Global Context Guidance for Diffusion Large Language Models

Jeimin Jeon[1,2] **Bumsub Ham**[1*]

[1]Yonsei University [2]Articron Inc.

https://cvlab.yonsei.ac.kr/projects/GCG/

Abstract

Diffusion large language models (LLMs) have recently emerged as a promising alternative to autoregressive LLMs, enabling parallel generation through iterative denoising. Recent guidance methods improve diffusion LLMs by introducing a weak model to guide the prediction of the base model during denoising. However, the influence of the weak model during denoising remains less clear, and computing its prediction requires an additional forward pass at each denoising step. We present in this paper an empirical analysis of how diffusion LLMs use contextual information during denoising. Our analysis reveals that diffusion LLMs rely more heavily on local context while often underutilizing global context. Based on this observation, we introduce Global Context Guidance (GCG), a feature-space guidance method that enhances global-context utilization without constructing an explicit weak model. Extensive experiments on LLaDA-8B, Dream-7B, and LLaDA-1.5 across diverse benchmarks demonstrate that GCG consistently outperforms existing guidance methods, while incurring much lower inference overhead.

1 Introduction

Autoregressive large language models (LLMs) [1, 2, 3] have become a dominant paradigm for language generation, achieving remarkable performance across a wide range of tasks through next-token prediction. By generating text in a left-to-right manner, they model linguistic structure effectively, showing strong in-context learning and reasoning capabilities. However, their sequential decoding process inherently limits parallelization and makes each prediction depend only on previously generated tokens. This limitation has led to growing interest in alternative frameworks that can model and refine tokens more globally during inference.

Diffusion LLMs [4, 5, 6] have recently emerged as a promising alternative to autoregressive LLMs. Instead of generating text sequentially from left to right, they begin with a masked sequence and generate text with an iterative denoising process, progressively unmasking tokens over multiple steps. This enables parallel generation and allows the model to leverage bidirectional context during denoising. Recent studies [7, 8, 9] have increasingly focused on improving the performance of diffusion LLMs. One promising direction is to incorporate guidance during denoising. Motivated by the success of guidance in visual diffusion models [10], recent works have incorporated guidance strategies for diffusion LLMs. These approaches typically introduce a weak model and adjust the prediction of the base model, using the difference between the outputs of the base and weak models. They mainly differ in how the weak model is constructed. For example, U-CFG [11] replaces the prompt with masked tokens to construct a weak model, while A-CFG [12] forms a weak model by remasking unconfident tokens in the output of the base model. Although these approaches show that

*Corresponding author

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

guidance can improve diffusion LLMs, they have two key limitations. First, the influence of the weak model on the denoising process remains less clear. In diffusion LLMs, masked tokens are prediction targets rather than null conditions as in visual diffusion [13, 10], leaving the role of the weak model unclear. As a result, guidance may steer denoising in undesirable directions, potentially degrading the performance. Second, these methods incur substantial inference overhead. They use predictions of both the base and weak models, requiring two forward passes at each denoising step. This doubles the inference cost and weakens the faster inference enabled by parallel denoising in diffusion LLMs.

To address these limitations, we propose Global Context Guidance (GCG), a feature-space guidance method for diffusion LLMs that enhances the utilization of contextual information during denoising. In diffusion LLMs, contextual information plays an important role in parallel denoising, as bidirectional attention allows the model to gather contextual cues from different positions across the sequence [6, 14]. To better understand this behavior, we first investigate how diffusion LLMs use contextual information during denoising. Specifically, we construct a synthetic infilling dataset that controls the distance between masked tokens and their supporting evidence in bidirectional documents, ranging from nearby evidence (*i.e.*, local context) to distant evidence (*i.e.*, global context). Through empirical analysis, we find that diffusion LLMs rely more heavily on local context while often underutilizing global context, suggesting that diffusion LLMs do not fully exploit the entire context available during denoising. We further observe that a subset of attention heads is particularly important for leveraging global context. Based on this observation, GCG identifies such heads, which we refer to as global heads, and strengthens their outputs during inference. Specifically, GCG projects outputs of global heads onto a subspace formed by local heads (*i.e.*, local space) and amplifies the residual between the global-head outputs and their projections, thereby increasing the contribution of global context. Since this operation is performed directly in the feature space within a single forward pass, GCG remains efficient without requiring an additional weak model. We evaluate GCG on various models and tasks, and show that it consistently outperforms existing guidance methods, while incurring much lower inference overhead. We summarize our contributions as follows:

