, Xuewen Liu, Wangbo Zhao, Pan Du, Kaicheng Zhou, Qingyi Gu, Yang You, Zhen Dong, Kurt Keutzer, et al. K-sort eval: Efficient preference evaluation for visual generation via corrected vlm-as-a-judge. In *The Fourteenth International Conference on Learning Representations*.

Zhikai Li, Xuewen Liu, Dongrong Joe Fu, Jianquan Li, Qingyi Gu, Kurt Keutzer, and Zhen Dong. K-sort arena: Efficient and reliable benchmarking for generative models via k-wise human preferences. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 9131–9141, 2025b.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. Hpsv3: Towards wide-spectrum human preference score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15086–15095, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

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. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Christoph Schuhmann, Romain Vencu, Romai Beaumont, et al. LAION-Aesthetics: Predicting the aesthetic quality of images. https://laion.ai/blog/laion-aesthetics/, 2022.

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

Chenyang Si, Ziqi Huang, Yuming Jiang, and Ziwei Liu. Freeu: Free lunch in diffusion u-net. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4733–4743, 2024.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021, 2020.

LL Thurstone. A law of comparative judgment. *Psychological Review*, 34(4):273–286, 1927.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8228–8238, 2024.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, et al. Dancegrpo: Unleashing grpo on visual generation. *arXiv preprint arXiv:2505.07818*, 2025.

Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM computing surveys*, 56(4):1–39, 2023.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E Weston. Self-rewarding language models. In *Forty-first International Conference on Machine Learning*, 2024.

Huaisheng Zhu, Teng Xiao, and Vasant G Honavar. Dspo: Direct score preference optimization for diffusion model alignment. In *The Thirteenth International Conference on Learning Representations*, 2025.

A Derivation of Bayesian Updating

Suppose that the observation D indicates that model M_1 outperforms model M_2. The latent performance of model follows a Gaussian distribution, $\theta_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$. Conditioned on the latent performance variables θ_1 and θ_2, the likelihood of this event is given by:

$$P(D \mid \theta_1, \theta_2) = P(X_1 > X_2) = \Phi\left(\frac{\theta_1 - \theta_2}{\sqrt{\beta_1^2 + \beta_2^2}}\right),$$

where $\Phi(\cdot)$ and $\phi(\cdot)$ denote the cumulative distribution function (CDF) and the probability density function (PDF) of the standard normal distribution, respectively:

$$\Phi(x) = \int_{-\infty}^{x} \phi(u)\, du, \qquad \phi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}.$$

Assume independent Gaussian priors $\theta_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$ and $\theta_2 \sim \mathcal{N}(\mu_2, \sigma_2^2)$. β represents the uncertainty in a model's single-instance performance. By Bayes' theorem, the joint posterior distribution can be written as:

$$P(\theta_1, \theta_2 \mid D) \propto \phi\left(\frac{\theta_1 - \mu_1}{\sigma_1}\right) \phi\left(\frac{\theta_2 - \mu_2}{\sigma_2}\right) \Phi\left(\frac{\theta_1 - \theta_2}{\sqrt{\beta_1^2 + \beta_2^2}}\right).$$

Marginalizing out θ_2 yields the posterior of θ_1:

$$P(\theta_1 \mid D) \propto \phi\left(\frac{\theta_1 - \mu_1}{\sigma_1}\right) \Phi\left(\frac{\theta_1 - \mu_2}{\sqrt{\beta_1^2 + \beta_2^2 + \sigma_2^2}}\right).$$

The posterior mean of θ_1 is then given by:

$$\begin{aligned}
\hat{\mu}_1 = \mathbb{E}[\theta_1 \mid D] &= \mu_1 + \frac{\sigma_1^2}{\sqrt{\sum(\beta_i^2 + \sigma_i^2)}} \frac{\phi\left(\frac{\mu_1 - \mu_2}{\sqrt{\sum(\beta_i^2 + \sigma_i^2)}}\right)}{\Phi\left(\frac{\mu_1 - \mu_2}{\sqrt{\sum(\beta_i^2 + \sigma_i^2)}}\right)} \\
&= \mu_1 + \frac{\sigma_1^2}{c_{12}} \mathcal{V}\left(\frac{\mu_1 - \mu_2}{c_{12}}\right),
\end{aligned}$$

where

$$\mathcal{V}(x) = \frac{\phi(x)}{\Phi(x)}, \qquad c_{ij}^2 = \sum(\beta_i^2 + \sigma_i^2).$$

Similarly, the posterior variance is updated as:

$$\begin{aligned}
\hat{\sigma}_1^2 = \mathrm{Var}[\theta_1 \mid D] &= \sigma_1^2 \left(1 - \frac{\sigma_1^2}{\sum(\beta_i^2 + \sigma_i^2)} \mathcal{W}\left(\frac{\mu_1 - \mu_2}{\sqrt{\sum(\beta_i^2 + \sigma_i^2)}}\right)\right) \\
&= \sigma_1^2 \left(1 - \frac{\sigma_1^2}{c_{12}^2} \mathcal{W}\left(\frac{\mu_1 - \mu_2}{c_{12}}\right)\right),
\end{aligned}$$

with

$$\mathcal{W}(x) = \mathcal{V}(x)\big(\mathcal{V}(x) + x\big).$$

This completes the Bayesian update of the performance distribution for a single pairwise comparison in which M_1 is observed to outperform M_2.

