Extended Abstract

Motivation Discrete diffusion models have emerged as a promising alternative to autoregressive language models, offering potential advantages in controllability, robustness, and efficiency. However, these models typically rely on fixed or heuristic masking strategies during inference, which may be suboptimal for complex reasoning tasks. The performance of discrete diffusion models is highly sensitive to the masking strategies employed, presenting an opportunity to use reinforcement learning (RL) to learn more effective denoising schedules that can improve both efficiency and accuracy.

Method We use an RL approach to optimize token-level masking policies for discrete diffusion models. Our method learns a masking policy network that adaptively selects which tokens to re-mask at each denoising step, moving beyond fixed schedules or uniform random masking. We formulate this as a Markov Decision Process (MDP), where the state represents the denoiser's logits, actions are binary masking decisions per token, and rewards are based on downstream task performance. We employ GRPO (Group Relative Policy Optimization) to train the masking policy, with behavior cloning providing warm-start initialization.

Implementation We implement our approach using the LLaDA discrete diffusion model as the base environment. The masking policy is parameterized as a Transformer network that observes denoiser logits and outputs per-token binary distributions for keep/re-mask decisions. We train on mathematical reasoning tasks using the tinyGSM8K dataset, with binary rewards based on answer correctness. Our implementation includes both linear denoising schedules and adaptive threshold-based approaches that can vary the number of denoising steps.

Results Our learned masking policies achieve significant improvements over baseline heuristics. The BC + GRPO approach achieves 71.90% accuracy on tinyGSM8K, substantially outperforming random masking (49.24%), low-confidence heuristics (60.79%), and semi-autoregressive approaches (68.95%). Additionally, our nonlinear threshold-based approach maintains comparable performance (68.98%) while requiring 15.2% fewer diffusion steps on average, demonstrating improved computational efficiency.

Discussion The results demonstrate that learned masking strategies can significantly improve discrete diffusion model performance on reasoning tasks. The ability to maintain accuracy while reducing computational steps suggests that our approach captures more efficient denoising patterns than fixed heuristics. The success of RL in this domain opens new avenues for optimizing discrete diffusion models beyond traditional approaches, particularly for tasks requiring multi-step reasoning and planning.

Conclusion This work establishes that RL can effectively optimize discrete diffusion models through adaptive masking policies, achieving substantial improvements in both accuracy and efficiency over traditional heuristic approaches. Future areas to research include implementing a d1-style end-to-end RL approach and allowing the model to re-mask tokens that have already been denoised in previous steps.

Optimizing Re-Masking Schedules for Reasoning in Discrete Diffusion Models

Radostin Cholakov
Stanford University
radicho@stanford.edu

Zeyneb N. Kaya
Stanford University
zeynebnk@stanford.edu

Nicole Ma
Stanford University
manicole@stanford.edu

Abstract

Discrete diffusion models represent a promising paradigm for text generation, offering advantages over autoregressive models in controllability and non-sequential generation capabilities. However, existing approaches rely on fixed or heuristic masking strategies during inference, which may be suboptimal for complex reasoning tasks. We propose learning adaptive token-level masking policies through reinforcement learning to optimize the denoising process in discrete diffusion models. Our approach formulates masking decisions as a Markov Decision Process, where a learned policy network determines which tokens to re-mask at each denoising step based on the current denoiser state. We train this policy using Group Relative Policy Optimization (GRPO) with rewards based on downstream task performance. Experiments on mathematical reasoning tasks demonstrate that our learned masking policies significantly outperform traditional heuristics, achieving 71.90% accuracy on tinyGSM8K compared to 60.79% for low-confidence baselines. Furthermore, our adaptive approach can maintain performance while reducing the required number of diffusion steps by 15.2%, improving computational efficiency. These results suggest that reinforcement learning can effectively optimize discrete diffusion models for reasoning tasks, opening new directions for adaptive text generation.

