DTCA: DENOISING TOKEN CREDIT ASSIGNMENT FOR REINFORCEMENT LEARNING IN DIFFUSION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion language models (DLMs) generate text through iterative denoising, enabling bidirectional context modeling and parallel token refinement. Recent studies have applied reinforcement learning (RL) to DLMs and improved their reasoning performance. However, the iterative denoising process makes credit assignment more challenging, as final outcome rewards provide limited information about how different tokens contribute to the final output. In this paper, we propose **Denoising Token Credit Assignment (DTCA)**, a framework for token-level credit assignment in reinforcement learning for diffusion language models. DTCA estimates token contributions by measuring the improvement in prediction confidence of remaining masked positions after token commitment and groups these signals according to part-of-speech (POS) categories. These contribution weights are used to select informative denoising timesteps and assign token-level learning signals during RL training. DTCA further calibrates these learning signals according to token-level entropy changes induced by policy updates. Experiments on LLaDA-8B-Instruct show that DTCA consistently improves mathematical reasoning performance across different generation budgets, with gains of up to 9.0 points on MATH500 and 5.3 points on GSM8K over the base model.

1 INTRODUCTION

Diffusion language models (DLMs) have recently attracted increasing attention as an emerging paradigm for language modeling due to their ability to model bidirectional context and perform parallel token refinement (Labs et al., 2025; Google DeepMind, 2025; Song et al., 2025), offering an alternative to autoregressive generation (Nie et al., 2025; Sahoo et al., 2024; Ye et al., 2025a; Yu et al., 2025). Recent studies have explored reinforcement learning for DLMs by leveraging the generation process, mainly constructing learning signals from denoising trajectories or intermediate states during training (He et al., 2026; Wang et al., 2025b; Oba et al., 2026; Monsefi et al., 2026; Huang et al., 2025). However, DLMs generate sequences through multiple denoising steps, where tokens are gradually committed and each commitment can affect the refinement of remaining masked positions. (Ou et al., 2025).Therefore, RL optimization in DLMs requires finer-grained credit assignment that considers how different tokens affect subsequent refinement.

Achieving finer grained credit assignment requires estimating the contribution of individual tokens during DLM denoising. Specifically, we measure token contribution by how much a committed token improves the model's prediction confidence on the remaining masked positions. However, directly estimating such token-level confidence improvements during RL training is challenging because multiple tokens are committed at the same denoising step and interact through the shared denoising process, making their individual effects difficult to isolate. Moreover, the estimated contribution can vary substantially across different rollout samples. To obtain more reliable estimates, we categorize tokens by part-of-speech (POS) categories and aggregate their refinement effects within each category. POS categories provide a natural grouping criterion because they capture different linguistic functions, with nouns and verbs typically carrying lexical information while determiners and adpositions mainly providing grammatical structure (de Marneffe et al., 2021).

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.

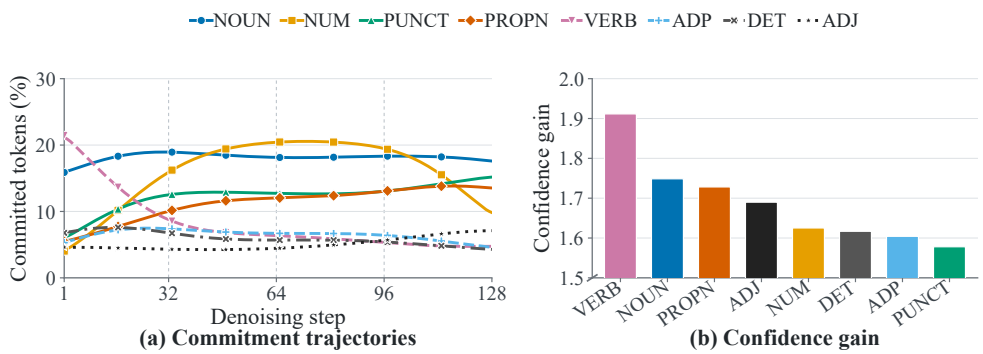


Figure 1: Preliminary analysis of POS-level differences in DLM denoising using rollouts generated from the GSM8K (Cobbe et al., 2021) training set. (a) Commitment timing of different POS categories across denoising steps. (b) Improvement in the prediction confidence of remaining masked positions after individual token commitment, averaged within each POS category. POS tags denote NOUN (noun), NUM (numeral), PUNCT (punctuation), PROPN (proper noun), VERB (verb), ADP (adposition), DET (determiner), and ADJ (adjective).

The denoising behaviors of different POS categories are further analyzed using rollouts generated from the GSM8K (Cobbe et al., 2021) dataset. As shown in Figure 1(a), different POS categories exhibit distinct commitment trajectories across denoising steps. NOUN maintains a relatively high commitment ratio, whereas VERB has a high ratio at the beginning that declines substantially over the denoising process. PROPN increases during early steps and then remains relatively stable, while NUM rises toward the middle of denoising and declines in later steps. Other categories, such as ADP, DET, and ADJ, exhibit relatively lower commitment ratios with more stable variations. Figure 1(b) further shows the confidence gain associated with different POS categories. Clear differences emerge across categories: VERB achieves the largest confidence gain, followed by NOUN and PROPN, whereas categories such as DET and ADP show relatively smaller gains. These observations indicate that POS categories capture different denoising behaviors and refinement effects, making them a useful basis for grouping tokens and estimating contribution weights.

Based on these observations, we propose **Denoising Token Credit Assignment (DTCA)**, a token-level credit assignment framework for reinforcement learning in diffusion language models. DTCA estimates token contributions from denoising refinement signals and aggregates them into POS-level contribution weights to obtain more stable and interpretable learning signals. These contribution signals are then used to select informative denoising timesteps and derive token-level learning signals during RL optimization. DTCA further incorporates an entropy-based credit calibration module inspired by OPEFO (Xu et al., 2026) to calibrate the learning signals according to token-level entropy changes induced by policy updates. Experiments on LLaDA-8B-Instruct show that DTCA consistently improves mathematical reasoning performance across different generation budgets, with improvements of up to 9.0 percentage points on MATH500 (Lightman et al., 2024) and 5.3 percentage points on GSM8K (Cobbe et al., 2021) over the base model.

The main contributions are summarized as follows:

- We analyze different denoising behaviors across POS categories in diffusion language models and use POS categories to estimate the effect of token commitments on subsequent refinement.
- We propose **DTCA**, a token-level credit assignment framework that aggregates denoising refinement signals by POS categories and uses the resulting contribution weights for denoising timestep selection and token credit assignment. DTCA further adapts entropy-based credit calibration from OPEFO (Xu et al., 2026) to the diffusion language model setting for improved RL optimization.
- Extensive experiments demonstrate the effectiveness of DTCA on mathematical reasoning benchmarks and validate the benefits of its key components.

2 PRELIMINARIES

2.1 DIFFUSION LANGUAGE MODEL DENOISING

Masked diffusion language models generate responses through iterative denoising (Nie et al., 2025; Sahoo et al., 2024). Given a prompt q, let $\tau = (x_T, x_{T-1}, \ldots, x_0)$ denote a sampled trajectory, where x_t is the partially masked response at denoising step t and x_0 is the completed response. Under the monotonic unmasking schedule used in this work, each step reveals a subset of masked positions as predicted tokens; we call this *token commitment*. Committed tokens remain visible in subsequent states and can influence predictions at the remaining masked positions. The distribution $\pi_\theta(y \mid q)$ below is the response distribution induced by this iterative sampling procedure, not an autoregressive token factorization.

2.2 REINFORCEMENT LEARNING FOR DIFFUSION LANGUAGE MODELS

Given prompts $q \sim \mathcal{D}$ and sampled responses $y \sim \pi_\theta(\cdot \mid q)$, reinforcement learning seeks to maximize expected sequence-level reward:

$$\mathcal{J}(\theta) = \mathbb{E}_{q\sim\mathcal{D},\, y\sim\pi_\theta(\cdot|q)}[R(q, y)]. \tag{1}$$

Group-relative policy optimization converts rewards from multiple responses to the same prompt into response advantages (Shao et al., 2024). In masked diffusion RL, these advantages can be applied to predictions at selected denoising states (He et al., 2026). Let $\mathcal{T}$ be the selected steps, $\mathcal{B}_t$ their optimization target positions, and v_i the target token at position i. A schematic token-level surrogate is

$$\mathcal{L}_{\mathrm{RL}}(\theta) = -\mathbb{E}\left[\frac{1}{N}\sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{B}_t} A_{t,i}\log\pi_\theta(v_i \mid q, x_t)\right] + \beta_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}}, \tag{2}$$

where $N = \sum_{t\in\mathcal{T}}|\mathcal{B}_t|$ and $\mathcal{L}_{\mathrm{KL}}$ denotes a penalty relative to a frozen reference policy (Shao et al., 2024; He et al., 2026). The exact target set, credit $A_{t,i}$, KL estimator, and update loss used by DTCA are specified below and in Appendix B. A sequence-level reward alone does not identify which commitments help subsequent denoising; this motivates assigning credit at the denoising-state and target-token levels.

3 METHOD

We propose **Denoising Token Credit Assignment (DTCA)**, a token credit assignment framework for reinforcement learning in diffusion language models. As illustrated in Figure 2, DTCA estimates token contributions from denoising refinement dynamics and uses the resulting contribution weights for informative denoising state selection and token-level credit assignment. DTCA further calibrates the token-level learning signals according to entropy changes induced by policy updates.

