PADRE: Pseudo-Likelihood based Alignment of Diffusion Language Models

Anonymous Authors[1]

Abstract

Policy-gradient reinforcement learning (PGRL) is widely used to improve language model reasoning. However these methods do not work well with diffusion based language models (dLLMs). Most attempts to apply PGRL to dLLMs, either are extremely unscalable or use unprincipled approximations. Our proposed framework (PADRE) uses a novel objective pseudo-likelihood based objective for alignment of dLLMs. Our objective has the same optima as PGRL based optimization, but does not need to evaluate likelihood from dLLMs. Experiments on mathematical reasoning benchmarks show that PADRE matches or surpasses the performance of GRPO and related baselines. Our approach provides a stable and practical alternative for RL-based fine-tuning of reasoning-focused dLLMs.

1. Introduction

Large Language Models (LLMs) have become the backbone of modern natural language processing (NLP), powering applications including code generation (Gehring et al., 2024), robotic control (Wang et al., 2024b), and autonomous agents (Deng et al., 2023). Much of their success stems from extensive pretraining on massive text corpora, which equips them with broad linguistic knowledge and fluency (Ouyang et al., 2022). Yet, while this pretraining enables impressive surface-level capabilities, many high-value downstream tasks such as mathematical problem-solving require more xomplex reasoning (Webb et al., 2023; Wei et al., 2022). Reasoning demands the ability to perform structured, multi-step thinking and to generalize beyond the patterns seen in training data (Xu et al., 2025). Reinforcement learning (RL), particularly when driven by outcome-based rewards, has shown promise in enhancing LLMs' reasoning abilities (Luong et al., 2024). This is especially evident in domains like mathematics, where problems have objective, verifiable solutions. The ability to automatically assess correctness in such tasks provides a powerful training signal, enabling RL methods to fine-tune LLMs in a targeted, scalable manner (de Winter et al., 2024).

To date, policy-gradient (Williams and Peng, 1990) (PG) based RL methods (specifically Proximal Policy Optimization (PPO) (Schulman et al., 2017b) and Generalized Return Policy Optimization (GRPO) (Guo et al., 2025)) have been the dominant approaches for post-training LLMs on reasoning-heavy tasks like mathematical problem solving (Xu et al., 2025). These methods focus on optimizing a policy (the LLM's output distribution) to maximize task-specific rewards, such as correctly solving a given problem. The domain of mathematical reasoning is particularly well-suited to this framework: reward attribution is straightforward because each solution attempt can be automatically judged as correct or incorrect. Consequently, PGRL has emerged as the method of choice for enhancing LLM reasoning in many domains (Shao et al., 2024).

While most LLMS, use an autoregressive model, diffusion based models have recently emerged as an equally powerful way to train language models (Nie et al., 2025; Shi et al., 2024). Due to their ability to parallelly sample multiple tokens, these diffusion LLMs (dLLMs) can significantly outperform autoregressive (AR) models, especially when generating long sequences. Unfortunately, the policy-gradient methods which underpin the success of standard LLMs, cannot directly be applied to dLLMs. This is because these PGRL methods rely on the ability to compute likelihood of generations; something which is difficult for dLLMs.

Interestingly, the standard KL-regularized RL perspective of LLMs training naturally connects to probabilistic inference (Jaynes, 1979; Khalifa et al., 2020; Ziebart et al., 2008)). Building on this insight, we propose a novel objective for alignment of dLLMs that does not rely on monte-carlo estimation of probabilities. Our approach dubbed PADRE , is inspired by pseudo-likelihood (Besag, 1975; 2001) and achieves the same optimality conditions as RLHF and KL-regularized RL; but crucially avoids the inefficiencies of PGRL with dLLMs. This enables scalable, stable, and practical alignment for reasoning-intensive tasks in dLLMs.