B Derivation of Expectation of Truncated Normal Distribution

Let $X \sim \mathcal{N}(\mu, \sigma^2)$ be a Gaussian random variable with probability density function:

$$f_X(x) = \frac{1}{\sigma}\phi\left(\frac{x-\mu}{\sigma}\right),$$

where $\phi(\cdot)$ denotes the standard normal PDF.

We consider the conditional distribution of X under the truncation constraint $X > a$. The conditional density is given by:

$$f_{X|X>a}(x) = \frac{f_X(x)}{P(X>a)} = \frac{\frac{1}{\sigma}\phi\left(\frac{x-\mu}{\sigma}\right)}{1-\Phi\left(\frac{a-\mu}{\sigma}\right)}, \qquad x > a.$$

The conditional expectation is therefore:

$$\mathbb{E}[X \mid X > a] = \int_a^\infty x\, f_{X|X>a}(x)\, dx = \frac{1}{1-\Phi(\alpha)} \int_a^\infty x\, \frac{1}{\sigma}\phi\left(\frac{x-\mu}{\sigma}\right) dx,$$

where we define:

$$\alpha = \frac{a-\mu}{\sigma}.$$

Apply the change of variables:

$$z = \frac{x-\mu}{\sigma}, \qquad x = \mu + \sigma z, \qquad dx = \sigma dz.$$

Then the integral becomes:

$$\mathbb{E}[X \mid X > a] = \frac{1}{1-\Phi(\alpha)} \int_\alpha^\infty (\mu + \sigma z)\, \phi(z)\, dz.$$

Splitting the integral, we obtain:

$$\mathbb{E}[X \mid X > a] = \frac{1}{1-\Phi(\alpha)} \left[\mu \int_\alpha^\infty \phi(z)\, dz + \sigma \int_\alpha^\infty z\phi(z)\, dz\right].$$

Using standard properties of the Gaussian distribution,

$$\int_\alpha^\infty \phi(z)\, dz = 1 - \Phi(\alpha),$$

and

$$\int_\alpha^\infty z\phi(z)\, dz = \phi(\alpha),$$

we arrive at:

$$\mathbb{E}[X \mid X > a] = \frac{1}{1-\Phi(\alpha)} \left[\mu(1-\Phi(\alpha)) + \sigma\phi(\alpha)\right].$$

Finally, simplifying yields:

$$\mathbb{E}[X \mid X > a] = \mu + \sigma \frac{\phi(\alpha)}{1-\Phi(\alpha)}.$$

Defining the inverse Mills ratio:

$$\lambda(\alpha) = \frac{\phi(\alpha)}{1-\Phi(\alpha)},$$

we obtain the desired closed-form expression:

$$\mathbb{E}[X \mid X > a] = \mu + \sigma\lambda(\alpha).$$

C ARENA RESULTS OF PICK-A-PIC V2 AND HPD V3 DATASETS

The results of model Arena for Pick-a-Pic v2 and HPD v3 datasets are in Table 6. It can be observed that Pick-a-Pic v2 primarily emphasizes differences among models with similar architectures, whereas HPD v3 focuses more on comparisons across a diverse set of models.

Table 6: Arena results of diffusion models in preference datasets.

(a) Pick-a-Pic v2 Training Set

Model	μ	σ	Score
Stable-Diffusion-XL-beta-v2-2-5-e	27.99	0.78	25.66
Stable-Diffusion-XL-beta-v2-2-5-c	27.32	0.77	25.00
Stable-Diffusion-XL-beta-v2-2-5-g	27.26	0.77	24.95
Stable-Diffusion-XL-beta-v2-2-3-5	26.79	0.78	24.45
Stable-Diffusion-XL-beta-v2-2-5-b	26.65	0.77	24.33
Stable-Diffusion-XL-beta-v2-2-5-f	26.57	0.77	24.25
Stable-Diffusion-XL-beta-v2-2-5-d	26.21	0.77	23.89
Stable-Diffusion-XL-beta-v2-2-5-h	25.90	0.77	23.59
Stable-Diffusion-XL-v2-2	24.69	0.77	22.37
Dreamlike-Photoreal-2-flax	24.54	0.77	22.24
Stable-Diffusion-XL-beta-v2-2-2	24.43	0.77	22.11
Stable-Diffusion-XL-beta-v2-2-3	24.14	0.78	21.82
Stable-Diffusion-2-1	24.08	0.78	21.75