1 Introduction

Autoregressive models (ARMs) have dominated the landscape of language modeling in recent years. These models generate text one token at a time, with each new token conditioned on all previously generated tokens. This approach has proven remarkably effective, leading to the development of powerful models like GPT-4 and LLaMA. However, recent advances in text diffusion LLMs have challenged this paradigm and shown potential for greater controllability, robustness, and efficiency (Nie et al., 2025; Zou et al., 2023; Li et al., 2022). Diffusion models, which have achieved remarkable success in image generation, represent a promising alternative paradigm. Unlike ARMs, diffusion models operate by gradually denoising data through an iterative process. While extensively studied in continuous domains like images, their application to discrete textual data presents unique challenges and opportunities. Unlike autoregressive approaches, diffusion-based methods can generate data in a non-sequential manner, potentially enhancing long-term planning capabilities, overcoming premise ordering limitations, offering better control over the generation process, and improving sampling efficiency. Furthermore, reinforcement learning (RL) has become a key paradigm to enabling reasoning in LLMs (Havrilla et al., 2024). RL provides a key method to robust learning in complex environments, especially with delayed learning signals.

Thus, we aim to harness the advantages of diffusion LLMs with RL towards reasoning. To do so, we use RL to learn and optimize the denoising strategy of diffusion LLMs. Our objective is to train a masking policy (MP) that determines the optimal positions to denoise at each diffusion step during inference, enabling planning and hierarchical problem-solving, and our main research question is

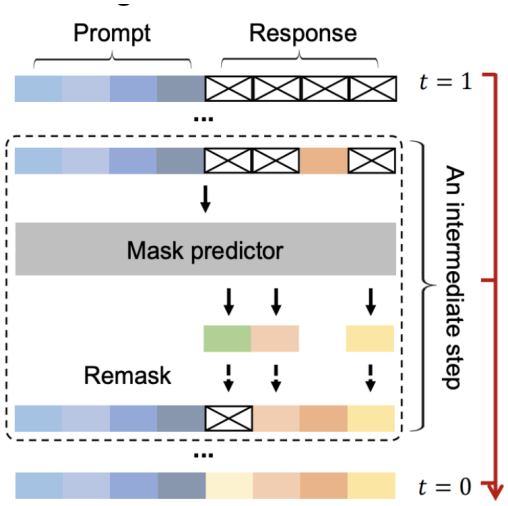

Figure 1: Mask diffusion method using LLaDA (Liu et al., 2025)

whether this optimized re-masking strategy can both improve the accuracy of text diffusion LLMs and reduce the number of diffusion steps needed to reach correct outputs.

2 Related Work

Diffusion for Language. While earlier works on text diffusion models investigated continuous diffusion by noising and denoising in the embedding space (Li et al., 2022), recent papers (e.g. LLaDA (Nie et al., 2025), Dream (Ye et al., 2025)) have presented effective large-scale *discrete* diffusion models for text by learning masked denoising schedules. With their scale and effectiveness, text diffusion models have thus emerged as a new yet powerful paradigm for LLMs.

Inference with discrete diffusion LLMs include an input which consists of the prompt and then a set length of [MASK] tokens which the LLM will denoise; at each diffusion step, the LLM generates predictions—a probability distribution over the vocab space—for *all* of the masked positions simultaneously. A number of token positions, typically following a linear denoising schedule, are denoised and are included when conditioning for the next diffusion step. The remaining positions are then remasked (see Fig. 1). These models typically use uniform or heuristic masking strategies; they primarily examine random, low-confidence-based, and semi-autoregressive (where sequential 'blocks' are denoised at a time) approaches. In contrast, Liu et al. (2025) present Discrete Diffusion with Planned Denoising (DDPD) (Liu et al., 2025), which separates training between a planner and a denoiser, demonstrating the impact of adaptive denoising sampling. They use a supervised cross-entropy loss based on the forward process; RL can enable optimization of the method to maximize performance and reasoning. The works have demonstrated that performance is highly sensitive to the masking strategies employed during inference, presenting potential for learning more optimal methods that can achieve both more efficient and effective generations.