2. Preliminaries

Proximal Policy Optimization (PPO) PPO (Schulman et al., 2017b) is a classic on-policy algorithm for policy gradient based optimization. While the vanilla Reinforce (Williams, 1992) and other similar gradient methods yield an unbiased gradient, taking large steps often leads to instability in training. Improving upon the TRPO (Schulman et al., 2017a) method, PPO uses a clipped surrogate objective to prevent large charge updates. Each iteration of PPO can be written as optimizing:

$$J(\pi) = \mathbb{E}_{D_k}\Bigg[\min\Bigg(\frac{\pi(a_h \mid s_h)}{\pi_{\theta_k}(a_h \mid s_h)} \cdot A_h(s_h, a_h), \\ \text{clip}\left(\frac{\pi(a_h \mid s_h)}{\pi_{\theta_k}(a_h \mid s_h)}, \varepsilon\right) \cdot A_h(s_h, a_h)\Bigg)\Bigg]$$

$\widehat{A}_h$ is an advantage function, ε is a hyperparameter, π_{θ_k} is the previous policy (parameterized via θ), clip is a function which clamps its input in the range $[1-\varepsilon, 1+\varepsilon]$, and D_k is a set of trajectories obtained by executing π_{θ_k} on the MDP.

KL Regularization Commonly in RL the objective is to find a policy π that maximizes the expected cumulative reward $J(\pi) = \mathbb{E}_{\tau\sim\pi}[r(\tau)]$, where $r(\tau)$. However direct maximization is often undesired (especially in the context of LLMs), as the resulting models converge to narrow, high-reward outputs with low diversity (Choshen et al., 2019; Paulus et al., 2017). To alleviate this models are often trained with a regularization term (given by the KL divergence to a reference model). This approach inherently prevents distribution collapse. By maintaining diversity through KL regularization, the model retains its generative capabilities while learning to favor high-reward behaviors (Ziebart et al., 2008; Ziebart, 2010; Neu et al., 2017; Ouyang et al., 2022). The standard KL regularized return is defined as

$$J_\beta(\pi) = J(\pi) - \beta \cdot \mathbb{E}_{\tau\sim\pi}\left[\log \frac{\pi(\tau)}{\pi_{ref}(\tau)}\right]$$

where $\beta > 0$ is a regularization parameter that controls the strength of the penalty $D_{\text{KL}}(\pi\|\pi_{ref}) = \mathbb{E}_{\tau\sim\pi}\left[\log \frac{\pi(\tau)}{\pi_{ref}(\tau)}\right]$ which is the Kullback-Leibler divergence from π to π_{ref}. This is effectively equivalent to adding the log propensity term to the rewards r.

Generalized Reward-Penalized Optimization (GRPO) GRPO (Shao et al., 2024) is a PPO based method for fine-tuning LLMs. GRPO usually samples multiple responses y_i for each prompt x, uses a verifier (for math-like problems) or other reward functions to rate these samples, and computes advantages by normalizing rewards within each prompt group. The advantage for the i-th response y^i is computed as:

$$\widehat{A}^i = \frac{r(x, o^i) - \text{mean}(r(x, o^1), \ldots, r(x, o^G))}{\text{stdev}(r(x, o^1), \ldots, r(x, o^G)),} \tag{1}$$

where $r(x, o^i)$ is the outcome for response o^i to prompt x as we defined above.

This response-level advantage $\widehat{A}^i$ is in the PPO objective $\mathcal{L}^{\text{PPO}}$, along with KL-regularization to compute the update

$$J^{\text{GRPO}}(\pi) = \sum_{i=1}^{G}\frac{1}{|y_i|}\sum_{t=1}^{|y_i|}\min\left[\frac{\pi(a_t^i \mid s_t^i)}{\pi_{\theta_k}(a_t^i \mid s_t^i)}\widehat{A}^i, \text{clip}\left(\frac{\pi(a_t^i \mid s_t^i)}{\pi_{\theta_{\theta_k}}(a_t^i \mid s_t^i)}, \epsilon\right)\widehat{A}^i\right] \\ - \beta D_{\text{KL}}(\pi\|\pi_{ref}),$$

where a_t^i is the t^{th} token in the sequence y^i, and $s_t^i = (y_{<t}^i, x)$ is the concatenation of all processed tokens. Effectively instead of a per step/action reward as in PPO; by using the entire trajectory reward in the objective as given, GRPO implicitly assigns each token in the response the corresponding reward. The standardization of the reward replaces the value function estimate in standard PPO. However its overall effect is similar, to stabilize training by reducing variance. Thus GRPO is often simpler to implement than PPO for post-training LLMs (Wang et al., 2024a).

2.1. Masked Diffusion

Masked Diffusion Language Models (MDLMs) are a class of discrete diffusion models that generate text by gradually denoising a sequence of tokens, starting from a fully masked state. Unlike autoregressive (AR) models that generate tokens sequentially, or standard BERT-style masked language models that perform single-step infilling, MDLMs iteratively refine predictions over multiple steps, allowing for more flexible and globally coherent generation.