3.1 TOKEN CONTRIBUTION ESTIMATION BY POS CATEGORY

As illustrated in Figure 2, DTCA estimates token-level refinement contributions from denoising dynamics and aggregates these estimates into category-level contribution weights. Given a denoising trajectory τ, different token commitments may have different effects on subsequent refinement. DTCA therefore estimates the refinement contribution of each committed token by measuring the improvement in prediction confidence of the remaining masked positions after the commitment.

Specifically, let $\mathcal{M}_t$ denote the remaining masked positions before a token commitment at denoising step t. We define the confidence of a denoising state as the average probability assigned to the target tokens of these remaining positions:

$$F(x_t) = \frac{1}{|\mathcal{M}_t|}\sum_{j\in\mathcal{M}_t} p_\theta(y_j|x_t), \tag{3}$$

where y_j denotes the target token at position j. For a committed token v_i, its refinement contribution estimate is computed as:

$$\Delta P_i = F(x_t^{(i)}) - F(x_t), \tag{4}$$

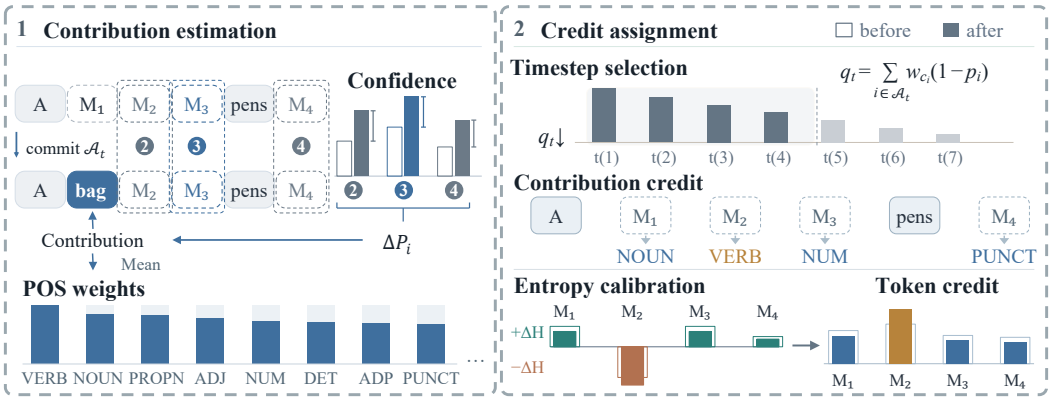


Figure 2: Overview of Denoising Token Credit Assignment (DTCA). DTCA estimates token contributions from denoising refinement dynamics and uses the resulting contribution weights for denoising state selection and token-level credit assignment. The obtained learning signals are further calibrated according to entropy changes induced by policy updates for reinforcement learning optimization.

where $x_t^{(i)}$ denotes the denoising state after committing token v_i individually. A larger ΔP_i indicates a stronger estimated refinement effect of the committed token on subsequent denoising.

Obtaining reliable token-level estimates directly during RL training is challenging because evaluating individual token effects requires re-evaluating token commitments with additional forward passes, introducing substantial computational overhead during policy optimization. Moreover, these estimates can vary across different rollout samples. Therefore, DTCA constructs contribution estimates from offline denoising rollouts before RL training and groups them according to part-of-speech (POS) categories to obtain more stable category-level contribution weights.

During offline estimation, DTCA uses a denoising budget equal to half of the maximum generation length. At each denoising step, two tokens are committed, and DTCA evaluates each token separately through additional forward passes based on the confidence improvement of remaining masked positions. These estimates are aggregated at the POS level to reduce variance across rollouts.

For each committed token v_i, we assign its POS category as $c_i = \phi(v_i)$. The contribution of each POS category is obtained by averaging the refinement contribution estimates of tokens belonging to that category:

$$m_c = \frac{1}{N_c} \sum_{i:c_i=c} \Delta P_i, \tag{5}$$

where N_c denotes the number of committed tokens in POS category c. The resulting category-level contribution m_c represents the average refinement effect estimated for tokens in each category.

POS Contribution Weight Construction. The estimated POS contributions are normalized into bounded contribution weights:

$$w_c = 0.8 + 0.4 \frac{m_c - \min_{j \in \mathcal{C}} m_j}{\max_{j \in \mathcal{C}} m_j - \min_{j \in \mathcal{C}} m_j}, \tag{6}$$

where $\mathcal{C}$ denotes the set of POS categories. The weight w_c lies in the range $[0.8, 1.2]$ and represents the scaling factor derived from the estimated refinement contribution of category c. We constrain this range to prevent excessive credit scaling while preserving the original RL objective. These weights are used to adjust token-level credits during RL optimization.

3.2 Contribution-Based Timestep Selection

The category contribution weights are also used to identify informative denoising timesteps for RL training. Following RLDF (He et al., 2026), DTCA optimizes only a subset of denoising timesteps

from each rollout. For a denoising timestep t, let $\mathcal{A}_t$ denote the set of tokens committed at this step. We compute the importance score of each timestep as:

$$q_t = \sum_{i \in \mathcal{A}_t} w_{c_i}(1 - p_i), \tag{7}$$

where p_i denotes the confidence of token i when it is committed and w_{c_i} denotes the contribution weight of its POS category. The term $1 - p_i$ assigns higher scores to timesteps containing lower-confidence token commitments, while w_{c_i} incorporates the estimated refinement contribution of the committed tokens.

DTCA ranks denoising timesteps according to q_t and selects the smallest set of timesteps whose cumulative score reaches a predefined fraction μ of the total score. The number of selected timesteps therefore adapts to the distribution of contribution signals in each rollout. Lower and upper bounds are applied to the number of selected timesteps for stable and efficient training.

3.3 Contribution-Based Credit Assignment

The category contribution weights are used to assign different credits to tokens within each selected denoising timestep. For a selected timestep t, let $\mathcal{B}_t$ denote the tokens used for optimization, including the committed tokens at timestep t and the retained masked positions selected by the threshold-based filtering strategy in RLDF (He et al., 2026). POS categories for all tokens in $\mathcal{B}_t$ are obtained from the completed rollout, where retained masked positions are associated with their final predicted tokens. The credit weight of token v_i is determined by the contribution weight of its POS category:

$$a_{t,i} = \frac{w_{c_i}}{\frac{1}{|\mathcal{B}_t|} \sum_{k \in \mathcal{B}_t} w_{c_k}}, \tag{8}$$

where c_i is the POS category of token v_i. The normalization keeps the average credit within each timestep equal to one, so that the overall scale of the timestep advantage remains unchanged. Given the original timestep advantage A_t, DTCA computes the token-level learning signal as

$$A_{t,i} = a_{t,i} A_t. \tag{9}$$

Tokens from categories with larger contribution weights receive stronger learning signals, while tokens from categories with smaller contribution receive weaker signals. In this way, DTCA transforms the original response-level feedback into more fine grained token credit during RL training.

3.4 Entropy-Based Credit Calibration

Although DTCA provides token-level learning signals according to denoising contribution, different denoising states may induce different entropy changes during policy updates. To further calibrate the token-level learning signals, DTCA incorporates an entropy-based credit calibration module inspired by OPEFO (Xu et al., 2026). Unlike the original formulation for autoregressive generation, we adapt the token-level entropy change estimation to selected denoising states and their remaining target positions in diffusion language models.

For each response b, selected denoising state t, and remaining target position i, the entropy change induced by the policy update is estimated as:

$$\Delta H_{t,i}^{(b)} = -\eta A_{t,i}^{(b)} (1 - p_i^{(b)})^2 (\log p_i^{(b)} + H_i^{(b)}), \tag{10}$$

where $p_i^{(b)}$ denotes the probability assigned by the current policy to the target token and $H_i^{(b)}$ denotes the entropy of the full prediction distribution at this position. This formulation follows the first-order entropy change estimation of OPEFO (Xu et al., 2026) while replacing autoregressive decoding steps with selected denoising states and target positions.

Within each response, the entropy-increasing and entropy-decreasing updates are separated according to the sign of $\Delta H_{t,i}^{(b)}$:

$$S_b^+ = \{(t,i) | \Delta H_{t,i}^{(b)} > 0\}, \qquad S_b^- = \{(t,i) | \Delta H_{t,i}^{(b)} < 0\}. \tag{11}$$

Following OPEFO (Xu et al., 2026), we compute a separate entropy flow balancing coefficient for each response:

$$\lambda_b = \mathrm{clip}\left(\frac{\sum_{(t,i)\in S_b^-} |\Delta H_{t,i}^{(b)}| - \sum_{(t,i)\in S_b^+} \Delta H_{t,i}^{(b)}}{\sum_{(t,i)\in S_b^-} |\Delta H_{t,i}^{(b)}| + \sum_{(t,i)\in S_b^+} \Delta H_{t,i}^{(b)}}, -0.95, 0.95\right). \tag{12}$$

The entropy balancing coefficient is then used to rescale token-level learning signals to balance entropy-increasing and entropy-decreasing updates. When $\lambda_b > 0$, entropy-increasing updates are amplified and entropy-decreasing updates are attenuated. When $\lambda_b < 0$, the adjustment is reversed. When $\lambda_b = 0$, the credits remain unchanged:

$$\hat{A}_{t,i}^{(b)} = \begin{cases} (1+\lambda_b)A_{t,i}^{(b)}, & \Delta H_{t,i}^{(b)} > 0, \\ (1-\lambda_b)A_{t,i}^{(b)}, & \Delta H_{t,i}^{(b)} \le 0. \end{cases} \tag{13}$$

4 Experiments

4.1 Experimental Setup

Model and data. We initialize RL training from LLaDA-8B-Instruct (Nie et al., 2025) without additional SFT. Following RLDF (He et al., 2026), we use the MATH (Hendrycks et al., 2021) training split with the same data processing pipeline for RL optimization. To construct the fixed POS contribution table, DTCA collects offline denoising rollouts from the training data of MATH and GSM8K and builds a shared POS contribution table without updating model parameters.