RL for Text Diffusion. RL has very recently been applied to tune diffusion text generation policies with automated rewards. The d1 framework (Zhao et al., 2025) introduces a critic-free policy gradient algorithm tailored for masked diffusion models, and SEPO (Zekri and Boulle, 2025) introduces a theoretically justified policy gradient algorithm for fine-tuning discrete diffusion models using non-differentiable rewards. However, these methods still use traditional denoising frameworks with fixed number of generation steps and randomized remasking. We aim to explore the possibility to learn a remasking strategy with RL and thus also control for inference time compute scaling through refinement.

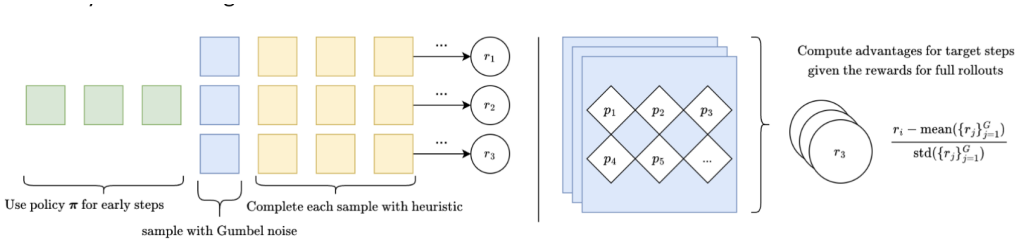

Figure 2: Methodology for RL rollout and GRPO advantage calculation.

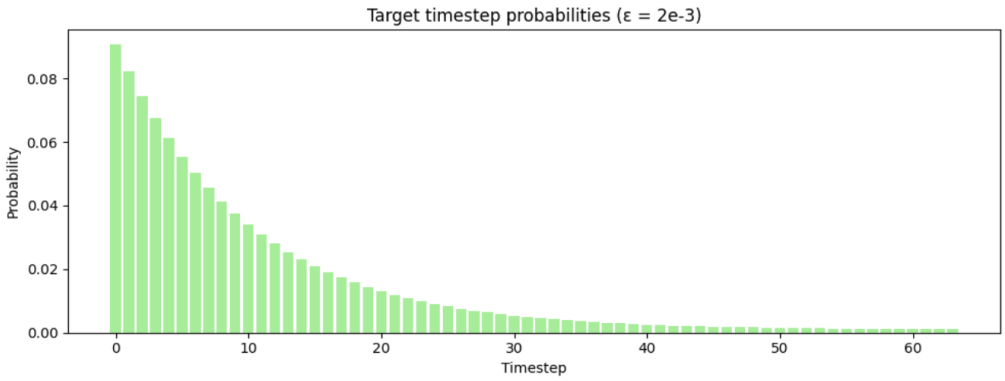

Figure 3: Distribution from which we sample the time step to train on for each sample

3 Method

We aim to enhance diffusion-based text generation by learning a token-level masking policy (MP) that adaptively selects which tokens to re-mask at each denoising step through RL. Unlike fixed schedules or uniform random masking, our MP network observes the current denoiser logits $\ell_t \in \mathbb{R}^{B\times L\times D}$ (before the LM head) and outputs a per-token binary distribution $\pi_\theta(a_t \mid s_t) \in [0,1]^{B\times L\times 2}$ indicating whether to keep or re-mask each position. We treat the pretrained deterministic diffusion model (e.g. greedy LLaDA at temperature 0) as the environment, collect trajectories of states and actions, and assign a final reward based on task success. We then apply GRPO to train the MP to maximize expected reward, yielding a more efficient and accurate denoising schedule for downstream tasks such as math problem solving.

MDP Formulation.

- **State** s_t: denoiser's logits at step t, $\ell_t \in \mathbb{R}^{B\times L\times D}$.
- **Action** a_t: binary mask sample per token indicating *re-mask* vs. *keep*, drawn from policy $\pi_\theta(a \mid s)$.
- **Transition** $s_{t+1} = f_{\text{denoise}}(s_t, a_t)$: deterministic update by the masked denoiser.
- **Reward** $r_T \in \{0,1\}$ at final step T, based on task correctness.