Forward and Reverse Process The forward process in MDLMs is a discrete noising process that gradually corrupts an input sequence x_0 (where x_0 is a sequence of one-hot token vectors) by replacing tokens with a special `[MASK]` token. Let x_t denote the sequence at timestep $t \in [0, 1]$, where $t = 0$ corresponds to the clean input and $t = 1$ corresponds to the fully masked state. The corruption is governed by a noise schedule α_t, which is strictly decreasing in t.

For each token $x_t^{(i)}$ in the sequence at time t, the forward process is defined as:

$$q(x_t^{(i)}|x_0^{(i)}) = \begin{cases}\alpha_t, & \text{if } x_t^{(i)} = x_0^{(i)} \quad \text{(token remains unchanged)},\\ 1-\alpha_t, & \text{if } x_t^{(i)} = \texttt{[MASK]} \quad \text{(token is masked)}.\end{cases}$$

This can also be written as a categorical distribution:

$$q(x_t|x_0) = \text{Cat}\left(x_t; \alpha_t x_0 + (1-\alpha_t)\,\texttt{[MASK]}\right).$$

Here, α_t controls the probability of a token being preserved . Common schedule choices include: linear: $\alpha_t = 1-t$ and cosine: $\alpha_t = \cos(\frac{\pi}{2}t)$schedules. LLada (Nie et al., 2025) propose using the linear schedule.

The reverse process learns to denoise x_t back to x_0. Unlike the forward process, which is fixed, the reverse process is parameterized by a neural network f_θ that predicts the original tokens given a masked sequence. The reverse transition $q(x_s|x_t)$ for $s < t$ is derived from Bayes' rule and has three cases:

(1) **If $x_t^{(i)} \neq$ `[MASK]`:** The token is already unmasked and remains unchanged:

$$q(x_s^{(i)}|x_t^{(i)}) = \delta(x_s^{(i)} = x_t^{(i)}).$$

(2) **If $x_t^{(i)} =$ `[MASK]` and $x_s^{(i)} =$ `[MASK]`:** The token stays masked:

$$q(x_s^{(i)}|x_t^{(i)}) = \frac{1-\alpha_s}{1-\alpha_t}.$$

(3) **If $x_t^{(i)} =$ `[MASK]` and $x_s^{(i)} \neq$ `[MASK]`:** The token is unmasked, and the model predicts the original token:

$$q(x_s^{(i)}|x_t^{(i)}) = \frac{\alpha_s - \alpha_t}{1-\alpha_t} \cdot f_\theta(x_0^{(i)}|x_t).$$

The model f_θ is trained to predict x_0 given x_t, similar to BERT but conditioned on the masking level t.

Training Objective The model is trained to minimize the Negative Evidence Lower Bound (NELBO), which simplifies to a weighted negative log-likelihood (NLL) over masked tokens. The loss is:

$$\mathbb{E}_{t,x_0,x_t}\left[\frac{\alpha_t'}{1-\alpha_t}\sum_{i=1}^{L}\mathbb{1}[x_t^{(i)} = \texttt{[MASK]}]\cdot\log f_\theta(x_0^{(i)}|x_t)\right],$$

where: $\alpha_t' = \frac{d\alpha_t}{dt}$ is the derivative of the noise schedule) and The indicator function $\mathbb{1}[x_t^{(i)} = \texttt{[MASK]}]$ ensures only masked tokens contribute to the loss.

For a linear schedule $\alpha_t = 1 - t$, such as that used in Nie et al. (2025) this reduces to:

$$\mathcal{L}(\theta) = \mathbb{E}_{t,x_0,x_t}\left[\frac{1}{t}\sum_{i=1}^{L}\mathbb{1}[x_t^{(i)} = \texttt{[MASK]}]\cdot\log f_\theta(x_0^{(i)}|x_t)\right].$$

3. Related Work

Reinforcement learning (RL) with Kullback-Leibler (KL) regularization KL regularized learning has its roots in maximum-entropy RL Ziebart et al. (2008); Neu et al. (2017), where a KL penalty ensures that learned policies remain close to a reference distribution. This framework has led to several influential algorithms, including Soft Q-Learning (SQL) (Haarnoja et al., 2017) and Soft Actor-Critic (SAC) (Haarnoja et al., 2018), as well as more recent approaches such as those of Ji et al. (2024) and Wang et al. (2024a). Additionally, direct alignment algorithms like DPO (Rafailov et al., 2023), IPO (Azar et al., 2024), KTO (Ethayarajh et al., 2024) have gained prominence for their simplicity and effectiveness.