- We present an analysis of how diffusion LLMs exploit contextual information by constructing a synthetic infilling dataset. Our analysis reveals that diffusion LLMs are biased toward local context during denoising, and a subset of attention heads plays a critical role in leveraging global context.
- Based on the analysis, we propose GCG, an effective and efficient feature-space guidance method. GCG projects the outputs of selected global heads onto a local space, and amplifies the difference between the outputs of global heads and their projections, strengthening global-context features without requiring an additional weak model.
- Extensive experiments on various models and tasks show that GCG is both more effective and efficient than existing guidance methods.

2 Related Work

In this section, we review representative works on diffusion LLMs and guidance methods.

2.1 Diffusion LLMs

Diffusion LLMs [4, 15, 11] have emerged as an alternative paradigm to autoregressive LLMs [1, 2, 3], generating text through iterative denoising rather than left-to-right next-token prediction. This paradigm allows parallel generation and greater flexibility in generation order, offering a promising framework for language generation. Recent large-scale models such as LLaDA [5] and Dream [6] demonstrate that diffusion LLMs can achieve performance competitive with strong autoregressive baselines across a wide range of tasks, including general instruction following, mathematical reasoning, and code generation [16, 17, 18].

Recent studies have explored training-free methods for improving the efficiency and effectiveness of diffusion LLMs. One line of work focuses on improving inference efficiency through KV caching. Since diffusion LLMs repeatedly compute bidirectional attention over the full sequence during denoising, standard KV caching used in autoregressive LLMs is not directly applicable. Several works [9, 7] address this limitation based on the observation that the key-values of prompt and unmasked tokens change only slightly between adjacent denoising steps. They thus propose to cache

the states of these tokens and reuse them across steps, reducing the computational cost. Another line of work adapts the denoising process to use the inference budget more effectively. For example, Fast-dLLM [7] dynamically controls the number of tokens updated at each step, while DAEDAL [8] adjusts the output length during denoising. Recent works [19, 20] have also revisited the irreversible nature of standard diffusion LLMs, where tokens typically remain fixed once they are unmasked. To alleviate this limitation, these methods remask uncertain tokens and refine them again in later steps, which mitigates error propagation from incorrect early predictions.

2.2 Guidance

Guidance has been widely studied in diffusion models, particularly for image generation. A representative approach is classifier guidance [13], which steers the denoising process using gradients from an external classifier so that generated samples better satisfy a target condition. Although effective, this approach requires a separately trained classifier and increases inference complexity. To address this limitation, classifier-free guidance (CFG) [10] has been proposed, which combines conditional and unconditional predictions from the diffusion model itself. CFG improves condition alignment and sample quality without requiring an additional classifier, and has become one of the most widely used guidance methods in image diffusion models. Autoguidance [21] further extends this framework, introducing a weak model instead of the unconditional model, which is constructed by training with fewer iterations than the base model. It then uses the difference between the outputs of base and weak models as a guidance signal, and finds that this strengthens components that are suppressed in the weak model. Subsequent works have explored various ways to construct the weak model, such as perturbing the attention maps [22] or stochastically dropping network blocks [23], suggesting that the design of the weak model is crucial for effective guidance.