3.1 Behavior Cloning

To obtain a good initial head that is not drawing the denoiser towards chaotic behavior and to validate our environment setup, we explored supervised learning approaches to train a policy by imitating the low-confidence masking strategy. We formulate a BCE objective $L_n = -w_n[y_n \log x_n + (1 - y_n)\log(1 - x_n)]$, where x are outputs from our Transformer-parameterized head and y are the ground truth labels from a low confidence heuristic (1 if a token is masked on this step and 0 if it is denoised). We use this setup because it captures the binary nature of the problem (mask or keep unmasked). To evaluate our experiments, we use the lm-eval-harness software on the tinyGSM-8K dataset.

3.2 RL with GRPO

To continue training we use GRPO-style advantage computation, KL divergence penalty to prevent policy drift, and log-probability policy gradients for optimization.

Based on the GRPO objective defined in DeepSeekMath (Shao et al., 2024) and our MDP formulation, our objective equation is

$$J_{\mathrm{GRPO}}(\theta) = \mathbb{E}_{\substack{q\sim P(Q) \\ \{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\mathrm{old}}}}} \Bigg[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \Big(\min\big[r_{i,t}\,\hat{A}_{i,t},\ \mathrm{clip}(r_{i,t},\,1-\epsilon,\,1+\epsilon)\,\hat{A}_{i,t}\big] \\ -\ \beta\, D_{\mathrm{KL}}\big(\pi_\theta(\cdot \mid s_{i,t}) \,\big\|\, \pi_{\mathrm{ref}}(\cdot \mid s_{i,t})\big)\Big)\Bigg], \tag{1}$$

where we define

$$r_{i,t} \;=\; \frac{\pi_\theta(o_{i,t} \mid s_{i,t})}{\pi_{\theta_{\mathrm{old}}}(o_{i,t} \mid s_{i,t})},$$

$\hat{A}_{i,t}$ is the advantage estimate at step t, $\epsilon > 0$ is the PPO clipping parameter, $\beta > 0$ weights of the KL regularizer.

Advantages are calculated as $\frac{r_i - \mathrm{mean}(\{r_j\}_{j=1}^{G})}{\mathrm{std}(\{r_j\}_{j=1}^{G})}$ (Fig. 2).

Outline of our RL Setup:

Policy Update Strategy: We perform policy gradient steps targeting a single denoising step per training iteration. The policy network predicts which tokens should be masked at each step. For standard experiments, we use top-k selection and maintain a linear denoising schedule. In an extension experiment, we explore threshold-based selection that allows nonlinear denoising, terminating once all gen_len tokens have been denoised.

Training Procedure: We use the current policy without gradients to generate masks up to the target step. Then, at the target step, we apply the policy with gradient computation, adding Gumbel noise to sample G different masking options. After the target step, for each of the G samples, use low-confidence masking to complete the denoising chain and obtain a textual example (Fig. 2). We choose the time step to train on by sampling from an exponential (Fig. 3) with target probabilities $\frac{\lambda e^{-\lambda t}+\epsilon}{\sum_{t'=0}^{T-1}(e^{-\lambda t'}+\epsilon)}$, where $\epsilon = 5 \times 10^{-4}$, $T = 64$, $\lambda = 0.1$.

We use the heuristic in the post-target steps to isolate the impact of the policy beyond the target step and provide a consistent quantification of its impact on the final results. We justify this with insights from Sclocchi et al. (2024), drawing on the intuition that we can try to maximize the information present at a given denoising level to support the remaining denoising process. We use the policy for previous steps to expose the policy to diverse settings it would encounter at inference—resulting by its own previous decisions. This formulation assumes that this setup mostly isolates the role of the target step in the final rewards and does not consider the entire remasking trajectory throughout the denoising process.

Reward: We implemented a binary reward system using Flexible-Extract correctness evaluation on a subsample of the GSM8k train split. The current reward signal is 0 or 1 based on whether the final generated output is correct. To extract the answers from the model, we detokenize the sequences and use regular expression-based flexible extraction to handle variations in formatting. For the experiment with nonlinear scheduling, we add a small additional reward for saving steps, such that the reward becomes $\alpha(\text{max\_steps} - \text{termination\_steps})$ to reward using fewer steps while still prioritizing final correctness.