The relation between entropy-regularized control and divergence-minimisation is known since the seminal work of Jaynes (1979). Rafailov et al. (2023) used the form of the optimal policy of such a procedure (Ziebart et al., 2008) to propose DPO. Other work in RL have noted the relation between bayesian inference and optimal control (Levine, 2018). Building on the same,Khalifa et al. (2020); Korbak et al. (2022) proposed an alternative for fine-tuning language models, achieving results comparable to KL-regularized RL. Their GDC method minimizes $KL(q||p)$ rather than $KL(p||q)$, making it theoretically distinct from standard RL objectives (Korbak et al., 2022).

Policy Gradient Methods Policy gradient methods (Williams and Peng, 1990) have been foundational in modern RL. Recent advancements for language model training (Ouyang et al., 2022; Shao et al., 2024) have been based on PPO (Schulman et al., 2017b) and its variants (Wu et al., 2023). However these methods rely on being able to compute the log-probabilities of the generated samples. Furthermore, when used in off-policy manner one further needs techniques like importance sampling to ensure stable training (Wu et al., 2023). However, in the context of alignment of fine-tuning diffusion language models, the densities required for computing these are not available efficiently. Furthermore methods which use approximations of the density (Zhao et al., 2025) , are using biased gradients and hence not optimizing the expected reward. PADRE on the other hand uses on-policy data to compute unbiased estimates of a principled objective.

Reinforcement Learning for LLM Reasoning Since the work of Ouyang et al. (2022), application of RL to large language models (LLMs) has seen significant progress, especially in MDP-based formulations of reasoning, as seen in OpenAI's O1 and DeepSeek's R1. While policy-based methods such as GRPO (Guo et al., 2025), and their variants (e.g., DAPO (Yu et al., 2025), Dr. GRPO (Liu et al., 2025)) dominate this space, other approaches like ReMax (Li et al., 2023) and RAFT (Dong et al., 2023) have also been explored. Building on the idea of Bellman Residuals (Schweitzer and Seidmann, 1985; Baird, 1995), recently value based methods (Jia et al., 2025) have also been pro-

posed for reasoning.

Language Diffusion Models While diffusion models have revolutionized continuous data generation in the visual domain (Song et al., 2020; Ho et al., 2020), their adaptation to discrete textual data presents unique challenges. The fundamental tension arises from the categorical nature of language tokens, which necessitates specialized approaches beyond the standard diffusion framework. The masked diffusion paradigm has emerged as a particularly successful instantiation of discrete diffusion for language (Sahoo et al., 2024; Shi et al., 2024). This approach, a special case of discrete diffusion (Austin et al., 2021), has recently achieved significant scaling milestones. Recently, LLaDA-8B (Nie et al., 2025), has been shown to match or surpasses LLaMA-3 8B on MMLU, ARC-C and few-shot reasoning.

There have been a few methods proposed for alignment of dLLMs. The naive method to adopt PGRL does not work well due to intractable log-likelihoods. Nie et al. (2025) estimate the log-likelihood $\log p_\theta(y|x)$, by randomly masking different portions of the output and then performs a single denoising step to approximate likelihood for each token position t. Instead of MC estimates, Zhao et al. (2025) propose a mean-field approximation for the likelihood of an output. However both these methods, effectively end up using biased gradients, and hence even under idealized conditions are not guaranteed to optimize the reward.

4. Method

Modern diffusion-based language models, such as LLaDA (Nie et al., 2025), generate sequences by iteratively denoising masked tokens toward complete text. Let x denote the prompt and $y = (y^1, \ldots, y^T)$ the completed generation. Unlike autoregressive (AR) models, which decompose log-likelihoods as $\log p_\theta(y|x) = \sum_{t=1}^{T} \log p_\theta(y^t|y^{<t}, x)$, diffusion models produce outputs in a non-sequential and non-factorized manner. As such the straightforward method of using PGRL for dLLMs, does not scale. Additionally, the estimation procedure used for computing the probability can be biased, which then leads to unstable behavior. To alleviate this we propose a new probabilistic objective for alignment of dLLMs.

4.1. RL alignment as Probabilistic Inference

We begin with a key result regarding the KL-constrained RL methods, which will then motivate a different probabilistic objective that can be used with dLLMs.