Training. We train LLaDA-8B-Instruct with reinforcement learning without an intermediate SFT stage. For each rollout, DTCA selects denoising states with the highest POS-weighted confidence priority until covering 70% of the total priority mass, with the number of selected states constrained between 2 and 32. We perform policy optimization using group-relative advantages and regularize the policy against a frozen reference model with the k3 KL estimator (Schulman, 2020). Rollouts are generated using block-wise decoding with KV cache, following Fast-dLLM (Wu et al., 2026b). Full implementation details and hyperparameters are provided in Appendix C.

Evaluation and baselines. We evaluate mathematical reasoning performance on MATH500 (Lightman et al., 2024) and GSM8K (Cobbe et al., 2021) using final-answer accuracy. Following prior work, we evaluate all models under generation budgets of 128, 256, and 512, with denoising steps matched to the corresponding maximum generation lengths. We compare DTCA with LLaDA-8B-Instruct, Uniform SFT, LLaDA-1.5, and representative reinforcement learning methods for diffusion language models, including d1 (Zhao et al., 2025), ESPO (Ou et al., 2026), TraceRL (Wang et al., 2025b), Coupled-GRPO (Gong et al., 2026), and RLDF (He et al., 2026). Additional details on training configurations and baseline evaluation settings are provided in Appendix C.2 and Appendix C.4.

4.2 Main Results

Table 1 compares DTCA with supervised and reinforcement learning baselines on MATH500 and GSM8K. DTCA consistently improves over LLaDA-8B-Instruct across both benchmarks and all evaluated generation budgets. On MATH500, DTCA improves the base model by 9.0, 6.8, and 4.8 percentage points at budgets of 128, 256, and 512, respectively. On GSM8K, the corresponding improvements are 5.3, 2.1, and 2.5 percentage points. These results show that DTCA provides consistent gains under different denoising budgets rather than relying on a specific inference setting. Compared with supervised optimization and existing diffusion language model reinforcement learning methods, DTCA achieves competitive performance across the reported settings, demonstrating the effectiveness of the proposed token-level credit assignment framework.

The largest improvements on both benchmarks occur at the smallest evaluated generation budget of 128, while gains remain positive at budgets of 256 and 512. Although the relative improvements become smaller at larger generation budgets, DTCA consistently improves over the base model, highlighting its stronger gains under limited denoising computation.

Table 1: Mathematical reasoning accuracy (%) on MATH500 and GSM8K. The last rows report DTCA, its ablation variants, and the improvement over the base model. Boldface indicates the best result in each column, and Δ denotes the improvement over LLaDA-8B-Instruct.

	MATH500			GSM8K		
Method	128	256	512	128	256	512
LLaDA-8B-Instruct (Nie et al., 2025)	30.8	36.4	39.2	75.3	83.2	82.9
LLaDA-1.5 (Zhu et al., 2026)	32.6	39.4	39.8	76.2	81.6	83.0
Uniform SFT	24.6	30.0	32.6	74.8	80.7	78.6
d1 (Zhao et al., 2025)	33.6	37.6	37.2	75.8	82.4	81.8
ESPO (Ou et al., 2026)	37.6	38.0	40.2	78.1	83.2	84.0
TraceRL (Wang et al., 2025b)	32.6	34.2	34.0	76.6	80.4	81.8
Coupled-GRPO (Gong et al., 2026)	35.2	40.2	40.2	78.0	83.6	83.2
RLDF (He et al., 2026)	36.2	41.2	42.6	77.2	83.4	82.7
DTCA w/o Entropy Calibration	37.4	41.8	39.8	78.3	83.9	81.9
DTCA w/o Contribution Credit	36.2	39.4	41.2	77.1	82.9	82.3
DTCA (ours)	**39.8**	**43.2**	**44.0**	**80.6**	**85.3**	**85.4**
Δ vs. Base	+9.0	+6.8	+4.8	+5.3	+2.1	+2.5

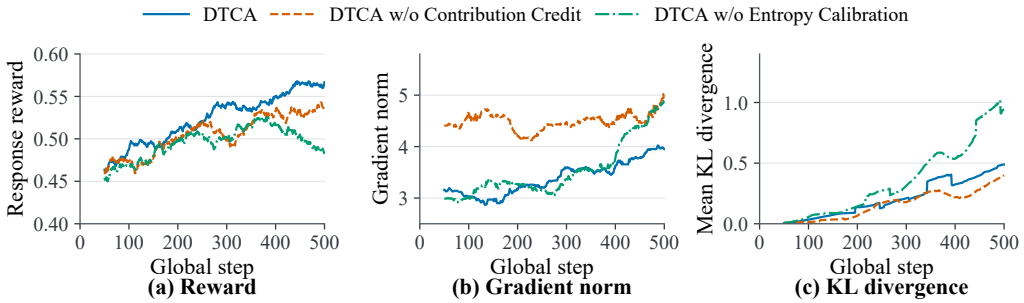


Figure 3: Optimization dynamics of DTCA variants during RL training. (a) Response reward, (b) gradient norm, and (c) mean KL divergence. DTCA is compared with variants without Contribution Credit and Entropy Calibration.

4.3 Ablation Study

Table 1 reports ablation results for different DTCA components. *DTCA w/o Contribution Credit* replaces the contribution-based token credits with uniform credits while keeping the same timestep selection strategy. *DTCA w/o Entropy Calibration* removes the entropy-based credit calibration module while preserving contribution-based credit assignment and timestep selection.

Removing either component decreases performance compared with the full DTCA model across both benchmarks and different generation budgets. At the 256 generation budget, removing Contribution Credit leads to decreases of 3.8 and 2.4 percentage points on MATH500 and GSM8K, respectively, while removing Entropy Calibration results in smaller drops of 1.4 and 1.4 points. At the 512 generation budget, removing Entropy Calibration causes larger performance decreases, with drops of 4.2 and 3.5 points on MATH500 and GSM8K. These results show that both components contribute to the overall performance of DTCA.

4.4 Analysis of Optimization Dynamics

To understand how different DTCA components influence RL optimization, we analyze training dynamics in terms of reward, gradient norm, and KL divergence. Figure 3 compares DTCA with variants without Contribution Credit and Entropy Calibration during training.

Removing Contribution Credit results in similar reward trajectories to DTCA, but leads to substantially different update behaviors. In particular, the variant with uniform credit assignment consis-

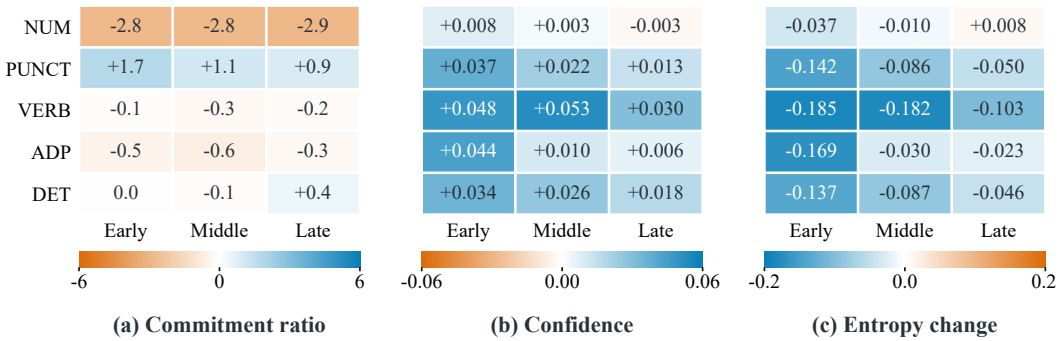


Figure 4: Analysis of commitment dynamics on MATH500. DTCA is compared with the base model across three denoising stages. Each cell shows the difference in (a) token commitment ratio, (b) confidence at commitment, and (c) entropy at commitment.

tently exhibits larger gradient norms throughout training while achieving comparable reward improvements. This suggests that contribution-based credit assignment mainly affects the quality of token-level learning signals, allowing the model to optimize with different update dynamics rather than simply changing the final reward trend. By considering token-level contribution differences, DTCA provides a more structured learning signal than uniform credit assignment.

Removing Entropy Calibration mainly affects the stability of later-stage policy updates. Although its reward trajectory initially follows a similar trend, the variant without entropy calibration shows a faster increase in KL divergence and lower reward in later training stages. This indicates that entropy calibration helps control policy changes by adjusting learning signals according to entropy changes, leading to more stable optimization during RL training.

4.5 Analysis of Token Commitment Dynamics

To understand whether contribution-based credit assignment changes the denoising behavior of diffusion language models, we analyze token commitment patterns after RL training. Token commitment reflects when tokens become fixed during denoising, while the commitment distribution and prediction states provide insights into the learned denoising behavior. Figure 4 compares DTCA with the base model on MATH500 across early, middle, and late denoising stages. Each cell reports the difference between DTCA and the base model in token commitment ratio, prediction confidence at commitment, and target entropy at commitment.

DTCA changes the commitment distribution across token categories rather than uniformly shifting all commitments. For example, DTCA consistently reduces the commitment ratio of NUM tokens while increasing the commitment ratio of PUNCT tokens across different denoising stages. These changes suggest that DTCA learns different commitment patterns across token categories instead of simply accelerating or delaying all token predictions. Notably, changes in commitment frequency do not always correspond to changes in commitment states. For example, VERB shows limited changes in commitment ratio but substantial increases in confidence and reductions in entropy at commitment. This indicates that DTCA influences not only which tokens are committed, but also the model states when commitments occur. The observed changes are consistent with leveraging denoising refinement signals to improve token-level credit assignment during reinforcement learning.