4 Experimental Setup

To explore the impact of masking strategy on discrete diffusion models and post-train them with RL, we set up an environment where we can collect trajectories, compute rewards. and update a remasking head. Specifically, we used the LLaDA language model, which denoises a sequence of

tokens with a given schedule in multiple steps (e.g., if we set the sequence length to 128 and denoise for 64 steps, on each step, the schedule will denoise 2 tokens). Furthermore, we hypothesize that efficiency can be further improved with our masking policy; we run an additional experiment where we select tokens that the policy selects with confidence above a threshold τ, enabling it to denoise more than 2 tokens at each step.

In the LLaDA paper, there are two heuristics on choosing which tokens to denoise: (1) uniformly random, (2) highest confidence tokens are denoised, and the rest are masked. Our initial hypothesis was that we can have an additional learnable head that can be trained to guide the denoiser with a more involved, predictable strategy. Our setup is available at https://github.com/radi-cho/diff-remasking. We initially implemented a mask strategy parameterized as a Transformer model denoted with 'MiniMaskHead' in our source. Its architecture is 2 or more Transformer encoder blocks, RoPE positional embeddings, and an MLP feed-forward in each block following this configuration. Hidden states from the previous time step of the denoiser are given as an input to the mask predictor head.

We examine multiple baselines and evaluations. We present the original heuristic approaches with random remasking, low-confidence based remasking, and low-confidence based semiautoregressive remasking (where the output is denoised in sequential blocks). From our own results, we show the results of both the naive and rescaled behavior cloning experiments, which are trained to imitate low-confidence performance. Then, we intialize the head with the BC results and perform GRPO as described above.

We benchmark the baselines and evaluations on tinyGSM8k, a dataset (100 evaluation examples) of linguistically diverse grade school math word problems (Liu et al., 2023). For metrics, we consider the flexible-extract match accuracy and the number of diffusion steps the model takes to produce the correct answer.

5 Results

5.1 Quantitative Evaluation

Table 1: Results on the tinyGSM8k benchmark with 64 steps (max steps for nonlinear approach), gen_len 128. Random heuristic, low-confidence heuristic, and low-confidence heuristic (SAR) are performed according the the methods of Nie et al. (2025) with LLaDA; the latter indicates semiautoregressive denoising with block size 8. Our BC approaches are Naive BC and Rescaled BC; our end-to-end GRPO approaches are BC + GRPO and BC + GRPO (nonlinear), where nonlinear indicates threshold-based masking without a set linear schedule.

Method	Task	n-shot	Value	Mean Steps
Random heuristic	tinyGSM8k	4	0.4924	64
Low confidence heuristic	tinyGSM8k	4	0.6079	64
Low confidence heuristic (SAR)	tinyGSM8k	4	0.6895	64
Naive BC	tinyGSM8k	4	0.5215	64
BC + GRPO	tinyGSM8k	4	0.7190	64
BC + GRPO (nonlinear)	tinyGSM8k	4	0.6898	54.3
Rescaled BC	tinyGSM8k	4	0.7300	64

5.1.1 BC

Our naive BC agent had a 16.8 percentage point loss (24.4% relative decrease) compared to the low confidence heuristic (SAR) for the tinyGSM8k task. Our rescaled BC agent had a 4.05 percentage point gain (5.9% relative improvement) compared to the low confidence heuristic (Table 5.1). The native BC agent was able to converge but gave poor results on the downstream task, while the rescaled BC agent had better performance than the low-confidence heuristic (SAR) for our experiment.

5.1.2 RL with GRPO

In our final masking policy results with BC + GRPO, we observe gains above all baselines presented in the original LLaDA paper. Our BC + GRPO approach achieves 71.90% accuracy on tinyGSM8K,

representing an 18% relative improvement over low-confidence heuristics and a 46% relative improvement over random masking. Furthermore, with nonlinear denoising schedule, we observe that we can maintain performance while reducing the number of diffusion steps required; in our analyses, the nonlinear approach requires 15.2% less diffusion steps with no accuracy loss. However, although our BC + GRPO approach showed significant gains for all baselines, it performed worse than the rescaled BC approach.