Consider the unnormalized target distribution $\tilde{q}(\tau)$ given as:

$$\tilde{q}(\tau) = p_{\text{ref}}(\tau) \exp(r(\tau)/\beta), \tag{2}$$

which leads to the Boltzmann distribution (Jaynes, 1979):

$$q(\tau) = \frac{1}{Z}\pi_{\text{ref}}(\tau) \exp(r(\tau)/\beta), \quad \text{where } Z = \sum_\tau p_{\text{ref}}(\tau) \exp(\frac{r(\tau)}{\beta}). \tag{3}$$

Expanding the KL term in J_β, we can rewrite the standard policy gradient obective J as:

$$J(\theta) = \mathbb{E}_{\tau\sim p_\theta}[r(\tau)] - \beta \mathbb{E}_{\tau\sim p_\theta}[\log p_\theta(\tau) - \log p_{\text{ref}}(\tau)] \tag{4}$$

$$= -\beta \mathbb{E}_{\tau\sim p_\theta}[\log p_\theta(\tau) - \log p_{\text{ref}}(\tau) - r(\tau)/\beta] \tag{5}$$

$$= -\beta(D_{\text{KL}}(p_\theta \| q) + \log Z). \tag{6}$$

Hence, maximizing J is equivalent to minimizing $D_{\text{KL}}(p_\theta \| q)$.

This interpretation while known from earlier ideas in inference and entropy-regularized reinforcement learning (Ziebart et al., 2008; Jaynes, 1979), provides new pathways for fine-tuning LLMs. Specifically, instead of committing exclusively to optimizing Kullback-Leibler (KL) divergence (which is implicitly what these methods are doing), one may adopt the broader perspective from probabilistic inference on distribution matching. Specifically, the goal is to align a learned policy distribution p_θ) with an unnormalized target $q \propto p_{\text{ref}}(\tau)e^{r(\tau)/\beta}$, where p_{ref} is reference policy and r the reward function.

More generally, however one can consider optimizing other divergences $D(p_\theta||q)$, not necessarily KL. Under ideal conditions (e.g., unlimited model expressivity and global optimization), the final learned policy is invariant to the choice of divergence. However, in practice, different objectives exhibit varying empirical behaviors. A natural alternative is the reverse KL, $D_{KL}(q||p)$, however this is difficult as it requires sampling from the EBM q. Additionally, when the divergence does not go down to 0, rKL can lead to mode covering models that can frequently generate incorrect trajectories.

For our applications, an ideal objective should satisfy three key criteria: a) avoid requiring the partition function Z of q (efficiency) and b) can be computed without access to full densities from p_θ (dLLM) and c) theoretical guarantees of convergence to the target distribution $\tilde{q}$. In the next section, we discuss an objective baed on pseudo-likelihood matching, a candidate objective with such properties. We call our proposed method PADRE , short for Pseudo-likelihood Alignment Diffusion-based REasoning

4.2. Pseudo-likelihood Alignment

The pseudo-likelihood objective (Besag, 1975) provides a statistically consistent approach to learning joint distributions through their local behaviour. The key idea is to approximate the joint distribution over sequences using their

conditional marginals. For a sequence $y = (y^1, \ldots, y^T)$, define the pseudo-likelihood of p_θ as the product of its conditional marginals:

$$\mathrm{PL}(y) = \prod_{i=1}^{|y|} p_\theta(y^(i)|y^{-i}), \tag{7}$$

where y^{-i} denotes the sequence with the i-th token removed. Besag (1975) established that this product forms a proper statistical score function when the joint distribution of y is positive everywhere. Building on this insight, rather than minimizing the KL divergence between the full distributions $p_\theta(y|x)$ and $\tilde{q}(y|x)$, we propose minimizing the sum of KL divergences over conditional marginals:

$$\sum_y \sum_{i=1} \mathrm{KL}(p_\theta(y^t|y^{-i}) \| \tilde{q}(y^t|y^{-t})). \tag{8}$$

where we have supressed the dependence on the prompt x for notational convenience It is easy to see that the above objective is 0, iff all the conditional distributions match; which by the pseudo-likelihood argument implies that p_θ and $\tilde{q}$ match. However, directly evaluating this loss is intractable due to the combinatorial size of the space of all sequences y.