4.6 Analysis of Entropy-Based Credit Calibration

To understand how entropy-based credit calibration affects token-level learning signals, we analyze entropy changes during RL training. Inspired by the entropy flow analysis in OPEFO (Xu et al., 2026), which studies how policy updates affect entropy dynamics in autoregressive models, DTCA extends this analysis to diffusion language models. Considering the different roles of tokens during denoising, we further analyze entropy changes according to POS categories and adjust token-level credits based on these entropy dynamics. Figure 5 compares DTCA with and without entropy calibration across different POS categories.

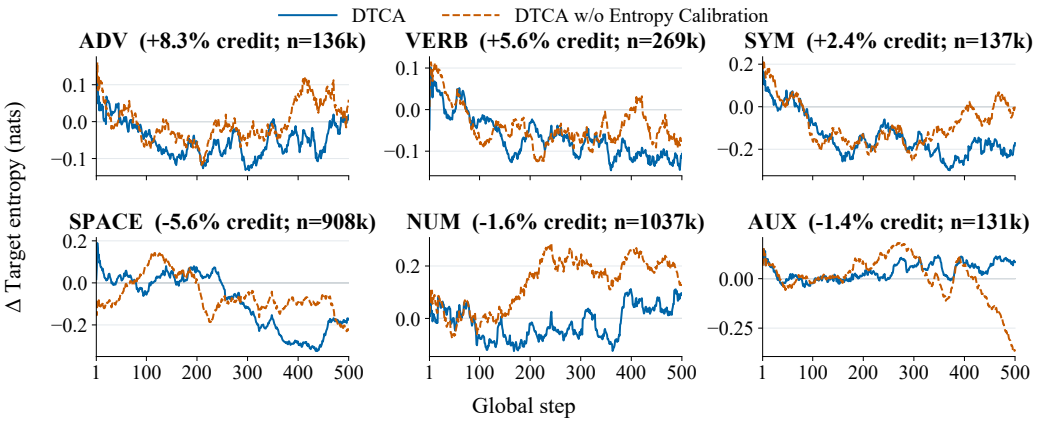


Figure 5: Analysis of entropy-based credit calibration. Each panel shows the target entropy change of a POS category during RL training. The blue and orange curves represent DTCA and DTCA without entropy calibration, respectively. Values in parentheses show the change in assigned credit after entropy calibration.

Different token categories exhibit different entropy changes during training, showing that policy updates affect token uncertainty in different ways. Values in parentheses indicate the relative change in assigned credit after entropy calibration. The adjustments vary across categories: ADV, VERB, and SYM receive increased credits, while SPACE, NUM, and AUX receive reduced credits. These category-specific adjustments show that entropy calibration is not a fixed category-based weighting scheme, but instead adapts credits according to the entropy effects of different token updates. Unlike applying the same entropy adjustment to all updates, DTCA further calibrates the already assigned token-level credits, where each token has its own contribution-based learning signal. Therefore, the final credit assignment combines contribution information with entropy dynamics, providing a more fine-grained adjustment of token-level optimization signals.

5 Conclusion

We presented **Denoising Token Credit Assignment (DTCA)**, a framework for fine-grained token-level credit assignment in reinforcement learning for diffusion language models. DTCA estimates token contributions from the confidence improvement of remaining masked positions after token commitment, aggregates these signals by POS categories, and uses the resulting contribution weights for denoising timestep selection and token-level credit assignment. DTCA further calibrates contribution credits according to entropy changes induced by policy updates, balancing entropy-increasing and entropy-decreasing updates. On LLaDA-8B-Instruct, DTCA improves the reproduced base model by 6.8 percentage points on MATH500 and 2.1 points on GSM8K under the 256-step evaluation setting. Training dynamics and component ablations show that contribution-based credit assignment improves token-level learning signals, while entropy-based calibration helps maintain more stable policy updates. These results demonstrate that denoising dynamics provide useful information for fine-grained credit assignment in reinforcement learning for diffusion language models.

6 Reproducibility Statement

Our experiments are based on the publicly available LLaDA-8B-Instruct checkpoint (Nie et al., 2025) and publicly available datasets, including MATH (Hendrycks et al., 2021) and GSM8K (Cobbe et al., 2021). We provide the core implementation of DTCA and key experimental configurations to facilitate verification of our approach. Detailed training settings, baseline configurations, and evaluation protocols are described in the Appendix. Upon acceptance, we will release the complete codebase, training scripts, and additional resources to support full reproduction of our results.

7 AI USE STATEMENT

Generative AI tools were used to assist with literature organization, experimental result analysis, and language editing. The authors are responsible for the ideas, methods, implementation, experiments, and conclusions presented in this paper. Generative AI tools were not used to develop the proposed method, write the main sections of the paper, generate experimental results, create synthetic datasets, or prove mathematical claims.

REFERENCES

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 17981–17993. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/958c530554f78bcd8e97125b70e6973d-Paper.pdf.

Daixuan Cheng, Shaohan Huang, Xuekai Zhu, Bo Dai, Xin Zhao, Zhenliang Zhang, and Furu Wei. Reasoning with exploration: An entropy perspective. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(36):30377–30385, 2026.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

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

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. Universal dependencies. *Computational Linguistics*, 47(2):255–308, 07 2021. ISSN 0891-2017. doi: 10.1162/coli_a_00402. URL https://doi.org/10.1162/coli_a_00402.

Sander Dieleman, Laurent Sartran, Arman Roshannai, Nikolay Savinov, Yaroslav Ganin, Pierre H. Richemond, Arnaud Doucet, Robin Strudel, Chris Dyer, Conor Durkan, Curtis Hawthorne, Rémi Leblond, Will Grathwohl, and Jonas Adler. Continuous diffusion for categorical data, 2022. URL https://arxiv.org/abs/2211.15089.

Zitian Gao, Haoming Luo, Lynx Chen, Jason Klein Liu, Ran Tao, Joey Zhou, and Bryan Dai. What makes diffusion language models super data learners?, 2025. URL https://arxiv.org/abs/2510.04071.

Shansan Gong, Ruixiang Zhang, Huangjie Zheng, Jiatao Gu, Navdeep Jaitly, Lingpeng Kong, and Yizhe Zhang. Diffucoder: Understanding and improving masked diffusion models for code generation. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 15723–15750, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/1a5b00e59818e4be1c89855a8b921903-Paper-Conference.pdf.

Google DeepMind. Gemini diffusion. https://deepmind.google/models/gemini-diffusion, 2025. Accessed: 2025-05-24.

Ishaan Gulrajani and Tatsunori B Hashimoto. Likelihood-based diffusion language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 16693–16715. Curran Associates, Inc., 2023. doi: 10.52202/075280-0730. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/35b5c175e139bff5f22a5361270fce87-Paper-Conference.pdf.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Haoyu He, Katrin Renz, Yong Cao, and Andreas Geiger. Mdpo: Overcoming the training-inference divide of masked diffusion language models, 2025. URL https://arxiv.org/abs/2508.13148.

Qi He, Huan Chen, Ya Guo, Huijia Zhu, Yi R. Fung, and Baojian Zhou. Reinforcement learning from denoising feedback, 2026. URL https://arxiv.org/abs/2605.25638.

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, 2021. URL https://arxiv.org/abs/2103.03874.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 12454–12465. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/67d96d458abdef21792e6d8e590244e7-Paper.pdf.

Zemin Huang, Zhiyang Chen, Zijun Wang, Tiancheng Li, and Guo-Jun Qi. Reinforcing the diffusion chain of lateral thought with diffusion language models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 152677–152710. Curran Associates, Inc., 2025. doi: 10.52202/085713-5105. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/dffd1c523512e557f4e75e8309049213-Paper-Conference.pdf.

Haozhe Jiang, Haiwen Feng, Pieter Abbeel, Jiantao Jiao, Angjoo Kanazawa, and Nika Haghtalab. Dipod: Diffusion policy optimization without drifting apart, 2026. URL https://arxiv.org/abs/2606.13795.

Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron Courville, and Nicolas Le Roux. Vineppo: Refining credit assignment in rl training of llms, 2025. URL https://arxiv.org/abs/2410.01679.

Inception Labs, Samar Khanna, Siddhant Kharbanda, Shufan Li, Harshit Varma, Eric Wang, Sawyer Birnbaum, Ziyang Luo, Yanis Miraoui, Akash Palrecha, Stefano Ermon, Aditya Grover, and Volodymyr Kuleshov. Mercury: Ultra-fast language models based on diffusion, 2025. URL https://arxiv.org/abs/2506.17298.

Jiahui Li, Lin Li, Tai-Wei Chang, Kun Kuang, Long Chen, Jun Zhou, and Cheng Yang. RED: Unleashing token-level rewards from holistic feedback via reward redistribution. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 4993–5022, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.252. URL https://aclanthology.org/2025.emnlp-main.252/.

Shufan Li, Konstantinos Kallidromitis, Akash Gokul, Yuta Kyuragi, and Aditya Grover. Guidance contrastive token credit assignment for discrete policy optimization, 2026. URL https://arxiv.org/abs/2605.29198.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 4328–4343. Curran Associates, Inc., 2022. doi: 10.52202/068431-0313. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/1be5bc25d50895ee656b8c2d9eb89d6a-Paper-Conference.pdf.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/aca97732e30bcf1303bc22ac3924fd16-Paper-Conference.pdf.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution, 2024. URL https://arxiv.org/abs/2310.16834.