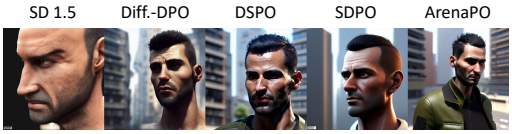
(b) HPD v3 Training Set

Model	μ	σ	Score
Kolors	28.54	0.81	26.11
FLUX.1-dev	28.21	0.80	25.82
Infinity	27.38	0.79	25.03
VQ-Diffusion	26.59	0.82	24.14
Real-Images	24.55	0.78	22.21
MidJourney	23.66	0.84	21.13
HunyuanDiT	23.41	0.77	21.10
Stable-Diffusion-3-Medium	23.01	0.77	20.69
Stable-Diffusion-1-4	22.97	0.83	20.48
Stable-Diffusion-1-1	22.80	0.80	20.38
Stable-Diffusion-2	22.67	0.78	20.34
Stable-Diffusion-XL	21.55	0.78	19.20
FuseDream	20.98	0.90	18.28
CogView2	20.11	0.80	17.71
Glide	14.91	0.87	12.31

D More Image Visualization of Different Methods



(a) Prompt: *Gothic cathedral in a stormy night.*



(b) Prompt: *30 year old short slim man, fuller round face, very short hair, black hair, black stubble, olive skin, immense detail/ hyper.*



(c) Prompt: *An anime girl, masterpiece, good line art, trending in pixiv.*



(d) Prompt: *A career woman on suits*



(e) Prompt: *A needle-felted palm tree.*



(f) Prompt: *Photograph of medieval warrior, with giant hammer, posing in battlefield.*



(g) Prompt: *Blonde german-italian girl smiling on a selfie.*



(h) Prompt: *Aelita from Code Lyoko in the forest, gathering flowers.*

Figure 8: More results of qualitative comparison of different methods for training Stable Diffusion 1.5 on Pick-a-Pic v2 dataset.

E Evaluation on K-Sort Eval Benchmark

To ensure a fair comparison, we follow prior work by adopting the same evaluation benchmark and standard metrics. However, as discussed earlier, these traditional metrics are often insufficiently discriminative, making it difficult to clearly distinguish the performance differences among methods.

To provide a more informative assessment, we additionally evaluate all methods on K-Sort Eval Li et al., a recently proposed preference-alignment benchmark. K-Sort Eval is an automated evaluation protocol derived from K-Sort Arena and has been shown to correlate well with human judgments.

The evaluation results are summarized in Table 7. Compared with fixed-margin baselines, ArenaPO consistently achieves the best performance, demonstrating the effectiveness of adaptive preference optimization.

Table 7: Performance comparison on the K-Sort Eval benchmark.

Method	K-Sort Eval Score
Stable Diffusion 1.5	20.03
Margin = 0.5	20.72
Margin = 2.0	21.16
Margin = 4.0	20.95
ArenaPO (ours)	**21.63**

F Comparison with RLHF Using an Explicit Reward Model

A natural question is whether the performance gains of ArenaPO can also be achieved by introducing an explicit reward model, similar to RLHF-style optimization. Following this suggestion, we augment Diffusion-DPO with ImageReward as an explicit reward model and compare it against ArenaPO.

The results are presented in Table 8. Incorporating ImageReward substantially improves the IR metric itself, indicating that the model tends to optimize specifically toward the reward model objective. However, improvements on other evaluation metrics remain limited and generally do not surpass ArenaPO. In some cases, such as HPS, the performance even degrades compared with ArenaPO. Similar observations have also been reported in the Diffusion-DPO paper Wallace et al. (2024), which notes that *"we did not observe stable improvement when training from public implementations on Pick-a-Pic."*

Table 8: Comparison with RLHF-style optimization using ImageReward as an explicit reward model. Best results are shown in bold.

Dataset	Method	Stable Diffusion 1.5				
		PickScore	HPS	Aes.	CLIP	IR
Pick v2	Pre-Trained	0.2088	0.2697	5.4933	0.3480	-0.0469
	IR as Reward Model	0.2162	0.2682	5.5728	0.3556	**0.7621**
	ArenaPO (ours)	**0.2184**	**0.2842**	**5.8591**	**0.3619**	0.6881

Moreover, introducing an explicit reward model incurs additional training complexity and computational overhead. Overall, these results suggest that ArenaPO achieves a better trade-off between alignment quality and training efficiency without relying on an external reward model.