To address this computational bottleneck, we introduce a weighted variant of the pseudo-likelihood objective. Rather than treating all trajectories equally, we weight them according to their probability under the model distribution $p_\theta(y^{-i})$.

$$J = \sum_y \sum_i p_\theta(y^{-t}) \mathrm{KL}(p_\theta(y^i|y^{-i}) \| \tilde{q}(y^t|y^{-i})) \tag{9}$$

$$= \sum_y \sum_i p_\theta(y_{-t}) \mathbb{E} y^i \sim p_\theta(\cdot|y^{-i}) \left[\log \frac{p_\theta(y^i|y^{-i})}{\tilde{q}(y^i|y^{-i})} \right] \tag{10}$$

$$= \sum_y \sum_i p_\theta(y) \left[\log \frac{p_\theta(y^i|y^{-i})}{\tilde{q}(y^i|y^{-i})} \right] \tag{11}$$

By multiplying each term by $p_\theta(y_{-i})$ and applying the law of total expectation, the weighted KL objective turns into our final tractable objective:

$$\mathcal{L}(\theta) = \mathbb{E} y \sim p_\theta \sum_{i=1}^{N} \left[\log p_\theta(y^i|y^{-i}) - \log \tilde{q}(y^i|y^{-i}) \right] \tag{12}$$

This formulation offers two key advantages. First, it eliminates the need for explicit sampling from the conditional distributions $p_\theta(\cdot|y^{-i})$. Second, it maintains the theoretical property that the global optimum is achieved precisely when all conditional distributions match their target counterparts, provided p_θ has full support.

We further note that the conditional marginal distribution of q inherits structure from the reference model:

$$\tilde{q}(y^i|y^{-i}) \propto p_{\text{ref}}(y^i|y^{-i}) \exp(r(y^i, y^{-i})) \tag{13}$$

This enables efficient computation when p_{ref}'s conditionals are tractable. The log-conditional of $\tilde{q}$ simply becomes:

$$\log \tilde{q}(y^i|y^{-i}) = \log p_{\text{ref}}(y^i|y^{-i}) + r(y^i, y^{-i}) - \log Z(y^{-i}) \tag{14}$$

where $Z(y^{-i})$ is the conditional partition function. Importantly, the $Z(y^{-i})$ term can be dropped from the KL divergence objective, as it contributes an additive constant that does not affect the optimization landscape.

4.2.1. Evaluating Conditional Likelihood

To evaluate the aforementioned objective efficiently we still need to be able to compute the conditional densities. For LLaDA like approaches, which learn the masked diffusion model, one can efficiently compute conditional token probabilities for any word $p_\theta(w_| y_{0,-i})$ in a single forward pass. This result stems from the unique factorization properties of the reverse diffusion kernel in these models.

Consider a clean sequence $y_0 = [y^1, \ldots, y^L]$ and its corrupted version y_t at timestep t, where `[MASK]` denotes the mask token. The key insight lies in the structure of the reverse process kernel $p_\theta(y_{t-1} \mid x_t)$, which factorizes over masked positions M_t:

$$p_\theta(y_{t-1} \mid y_t) = \prod_{k \in M_t} f_\theta(y_k \mid y_t) \prod_{k \notin M_t} \delta(y_t^k) \tag{15}$$

where f_θ is the denoising network and δ maintains unmasked tokens. This factorization emerges from the independent token corruption in the forward process.

To compute $p_\theta(y^i \mid y_0^{-i})$, we construct a partially masked sequence $y_1 = y_0^{-i} \cup \{\texttt{[MASK]}_i\}$ where only position i is masked. This represents a valid sample from the forward process with $M = \{i\}$.

Through marginalization over latent variables $y_{1:T}$, we have:

$$p_\theta(w_i \mid y_{0,-i}) = \sum_{y_{1:T}} p_\theta(w_i \mid y_1) p_\theta(y_{1:T} \mid y_0^{-i}) \tag{16}$$

The Markov property of the diffusion process ensures w depends only on y_1. Crucially, y_1 is deterministic given y_0^{-i}, causing the summation to collapse to:

$$p_\theta(w_i \mid y_0^{-i}) = p_\theta(w_i \mid y_1) \tag{17}$$

Examining the reverse kernel reveals:

$$p_\theta(y_0 \mid y_1) = f_\theta(w^i \mid y_1) \prod_{j \neq i} \delta(y_1^j) \tag{18}$$

Thus, the conditional probability equals the denoiser's output:

$$p_\theta(w_i \mid y_0^{-i}) = f_\theta(w_i \mid y_1) \tag{19}$$

Thus masked diffusion LMs can compute exact token-level conditionals through a single denoiser evaluation. Since the reference model as well as the model being learnt are masked diffusion LMs, we can use this trick to compute the probabilities needed for our objective. This efficient computation cannot be done on autoregressive models, and is unique to the masked diffusion LMs.