Amin Karimi Monsefi, Dominic Culver, Nikhil Bhendawade, Lokesh Boominathan, Manuel R. Ciosici, Yizhe Zhang, and Irina Belousova. Daca-grpo: Denoising-aware credit assignment for reinforcement learning in diffusion language models, 2026. URL https://arxiv.org/abs/2605.16342.

Jinjie Ni, Qian Liu, Longxu Dou, Chao Du, Zili Wang, Hang Yan, Tianyu Pang, and Michael Qizhe Shieh. Diffusion language models are super data learners, 2025. URL https://arxiv.org/abs/2511.03276.

Zanlin Ni, Shenzhi Wang, Yang Yue, Tianyu Yu, Weilin Zhao, Yeguo Hua, Tianyi Chen, Jun Song, Cheng Yu, Bo Zheng, and Gao Huang. The flexibility trap: Rethinking the value of arbitrary order in diffusion language models, 2026. URL https://arxiv.org/abs/2601.15165.

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. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 50608–50646. Curran Associates, Inc., 2025. doi: 10.52202/085713-1689. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/48b383b24230e0e6e649d9c98dae4d8c-Paper-Conference.pdf.

Daisuke Oba, Hiroki Furuta, and Naoaki Okazaki. Diffusion-state policy optimization for masked diffusion language models, 2026. URL https://arxiv.org/abs/2602.06462.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 64972–65009, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/a365e37c18fb91af547a2f0012a89e98-Paper-Conference.pdf.

Jingyang Ou, Jiaqi Han, Minkai Xu, Shaoxuan Xu, Jianwen Xie, Stefano Ermon, Yi Wu, and Chongxuan Li. Principled rl for diffusion llms emerges from a sequence-level perspective. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 4941–4966, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/0907335ecf28faf15be54485dbcbe70e-Paper-Conference.pdf.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022. doi: 10.52202/

068431-2011. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Mihir Prabhudesai, Mengning Wu, Amir Zadeh, Katerina Fragkiadaki, and Deepak Pathak. Diffusion beats autoregressive in data-constrained settings. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 10581–10606. Curran Associates, Inc., 2025. doi: 10.52202/085713-0354. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/0f705a932553c08ebf0d1bc520b7cbc6-Paper-Conference.pdf.

Haran Raajesh, Kulin Shah, Adam Klivans, and Philipp Krähenbühl. Mask-aware policy gradients for diffusion language models, 2026. URL https://arxiv.org/abs/2607.15200.

Kevin Rojas, Jiahe Lin, Kashif Rasul, Anderson Schneider, Yuriy Nevmyvaka, Molei Tao, and Wei Deng. Improving reasoning for diffusion language models via group diffusion policy optimization. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 1938–1973, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/0428b72cab14bc38db5b2bad183e1987-Paper-Conference.pdf.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 130136–130184. Curran Associates, Inc., 2024. doi: 10.52202/079017-4135. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/eb0b13cc515724ab8015bc978fdde0ad-Paper-Conference.pdf.

John Schulman. Approximating kl divergence. http://joschu.net/blog/kl-approx.html, March 2020. Accessed on September 26, 2026.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 103131–103167. Curran Associates, Inc., 2024. doi: 10.52202/079017-3277. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/bad233b9849f019aead5e5cc60cef70f-Paper-Conference.pdf.

Yuxuan Song, Zheng Zhang, Cheng Luo, Pengyang Gao, Fan Xia, Hao Luo, Zheng Li, Yuehang Yang, Hongli Yu, Xingwei Qu, Yuwei Fu, Jing Su, Ge Zhang, Wenhao Huang, Mingxuan Wang, Lin Yan, Xiaoying Jia, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Yonghui Wu, and Hao Zhou. Seed diffusion: A large-scale diffusion language model with high-speed inference, 2025. URL https://arxiv.org/abs/2508.02193.

Hongze Tan, Zihan Wang, Jianfei Pan, Jinghao Lin, Hao Wang, Yifan Wu, Tao Chen, Zhihang Zheng, Zhihao Tang, and Haihua Yang. Gtpo and grpo-s: Token and sequence-level reward shaping with policy entropy, 2026. URL https://arxiv.org/abs/2508.04349.

Xiaohang Tang, Rares Dolga, Sangwoong Yoon, and Ilija Bogunovic. wd1: Weighted policy optimization for reasoning in diffusion language models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 150357–150385, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/f381114cf5aba4e45552869863deaaa7-Paper-Conference.pdf.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback, 2022. URL https://arxiv.org/abs/2211.14275.

Chenyu Wang, Paria Rashidinejad, Andy (DiJia) Su, Song Jiang, Sid Wang, Siyan Zhao, Cai Zhou, Shannon Shen, Feiyu Chen, Tommi Jaakkola, Yuandong Tian, and Bo Liu. Spg: Sandwiched policy gradient for masked diffusion language models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 24623–24660, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/2a2de7a4dcda5c6aac2db8cd771e6a47-Paper-Conference.pdf.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9426–9439, 2024.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xiong-Hui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 115452–115486. Curran Associates, Inc., 2025a. doi: 10.52202/085713-3850. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/a797c2d2e0c1fdabf4d1ab8cd0b465c6-Paper-Conference.pdf.

Yinjie Wang, Ling Yang, Bowen Li, Ye Tian, Ke Shen, and Mengdi Wang. Revolutionizing reinforcement learning framework for diffusion large language models, 2025b. URL https://arxiv.org/abs/2509.06949.

Chengyue Wu, Hao Zhang, Shuchen Xue, Shizhe Diao, Yonggan Fu, Zhijian Liu, Pavlo Molchanov, Ping Luo, Song Han, and Enze Xie. Fast-dllm v2: Efficient block-diffusion llm. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 128353–128370, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/d0865cbe51d35ec322f9af9db7806fc7-Paper-Conference.pdf.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dllm: Training-free acceleration of diffusion llm by enabling kv cache and parallel decoding. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 57027–57051, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/5d8d4e6061c3ba96c240b7fa1ae3471d-Paper-Conference.pdf.

Shaoan Xie, Lingjing Kong, Xiangchen Song, Xinshuai Dong, Guangyi Chen, Eric P. Xing, and Kun Zhang. Advancing reasoning in diffusion language models with denoising process rewards. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 42703–42720, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.1978. URL https://aclanthology.org/2026.acl-long.1978/.

Zhihui Xie, Jiacheng Ye, Lin Zheng, Jiahui Gao, Jingwei Dong, Zirui Wu, Xueliang Zhao, Shansan Gong, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream-coder 7b: An open diffusion language model for code, 2025. URL https://arxiv.org/abs/2509.01142.

Huimin Xu, Shuai Zhao, Xiaobao Wu, and Anh Tuan Luu. Understanding and preventing entropy collapse in rlvr with on-policy entropy flow optimization. *arXiv preprint arXiv:2605.11491*, 2026.

Jingyi Yang, Guanxu Chen, Xuhao Hu, and Jing Shao. Taming masked diffusion language models via consistency trajectory reinforcement learning with fewer decoding step, 2025. URL https://arxiv.org/abs/2509.23924.

Jiacheng Ye, Shansan Gong, Liheng Chen, Lin Zheng, Jiahui Gao, Han Shi, Chuan Wu, Xin Jiang, Zhenguo Li, Wei Bi, and Lingpeng Kong. Diffusion of thought: Chain-of-thought reasoning in diffusion language models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 105345–105374. Curran Associates, Inc., 2024. doi: 10.52202/079017-3343. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/be30024e7fa2c29cac7a6dafcbb8571f-Paper-Conference.pdf.

Jiacheng Ye, Jiahui Gao, Shansan Gong, Lin Zheng, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Beyond autoregression: Discrete diffusion for complex reasoning and planning. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 77875–77898, 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/c1b3d1e2cf53bb28cabd801bd58b3521-Paper-Conference.pdf.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models, 2025b. URL https://arxiv.org/abs/2508.15487.

Runpeng Yu, Qi Li, and Xinchao Wang. Discrete diffusion in large language and multimodal models: A survey, 2025. URL https://arxiv.org/abs/2506.13759.

Oussama Zekri and Nicolas Boulle. Fine-tuning discrete diffusion models with policy gradient methods. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 152868–152906. Curran Associates, Inc., 2025. doi: 10.52202/085713-5112. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/e053a2ffa2a7f5156e2b9a852fde6d45-Paper-Conference.pdf.

Kaiyi Zhang, Wei Wu, and Yankai Lin. Delta: Discriminative token credit assignment for reinforcement learning from verifiable rewards, 2026. URL https://arxiv.org/abs/2605.21467.

Hanyang Zhao, Dawen Liang, Wenpin Tang, David Yao, and Nathan Kallus. Diffpo: Training diffusion llms to reason fast and furious via reinforcement learning, 2026. URL https://arxiv.org/abs/2510.02212.

Siyan Zhao, Devaansh Gupta, Qinqing Zheng, and Aditya Grover. d1: Scaling reasoning in diffusion large language models via reinforcement learning. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 56729–56762. Curran Associates, Inc., 2025. doi: 10.52202/085713-1898. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/52190a0362148d179f1cbd9080956872-Paper-Conference.pdf.

Jianyuan Zhong, Kaibo Wang, Ding Ding, Zijin Feng, Haoli Bai, Yang Xiang, Jiacheng Sun, and Qiang Xu. Stabilizing reinforcement learning for diffusion language models, 2026. URL https://arxiv.org/abs/2603.06743.