5.2 Qualitative Analysis

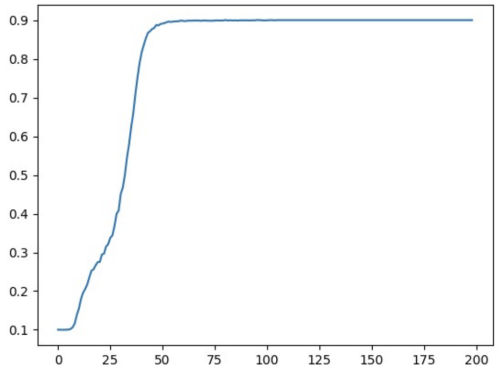

Figure 4: GRPO environment validation results; results of rewards during training to mimic low-confidence.

Before running the full GRPO training with rewards defined as above, we wanted to validate the GRPO-training environment. For this purpose, we performed a variation of BC with GRPO where we use the accuracy of mimicking low confidence as a reward. As seen in Fig 4, the GRPO implementation was able to converge to higher rewards. This is not meant to outperform the BC agent trained with SFT, but is a proof-of-concept that our RL environment can push the policy towards higher reward behavior.

Our BC + GRPO approach converged and had diverse rewards (both negative and positive samples), but compute cost limited us to not training it for long enough (over $200 provided credits + personal funds), which is why we believe our BC + GRPO approach performed worse than our rescaled BC approach.

Our initial experiment (naive behavior cloning) gave poor results on the downstream task. We investigated it and observed that since at each denoising step a lot more tokens are masked than denoised, the BC learns to predict a single class. To solve the unbalanced binary classification problem, we rescaled the positive class by roughly 1/100 (proportion of class imbalance observed empirically). This means, setting $w_n = 1/100$ when $y_n = 1$. Our rescaled BC agent had better performance than the expert, and we hypothesize this is because there is some stochasticity when evaluating downstream tasks.

In the experiments with the nonlinear denoising schedule, we observe that the policy predicts varying numbers of transition tokens that enable it to achieve high performance with the same amount of steps. We see that it tends to predict more tokens briefly at the very earliest timesteps and then primarily towards the mid-late timesteps. We can explain this based on our initial experiments of how rewards variance depended on timesteps (i.e. later timesteps showed less variation in end rewards as the main reasoning path is already determined). More tokens can be predicted at once where tokens carry similar amounts of information that impact the next steps.

6 Discussion

Our work demonstrates the potential of RL to optimize discrete diffusion models beyond traditional fixed masking strategies. The significant performance improvements achieved through learned

masking policies suggest that the conventional approaches of random or low-confidence masking may be leaving substantial performance on the table, particularly for reasoning-intensive tasks.

The success of our approach highlights several important insights about discrete diffusion models. First, the masking strategy plays a crucial role in model performance, with our learned policies achieving over 11 percentage points improvement compared to low-confidence heuristics. Second, the ability to reduce computational steps while maintaining performance suggests that learned policies capture more efficient denoising patterns that fixed schedules cannot achieve.

Several limitations warrant discussion. Our evaluation focuses primarily on mathematical reasoning tasks; broader evaluation across diverse text generation tasks would strengthen the generalizability claims. Secondly, the binary reward structure, while effective, may not capture nuanced aspects of text quality that could benefit from more sophisticated reward modeling. Furthermore, our results suggest that computational constraints prevented our BC + GRPO approach from reaching its full potential, as evidenced by the continued learning shown in our training curves and the successful generation of reward variance, but the lower performance when compared to rescaled BC.

7 Conclusion

We have presented a novel approach for optimizing discrete diffusion models through learned masking policies trained via RL. Our method addresses a key limitation of existing discrete diffusion models by moving beyond fixed or heuristic masking strategies to learn adaptive, task-specific denoising schedules.

Our work makes several key contributions to the field of discrete diffusion models. We formulate the masking decision problem as an MDP and demonstrate that RL can effectively optimize these decisions for downstream task performance. Our GRPO-based training approach, combined with BC initialization, successfully learns policies that significantly outperform traditional baselines on mathematical reasoning tasks.