5. Experiments

We use the recent SoTA dLLM LLaDA-8B-Instruct (Nie et al., 2025) as the baseline model. Following Zhao et al. (2025) we compare results using supervised fine-tuning (SFT), the dffu-grpo version of GRPO proposed by Zhao et al. (2025), and the combination of the two (a method referred to as d1). Experiments are conducted on three math reasoning benchmarks: a) GSM8K (Cobbe et al., 2021), a dataset of multi-step grade school math problems, b) MATH (Hendrycks et al., 2021) a set of high-level math problems and c) MATH500(Lightman et al., 2023), a curated subset of MATH. We also report results from the dream model of Ye et al. (2025) for another baseline comparison.

Methodology For fair comparison, we follow the experimental procedure of Zhao et al. (2025). We first do SFT training of the model on the s1k dataset for 20 epochs with a sequence length of 4096 tokens. For rewards, we use the composite reward function combining formatting and correctness rewards as recommended by Zhao et al. (2025).

We test our model under 0-shot prompting with the prompts as reported in Zhao et al. (2025). The results (reported in Table 1demonstrate that PADRE outperforms all baselines across benchmarks. On GSM8K, it achieves 85.6% accuracy, significantly surpassing competing methods. On the MATH500, PADRE reaches 40.9%, outperforming d1 by +0.7% . Additionally we see that PADRE matches DREAM (Ye et al., 2025) on MATH.

Table 1. Performance of Diffusion Language Models on Math Benchmarks. We can see that PADRE matches our outperforms other methods. The results denoted by - are due to the corresponding paper not having reported the results.

Method	**GSM8K**	**MATH500**	**MATH**
Dream 7B	81.1	-	42.9
LLaDA 8B	78.3	36.2	38.9
+ SFT*	81.1	34.8	-
+ diffu-GRPO*	81.9	39.2	-
+ d1-LLaDA	82.1	40.2	-
PADRE	85.6	40.9	43.0

6. Conclusion

We have introduced PADRE , a novel approach to tune dLLMs for reasoning . Our method is a version of the pseudo-likelihood training (Besag, 1975). The probabilistic objective only relies on sampling and estimating conditional distributions, both of which are efficient with dLLMs. Additionally, unlike other methods, our approach learns the same optima as standard PGRL methods. Experiments show that PADRE matches or outperforms other methods for finetuning dLLMs on challenging math based reasoning tasks.

References

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

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*, pages 4447–4455. PMLR, 2024.

Leemon Baird. Residual algorithms: Reinforcement learning with function approximation. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 30–37, 1995.

Julian Besag. Statistical analysis of non-lattice data. *Journal of the Royal Statistical Society Series D: The Statistician*, 24(3):179–195, 1975.

Julian Besag. [conditionally specified distributions: An introduction]: Comment. *Statistical Science*, 16(3):265–267, 2001.

Leshem Choshen, Lior Fox, Zohar Aizenbud, and Omri Abend. On the weaknesses of reinforcement learn-

ing for neural machine translation. *arXiv preprint arXiv:1907.01752*, 2019.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Joost CF de Winter, Dimitra Dodou, and Yke Bauke Eisma. System 2 thinking in openai's o1-preview model: Near-perfect performance on a mathematics exam. *Computers*, 13(11):278, 2024.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing Systems*, 36:28091–28114, 2023.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. *arXiv preprint arXiv:2304.06767*, 2023.

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

Jonas Gehring, Kunhao Zheng, Jade Copet, Vegard Mella, Quentin Carbonneaux, Taco Cohen, and Gabriel Synnaeve. Rlef: Grounding code llms in execution feedback with reinforcement learning. *arXiv preprint arXiv:2410.02089*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*, pages 1352–1361. PMLR, 2017.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning*, pages 1861–1870. PMLR, 2018.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

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

Edwin T Jaynes. Concentration of distributions at entropy maxima. *ET Jaynes: Papers on probability, statistics and statistical physics*, page 315, 1979.