Fengqi Zhu, Rongzhen Wang, Shen Nie, Xiaolu Zhang, Chunwei Wu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. LLaDA 1.5: Variance-reduced preference optimization for large language diffusion models. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11425–11460, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.524. URL https://aclanthology.org/2026.acl-long.524/.

A Related Work

A.1 Diffusion Language Models

Building on the denoising paradigm of diffusion models (Ho et al., 2020), diffusion language models (DLMs) generate text through iterative denoising (Hoogeboom et al., 2021; Dieleman et al., 2022; Gulrajani & Hashimoto, 2023) rather than autoregressive next-token prediction (Lou et al., 2024; Li et al., 2022; Ye et al., 2024). Early discrete diffusion language models explore masked generation objectives and iterative refinement for language modeling (Austin et al., 2021; Sahoo et al., 2024; Shi et al., 2024). LLaDA scales this paradigm to large language models with a masked diffusion objective and demonstrates competitive language generation performance (Nie et al., 2025). Subsequent works further study improved training objectives, generation strategies (Wu et al., 2026a; Xie et al., 2025), scaling behaviors (Ye et al., 2025b; Gong et al., 2026; Wu et al., 2026b), and data efficiency under limited-data and multi-epoch training settings (Prabhudesai et al., 2025; Ni et al., 2025; Gao et al., 2025). When applying reinforcement learning to these models, their iterative generation process requires particular consideration. Unlike autoregressive models, DLMs maintain multiple intermediate denoising states, where tokens are progressively committed and remaining masked positions are refined. This generation process introduces new challenges and opportunities for reinforcement learning in diffusion models.

A.2 Reinforcement Learning for Diffusion Language Models

Reinforcement learning for diffusion language models requires optimization methods that account for the iterative denoising process (Raajesh et al., 2026; He et al., 2025). Existing approaches mainly adapt RL objectives (Schulman et al., 2017; Shao et al., 2024; Zekri & Boulle, 2025; Ni et al., 2026) or exploit intermediate denoising information (Xie et al., 2026) to construct effective learning signals (Tang et al., 2026; Rojas et al., 2026; Zhu et al., 2026). d1 introduces diffu-GRPO (Zhao et al., 2025), Coupled-GRPO reduces likelihood estimation variance through complementary masks (Gong et al., 2026), and ESPO uses evidence lower bound as a proxy for sequence likelihood during RL optimization (Ou et al., 2026). Other methods leverage denoising trajectories (Yang et al., 2025) and intermediate states to improve RL training, including TraceRL, RLDF, and DiSPO (Wang et al., 2025b; He et al., 2026; Oba et al., 2026). These methods demonstrate how diffusion generation characteristics can be incorporated into RL optimization (Zhao et al., 2026). However, they mainly focus on optimization objectives (Zhong et al., 2026; Jiang et al., 2026), likelihood estimation (Wang et al., 2026), or denoising trajectories, while how to assign fine-grained credit to individual tokens during the denoising process remains underexplored.

A.3 Fine-grained Credit Assignment

Sequence-level rewards in language model reinforcement learning (Ouyang et al., 2022; Guo et al., 2025) provide limited information about which tokens or reasoning steps contribute to the final outcome, motivating recent studies on fine-grained credit assignment (Wang et al., 2025a; Cheng et al., 2026). Existing approaches improve reward granularity through process supervision (Uesato et al., 2022), step-level feedback, and token-level reward redistribution (Tan et al., 2026) or importance estimation (Lightman et al., 2024; Wang et al., 2024; Li et al., 2025; Cui et al., 2025; Kazemnejad et al., 2025; Li et al., 2026; Zhang et al., 2026). However, these methods are mainly developed for autoregressive language models, where tokens are generated sequentially. For diffusion language models, DACA-GRPO provides an initial exploration of token-level credit assignment by weighting committed tokens according to denoising progress and changes in the predicted probabilities of remaining masked positions (Monsefi et al., 2026). Different from DACA-GRPO, DTCA estimates token contributions from confidence improvements after token commitment, aggregates these signals according to POS categories, and uses the resulting contribution structure for denoising timestep selection and token-level credit assignment.

B Method Supplement

Notation and boundary cases. In Eq. 7, $\mathcal{A}_t$ contains the positions committed at denoising step t and is used for timestep selection. The optimization targets $\mathcal{B}_t$ may differ and include both commit-

ted positions and retained masked positions selected by threshold-based filtering. POS categories for all optimization targets are obtained from the corresponding tokens in the completed rollout. In Eq. 10, η denotes the policy update step size used in the first-order entropy change approximation, and S_b^+ and S_b^- collect targets with positive and negative estimated entropy changes in response b.

All 18 POS categories have entries in the contribution table. A POS tag outside this inventory, or a model token that cannot be aligned to a POS span, is assigned to the X category and uses its computed weight w_{X}. If all category means coincide, we set $w_c = 1$ for every category. If the total magnitude of estimated entropy change is zero, we set $\lambda_b = 0$. A zero response advantage yields zero token credits without applying Eq. 14. We exclude empty commitment sets from offline statistics and states with no optimization targets from training.

Algorithm 1 One RL Update with Denoising Token Credit Assignment (DTCA)

Input: Policy π_θ, reference π_{ref}, prompts $\mathcal{X}$, optimizer O
  Fixed offline POS weights $\{w_c\}$, threshold μ
$\tau \leftarrow \pi_\theta(\mathcal{X})$ ▷ Generate denoising rollouts
Compute rewards, advantages $\{A^{(b)}\}$, and token POS categories
for each response b in τ **do**
  for each denoising timestep t in response b **do**
    $q_t^{(b)} \leftarrow \sum_{i \in \mathcal{A}_t^{(b)}} w_{c_i}(1 - p_i)$
  end for
  Select the highest-scoring steps covering mass μ; apply count bounds to obtain $\mathcal{T}^{(b)}$
  for each selected timestep $t \in \mathcal{T}^{(b)}$ **do**
    Reconstruct $x_t^{(b)}$ and form the retained targets $\mathcal{B}_t^{(b)}$
    $a_{t,i}^{(b)} \leftarrow w_{c_i} / \mathrm{Mean}_{k \in \mathcal{B}_t^{(b)}}(w_{c_k})$ for $i \in \mathcal{B}_t^{(b)}$
    $A_{t,i}^{(b)} \leftarrow a_{t,i}^{(b)} A^{(b)}$
    Estimate $\Delta H_{t,i}^{(b)}$ from $A_{t,i}^{(b)}$, current target probabilities, and entropies
  end for
end for
for each response b in τ **do**
  Form S_b^+ and S_b^-; compute λ_b and clip it to $[-0.95, 0.95]$
  Scale $A_{t,i}^{(b)}$ by $1 + \lambda_b$ if $\Delta H_{t,i}^{(b)} > 0$, otherwise by $1 - \lambda_b$, to obtain $\hat{A}_{t,i}^{(b)}$
  Normalize $\hat{A}_{t,i}^{(b)}$ to obtain $\widetilde{A}_{t,i}^{(b)}$ ▷ Preserve total response credit
end for
$\theta \leftarrow O(\theta, \nabla_\theta \mathcal{L}_{\mathrm{DTCA}})$

Training objective. Let $A^{(b)}$ be response b's group-normalized advantage, used as A_t in Eq. 9 at each selected step $t \in \mathcal{T}^{(b)}$. The targets $\mathcal{B}_t^{(b)}$ include masked positions with predicted token probability at least 0.2 and positions committed at step t. Since the completed rollout is available after generation, POS categories can be assigned to all targets in $\mathcal{B}_t^{(b)}$. Eqs. 8–13 yield $\hat{A}_{t,i}^{(b)}$ for each target. We restore the response's total credit:

$$\widetilde{A}_{t,i}^{(b)} = \hat{A}_{t,i}^{(b)} \frac{\sum_{t' \in \mathcal{T}^{(b)}} \sum_{j \in \mathcal{B}_{t'}^{(b)}} A_{t',j}^{(b)}}{\sum_{t' \in \mathcal{T}^{(b)}} \sum_{j \in \mathcal{B}_{t'}^{(b)}} \hat{A}_{t',j}^{(b)}}. \tag{14}$$

This response-wide factor restores the overall credit scale after entropy calibration, while clipping λ_b may leave a residual entropy imbalance. The update loss is

$$\mathcal{L}_{\mathrm{DTCA}}(\theta) = \frac{1}{N} \sum_b \sum_{t \in \mathcal{T}^{(b)}} \sum_{i \in \mathcal{B}_t^{(b)}} \left[-\widetilde{A}_{t,i}^{(b)} \log \pi_\theta(v_i^{(b)} \mid q^{(b)}, x_t^{(b)}) + \beta_{\mathrm{KL}} \left(e^{d_{t,i}^{(b)}} - 1 - d_{t,i}^{(b)} \right) \right], \tag{15}$$

where $N = \sum_b \sum_{t \in \mathcal{T}^{(b)}} |\mathcal{B}_t^{(b)}|$ and $d_{t,i}^{(b)} = \log \pi_{\mathrm{ref}}(v_i^{(b)} \mid q^{(b)}, x_t^{(b)}) - \log \pi_\theta(v_i^{(b)} \mid q^{(b)}, x_t^{(b)})$. The reference policy is frozen. With one policy iteration, PPO ratio clipping is inactive. Entropy calibration modifies $\widetilde{A}_{t,i}^{(b)}$ rather than adding an entropy loss.