The experimental results validate our approach, showing substantial improvements over existing methods. Our BC + GRPO approach achieves 71.90% accuracy on tinyGSM8K, representing an 18% relative improvement over low-confidence heuristics and a 46% relative improvement over random masking. Importantly, our nonlinear approach demonstrates that learned policies can maintain strong performance while reducing computational requirements by over 15%.

This work opens several other promising avenues for future research. First, implementing a d1-style end-to-end approach where the masking policy and denoiser are jointly optimized could yield even greater performance gains. Unlike our current approach where the base diffusion model remains frozen, end-to-end training would allow the denoiser to adapt to the learned masking strategy.

Another potential research direction involves allowing the model to re-mask tokens that have already been denoised in previous steps. Current discrete diffusion approaches treat denoised tokens as immutable, but learned policies could identify when previously denoised tokens should be reconsidered, effectively implementing an error correction mechanism. This would enable the model to refine its outputs iteratively, potentially leading to higher quality generations at the cost of additional computation.

8 Team Contributions

- **Radostin Cholakov: Worked on the initial project proposal, reviewing literature and formulating objectives; implemented BC/SFT and helped with evaluation environment; integrated GRPO with the evaluation environment and warm-up BC stages; created graphics for the poster and paper.**
- **Zeyneb Kaya: Literature; training formulation; implemented GRPO (test and end-to-end approaches); created custom lm-eval-harness based evaluation suite; performed nonlinear schedule experiments/analysis; worked on paper.**
- **Nicole Ma: Worked on the initial project proposal and formulating objectives; Literature; GRPO advantage, loss, and objective formulations; made poster and wrote paper.**

Changes from Proposal Zeyneb and Radostin ended up working on the evaluation/benchmarking framework instead of Nicole, and Nicole instead worked on GRPO advantage, loss, and objective formulations.

References

Alex Havrilla, Yuqing Du, Sharath Chandra Raparthy, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, Sainbayar Sukhbaatar, and Roberta Raileanu. 2024. Teaching Large Language Models to Reason with Reinforcement Learning. arXiv:2403.04642 [cs.LG] https://arxiv.org/abs/2403.04642

Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. 2022. Diffusion-LM Improves Controllable Text Generation. arXiv:2205.14217 [cs.CL] https://arxiv.org/abs/2205.14217

Bingbin Liu, Sebastien Bubeck, Ronen Eldan, Janardhan Kulkarni, Yuanzhi Li, Anh Nguyen, Rachel Ward, and Yi Zhang. 2023. TinyGSM: achieving >80% on GSM8k with small language models. arXiv:2312.09241 [cs.LG] https://arxiv.org/abs/2312.09241

Sulin Liu, Juno Nam, Andrew Campbell, Hannes Stärk, Yilun Xu, Tommi Jaakkola, and Rafael Gómez-Bombarelli. 2025. Think While You Generate: Discrete Diffusion with Planned Denoising. arXiv:2410.06264 [cs.LG] https://arxiv.org/abs/2410.06264

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. 2025. Large language diffusion models.

Antonio Sclocchi, Alessandro Favero, and Matthieu Wyart. 2024. A Phase Transition in Diffusion Models Reveals the Hierarchical Nature of Data. arXiv:2402.16991 [stat.ML] https://arxiv.org/abs/2402.16991

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv:2402.03300 [cs.CL] https://arxiv.org/abs/2402.03300

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

Oussama Zekri and Nicolas Boulle. 2025. Fine-Tuning Discrete Diffusion Models with Policy Gradient Methods.

Siyan Zhao, Devaansh Gupta, Qinqing Zheng, and Aditya Grover. 2025. d1: Scaling Reasoning in Diffusion Large Language Models via Reinforcement Learning.

Hao Zou, Zae Myung Kim, and Dongyeop Kang. 2023. A Survey of Diffusion Models in Natural Language Processing. arXiv:2305.14671 [cs.CL] https://arxiv.org/abs/2305.14671