Xiang Ji, Sanjeev Kulkarni, Mengdi Wang, and Tengyang Xie. Self-play with adversarial critic: Provable and scalable offline alignment for language models. *arXiv preprint arXiv:2406.04274*, 2024.

Zeyu Jia, Alexander Rakhlin, and Tengyang Xie. Do we need to verify step by step? rethinking process supervision from a theoretical perspective. *arXiv preprint arXiv:2502.10581*, 2025.

Muhammad Khalifa, Hady Elsahar, and Marc Dymetman. A distributional approach to controlled text generation. *arXiv preprint arXiv:2012.11635*, 2020.

Tomasz Korbak, Hady Elsahar, Germán Kruszewski, and Marc Dymetman. On reinforcement learning and distribution matching for fine-tuning language models with no catastrophic forgetting. *Advances in Neural Information Processing Systems*, 35:16203–16220, 2022.

Sergey Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*, 2018.

Ziniu Li, Tian Xu, Yushun Zhang, Zhihang Lin, Yang Yu, Ruoyu Sun, and Zhi-Quan Luo. Remax: A simple, effective, and efficient reinforcement learning method for aligning large language models. *arXiv preprint arXiv:2310.10505*, 2023.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

Trung Quoc Luong, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. Reft: Reasoning with reinforced fine-tuning. *arXiv preprint arXiv:2401.08967*, 3, 2024.

Gergely Neu, Anders Jonsson, and Vicenç Gómez. A unified view of entropy-regularized markov decision processes. *arXiv preprint arXiv:1705.07798*, 2017.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models, 2025. URL https://arxiv.org/abs/2502.09992.

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. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*, 2017.

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.

Subham Sekhar Sahoo, Marianne Arriola, Aaron Gokaslan, Edgar Mariano Marroquin, Alexander M Rush, Yair Schiff, Justin T Chiu, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=L4uaAR4ArM.

John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. Trust region policy optimization, 2017a. URL https://arxiv.org/abs/1502.05477.

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

Paul J. Schweitzer and Abraham Seidmann. Generalized polynomial approximations in markovian decision processes. *Journal of Mathematical Analysis and Applications*, 110(2):568–582, 1985.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. *Advances in neural information processing systems*, 37:103131–103167, 2024.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2020.

Haoxiang Wang, Yong Lin, Wei Xiong, Rui Yang, Shizhe Diao, Shuang Qiu, Han Zhao, and Tong Zhang. Arithmetic control of llms for diverse user preferences: Directional preference alignment with multi-objective rewards, 2024a. URL https://arxiv.org/abs/2402.18571.

Zihan Wang, Brian Liang, Varad Dhat, Zander Brumbaugh, Nick Walker, Ranjay Krishna, and Maya Cakmak. I can tell what i am doing: Toward real-world natural language grounding of robot experiences. *arXiv preprint arXiv:2411.12960*, 2024b.

Taylor Webb, Keith J Holyoak, and Hongjing Lu. Emergent analogical reasoning in large language models. *Nature Human Behaviour*, 7(9):1526–1541, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256, 1992.

Ronald J Williams and Jing Peng. An efficient gradient-based algorithm for on-line training of recurrent network trajectories. *Neural computation*, 2(4):490–501, 1990.

Tianhao Wu, Banghua Zhu, Ruoyu Zhang, Zhaojin Wen, Kannan Ramchandran, and Jiantao Jiao. Pairwise proximal policy optimization: Harnessing relative feedback for llm alignment. *arXiv preprint arXiv:2310.00212*, 2023.

Fengli Xu, Qianyue Hao, Zefang Zong, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, Tianjian Ouyang, Fanjin Meng, Chenyang Shao, Yuwei Yan, Qinglong Yang, Yiwen Song, Sijian Ren, Xinyuan Hu, Yu Li, Jie Feng, Chen Gao, and Yong Li. Towards large reasoning models: A survey of reinforced reasoning with large language models, 2025.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b, 2025. URL https://hkunlp.github.io/blog/2025/dream.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Siyan Zhao, Devaansh Gupta, Qinqing Zheng, and Aditya Grover. d1: Scaling reasoning in diffusion large language models via reinforcement learning. *arXiv preprint arXiv:2504.12216*, 2025.

Brian D. Ziebart. *Modeling Purposeful Adaptive Behavior with the Principle of Maximum Causal Entropy*. PhD thesis, Carnegie Mellon University, 2010.

Brian D. Ziebart, Andrew L. Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI Conference on Artificial Intelligence*, 2008.