C Experimental Setup and Implementation Details

C.1 Base Models and Datasets

The policy and frozen reference model both start from LLaDA-8B-Instruct, without an intermediate SFT stage. RL training uses the MATH training set processed for RLDF. Offline POS contribution estimates are constructed from denoising rollouts over the MATH and GSM8K training splits without parameter updates to build a shared fixed POS contribution table. We evaluate on MATH500 and the GSM8K test split; neither evaluation split is used for RL training or POS table construction.

C.2 Training Configuration

We reproduce the compared methods under a unified experimental setup wherever applicable. RL methods start from LLaDA-8B-Instruct and use the same RLDF-processed MATH training split and mathematical correctness reward. All methods are evaluated using the common protocol in Appendix C.4. Each method retains its own objective and method-specific settings: we follow its original paper first, and use its released open-source code for details not specified in the paper.

For DTCA and the locally reproduced RLDF baseline, we use full-parameter bfloat16 training with AdamW, a constant learning rate of 10^{-6}, and zero weight decay. Each prompt produces four responses, whose rewards are standardized within the group to form response advantages. Both methods use the same low-probability threshold of 0.2 for training-target filtering, token-level loss normalization, and a frozen-reference k_3 KL penalty with coefficient 0.01.

Their training rollouts use a maximum response length of 256 tokens and 256 denoising steps. Decoding uses a block size of 32 and a dynamic confidence threshold of 0.9; when no position crosses the threshold, the highest-confidence position is committed. This decoding threshold is distinct from the 0.2 training-target filter. KV caching is used during rollout generation.

C.3 Implementation Details of DTCA

We tag completions with spaCy's `en_core_web_sm` POS model and align its word spans with model token positions. The 18 native POS categories share one contribution table. Before RL training, we map the offline category means to $[0.8, 1.2]$ using Eq. 6. These weights are used for both timestep selection and token credit assignment and remain fixed throughout RL training.

DTCA changes the timestep-selection and credit-assignment rules relative to the locally reproduced RLDF baseline. For state selection, we rank observed denoising steps by q_t and cover at least 70% of the total priority mass, selecting between 2 and 32 steps. At each selected step, training targets are the remaining masked positions retained by the shared low-probability filter, plus the positions committed at that step even if they fall below the filter threshold. POS credits are normalized to preserve the mean response advantage within each selected step. Entropy-flow calibration limits the absolute balancing coefficient to 0.95 and restores each response's total credit after rescaling.

C.4 Evaluation Settings

We evaluate each checkpoint at maximum response lengths of 128, 256, and 512 tokens, using the same number of denoising steps as the response length. For every setting, we use block size 32, dynamic confidence threshold 0.9, temperature 0, and one generated response per question. The same math prompt template and decoding settings are used for DTCA and all locally reproduced baselines.

We evaluate only whether the final answer is correct, without scoring the intermediate reasoning steps. Accuracy is the fraction of questions with correct final answers on the complete test split. All baseline scores in Table 1 are obtained from our local reproductions under this evaluation protocol.

D Additional Experiments and Analysis

D.1 Additional Ablation Studies

Contribution-based timestep selection. Table 2 compares selection rules on a shared dataset using our fixed POS weights in $[0.8, 1.2]$. DTCA uses $\mu \in \{0.40, 0.70\}$ and selects 2–32 states. RLDF samples without replacement from a temperature-1 softmax of state mean token surprisal, using 16 or 32 states or DTCA's per-response count. RLDF top-k takes the k highest-surprisal states, or all available states if fewer than k. Coverage is the fraction of committed tokens, confidence $\sum_i p_i$, low-confidence mass $\sum_i (1 - p_i)$, or entropy $\sum_i H_i$ in selected states. Ratios are averaged over responses. Sampled RLDF uses 2,000 independent selections.

Table 2: Coverage of selected denoising states. State ratio is the selected fraction per response. RLDF (same count) matches DTCA's count at each μ. All entries are mean percentages.

Method	State ratio	Token cov.	Conf. cov.	Low conf. cov.	Entropy cov.
DTCA ($\mu = 0.40$)	16.82	19.99	17.72	40.59	37.81
RLDF $k = 16$	23.31	21.33	20.79	26.52	25.56
RLDF (same count, $\mu = 0.40$)	16.82	15.01	14.34	19.93	19.22
RLDF top-16	23.31	10.01	7.33	42.46	35.88
DTCA ($\mu = 0.70$)	31.81	36.02	33.24	62.81	59.31
RLDF $k = 32$	45.42	42.61	41.73	49.91	48.73
RLDF (same count, $\mu = 0.70$)	31.81	28.94	28.06	36.46	35.21
RLDF top-32	45.42	28.75	24.99	66.39	60.66

At a matched state ratio of 16.82%, DTCA ($\mu = 0.40$) increases low-confidence coverage from 19.93% to 40.59% and entropy coverage from 19.22% to 37.81% relative to RLDF, while retaining broader token coverage. RLDF top-16 achieves slightly higher low-confidence coverage (42.46%) but covers fewer tokens (10.01% versus 19.99%).

At $\mu = 0.70$ and a matched state ratio of 31.81%, DTCA increases low-confidence coverage from 36.46% to 62.81% and entropy coverage from 35.21% to 59.31%. It also achieves higher coverage than RLDF $k = 32$ while selecting fewer states. RLDF top-32 obtains comparable uncertainty coverage but selects more states and covers fewer tokens. These results show that DTCA selects denoising states with different uncertainty distributions, while the impact of state selection on final performance is evaluated through the main results and ablations.

D.2 Analysis of Denoising Dynamics

Policy entropy changes across POS categories. Figure 6 and the upper panel of Figure 7 compare current-commit token entropy by POS. Over updates 401–500, NUM entropy is 0.063 nats below its early baseline in DTCA but 0.208 nats above it without calibration. Late positive shifts in the uncalibrated run also appear for PROPN, X, and SPACE, rather than being confined to NUM.

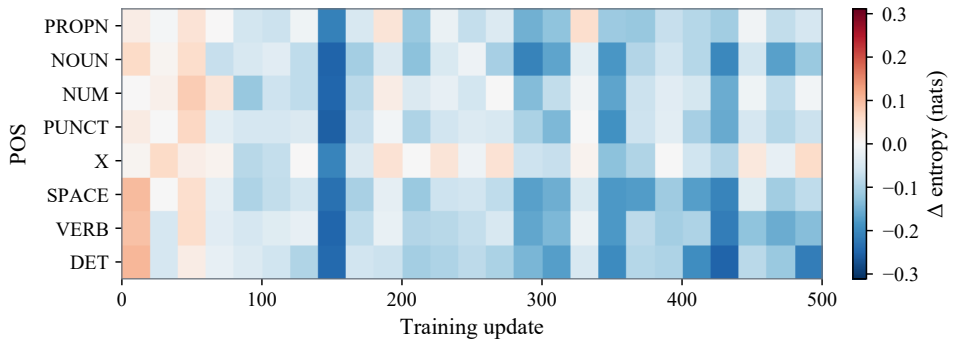
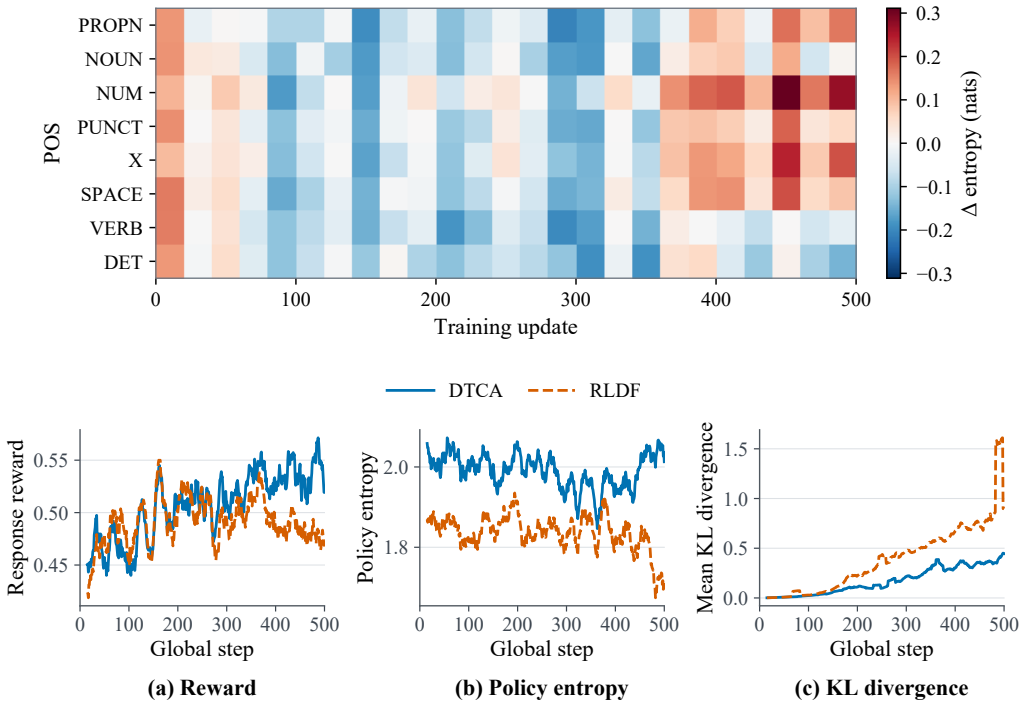


Figure 6: Current-commit token entropy changes in DTCA. Each 20-update bin subtracts the POS-specific mean over updates 1–100. The color scale matches the upper panel of Figure 7.

D.3 Additional Results

Training dynamics against RLDF. The lower panels of Figure 7 compare the training dynamics of DTCA and RLDF over 500 updates. The figure reports response reward, policy entropy, and mean KL divergence, with curves smoothed by a trailing 15-update mean. A higher response reward indicates that more sampled responses are correct.



(a) Reward **(b) Policy entropy** **(c) KL divergence**

Figure 7: Additional entropy and training-dynamics analyses. Top: current-commit token entropy changes in DTCA without entropy calibration, using the same early-training baseline, 20-update bins, and color scale as Figure 6. Bottom: DTCA versus RLDF in (a) response reward, (b) policy entropy, and (c) mean KL divergence, using trailing 15-update means.

The two methods start with similar reward trajectories, with average rewards of 0.464 for DTCA and 0.467 for RLDF over updates 1–100. Over updates 401–500, DTCA achieves a higher average reward of 0.537, compared with 0.482 for RLDF. During the same period, DTCA maintains higher policy entropy (2.00 vs. 1.78 nats) and lower mean KL divergence (0.36 vs. 0.89). These results show that DTCA and RLDF follow different optimization trajectories, with DTCA achieving higher reward while maintaining a smaller deviation from the reference policy.

The observed differences are consistent with DTCA's design, where contribution-based credit assignment provides token-dependent learning signals and entropy-based calibration adjusts updates according to entropy changes. This comparison reflects the combined effect of DTCA components and does not isolate individual module contributions.

Full POS entropy analysis. Figure 8 shows the nine POS categories most often retained as training targets in DTCA. Together with Figure 9, it extends Figure 5 to all 18 POS categories. These curves cover all retained targets at selected states, rather than only current-commit tokens. With run-specific early baselines, positive values indicate entropy increases, and curves compare changes rather than absolute levels.

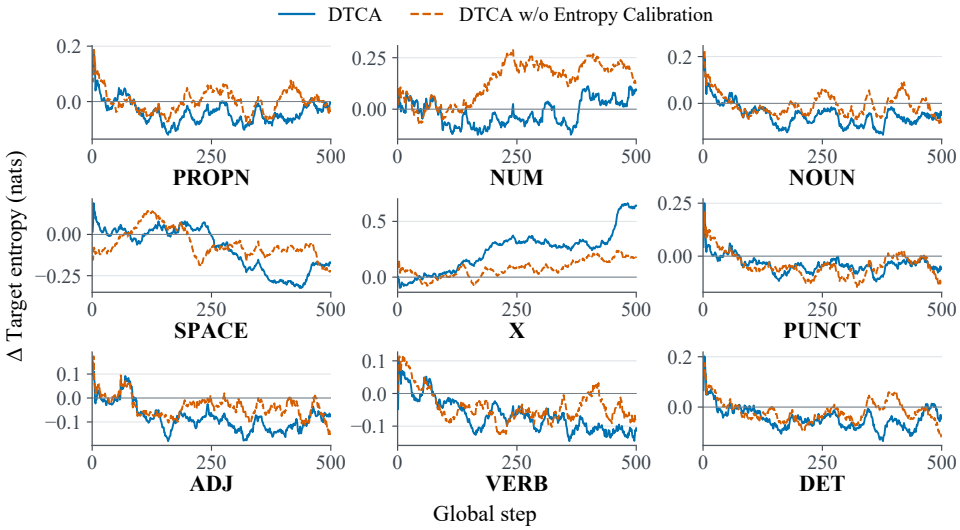


Figure 8: Predictive entropy changes for DTCA's nine most frequent target POS categories. Blue solid: DTCA. Orange dashed: without entropy calibration. Curves show trailing 25-update token-weighted means relative to each run's POS-specific baseline over updates 1–100.

The frequent categories show different trends. Over updates 401–500, NUM is 0.051 nats above its early baseline in DTCA, compared with 0.209 nats without calibration. For X, the shifts are 0.447 and 0.184 nats. SPACE, ADJ, and VERB generally move below their early baselines in both runs. Calibration does not uniformly lower entropy across POS.

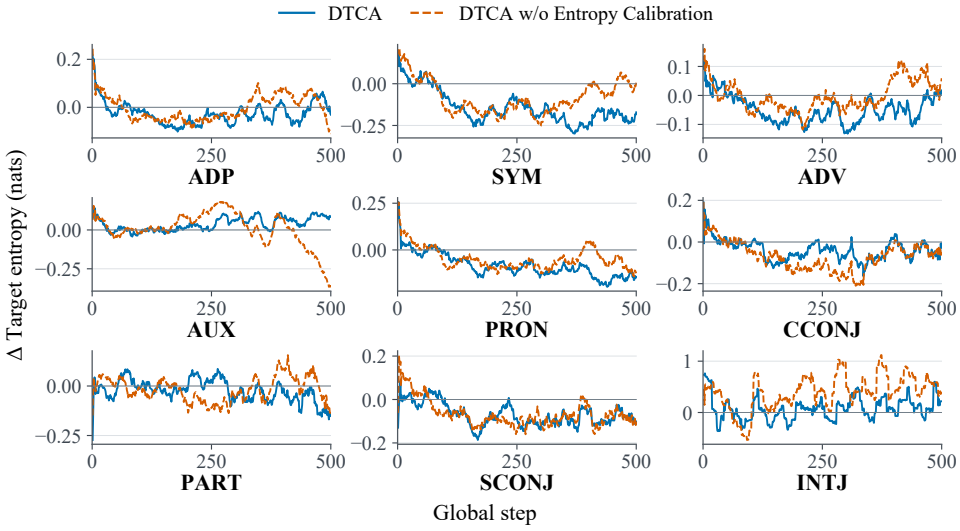


Figure 9: Predictive entropy changes for the nine remaining POS categories, ordered by their frequency among DTCA's retained targets. Colors, line styles, baseline, and smoothing match Figure 8.

The remaining categories also differ. Over updates 401–500, SYM falls by 0.192 nats from its early baseline in DTCA and by 0.014 nats without calibration. AUX instead rises by 0.072 nats in DTCA and falls by 0.133 nats without calibration. INTJ fluctuates sharply, warranting caution in interpreting its trend.

Entropy-flow calibration. Figures 8 and 9 summarize POS-level entropy trajectories over training. These are within-run trends rather than changes in matched tokens, since the models visit different states and retain different targets. Figure 10 complements these trajectories by examining the predicted entropy changes induced by individual policy updates, separated into entropy-increasing and entropy-decreasing parts.

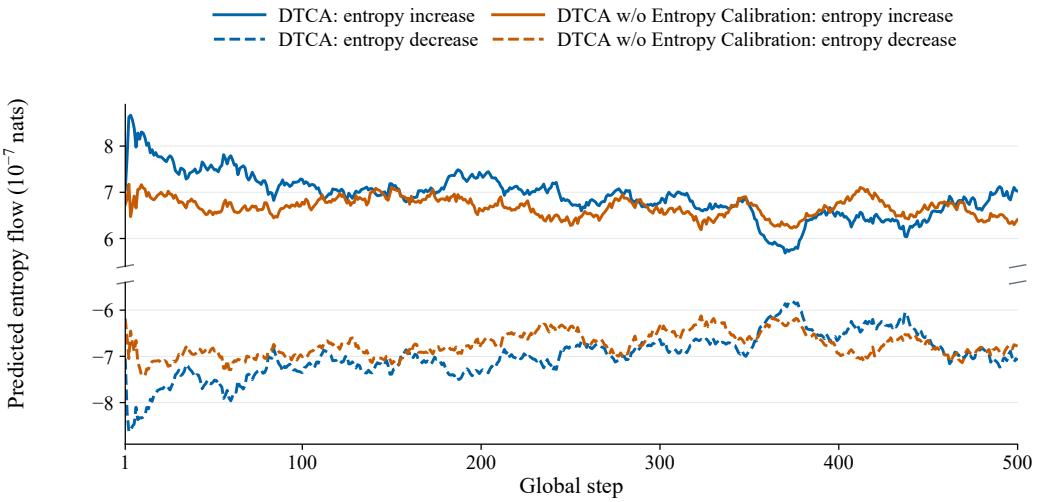


Figure 10: Predicted entropy-increasing (solid) and entropy-decreasing (dashed) components for DTCA (blue) and DTCA without entropy calibration (orange). Curves use trailing 25-update means. The vertical axis is broken, with values in 10^{-7} nats.

Solid curves show the total predicted increase, while dashed curves show the total predicted decrease. The broken vertical axis keeps both signs visible despite their similar magnitudes. Equal magnitudes imply a near-zero net predicted change.

DTCA keeps the two parts closer together. Over updates 1–100, its average net predicted change is -0.033×10^{-7} nats, compared with -0.267×10^{-7} nats without calibration. Over updates 401–500, the corresponding values are -0.081×10^{-7} and -0.201×10^{-7} nats. Thus, the predicted increase and decrease nearly cancel under DTCA in both windows. These values are first-order estimates based on the training signals, not direct measurements of entropy after a parameter update.

E LIMITATIONS

While DTCA provides an effective approach for fine-grained credit assignment in diffusion language model reinforcement learning, several aspects remain open for future exploration. DTCA uses POS categories to aggregate token-level confidence improvements and obtain stable contribution estimates. This design relies on the quality of POS alignment and annotation, and more accurate token-level labeling strategies may further improve the estimation process. Moreover, DTCA currently defines token contribution based on the confidence improvement of remaining masked positions after token commitment. Developing more principled theoretical formulations for contribution estimation in diffusion language models remains an interesting direction. Finally, our experiments focus on mathematical reasoning with LLaDA-8B-Instruct. Applying DTCA to other tasks, such as code generation, may require adapting the token grouping strategy to different token distributions and generation patterns. Further evaluation on diverse diffusion language model architectures is also needed to understand the general applicability of DTCA.