Guidance has recently been extended from image diffusion models to diffusion LLMs. In image generation, CFG constructs an unconditional model by replacing the condition with a null prompt, and guides generation using the difference between conditional and unconditional outputs. U-CFG [11] adapts this idea to diffusion LLMs, constructing a weak model, which replaces the prompt tokens with mask tokens. A-CFG [12] further replaces this static weak model with a dynamic weak model, constructed by re-masking unconfident tokens in the output of the base model at each denoising step. To our knowledge, these are the only guidance methods for diffusion LLMs, showing that guidance can also improve diffusion LLMs. They, however, require an additional forward pass to obtain predictions from the weak model at each denoising step. GCG instead performs guidance directly in the feature space without constructing a weak model, remaining efficient within a single forward pass.

3 Method

In this section, we review the formulation of conventional diffusion LLMs (Sec. 3.1), analyze how these models use contextual information during denoising (Sec. 3.2), and present our method, GCG (Sec. 3.3).

3.1 Preliminary

Given a prompt sequence $\mathbf{c}$, diffusion LLMs generate a target sequence $\mathbf{x}_0$ through an iterative denoising process. Starting from a fully masked sequence, the model progressively unmasks tokens over multiple steps, where we denote by $\mathbf{x}_\tau$ a partially masked sequence at step τ. At each step, the model takes $\mathbf{x}_\tau$ and $\mathbf{c}$ as input, and predicts the original tokens at the masked positions as follows:

$$p_\theta(\mathbf{x}_0 \mid \mathbf{x}_\tau, \mathbf{c}). \tag{1}$$

Existing guidance methods for diffusion LLMs typically introduce a weak model in addition to the base model. Let $\mathbf{z}_{\text{base}}(i, \tau)$ and $\mathbf{z}_{\text{weak}}(i, \tau)$ denote the logits at token position i and denoising step τ from the base and weak models, respectively. Guidance is then applied at the logit level as

$$\mathbf{z}_{\text{guided}}(i, \tau) = \mathbf{z}_{\text{base}}(i, \tau) + s\left(\mathbf{z}_{\text{base}}(i, \tau) - \mathbf{z}_{\text{weak}}(i, \tau)\right), \tag{2}$$

where s is a guidance scale. This formulation increases the contribution of directions that are suppressed in the weak model, but it requires an additional forward pass to obtain $\mathbf{z}_{\text{weak}}(i, \tau)$.

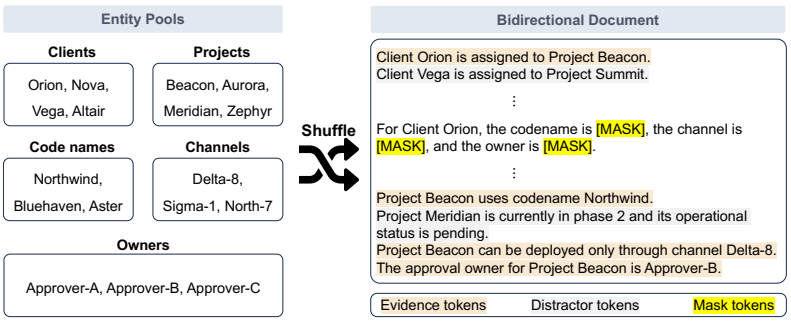


Figure 1: Data generation process for a synthetic infilling benchmark. Entities are randomly sampled from predefined pools and used to construct a bidirectional document. In the resulting document, supporting evidence is placed on both sides of masked tokens, and separated from them by distractor tokens. The distances from the masked tokens to the supporting evidence are varied randomly for each sample.

3.2 Analysis

3.2.1 Contextual-distance analysis

To analyze how diffusion LLMs leverage contextual information during denoising, we construct a synthetic infilling benchmark (Fig. 1). Specifically, we generate each sample through randomly sampling entities from predefined pools and using them to construct a bidirectional document. The resulting document contains masked tokens in the middle and supporting evidence on both sides. By randomly varying the distances from the masked tokens to the left and right evidence, we obtain examples spanning a broad range of local and global-context settings. For more details on the data generation process, please refer to Appendix A.

Using this benchmark, we analyze the effect of evidence–mask distance on the performance of LLaDA-8B [5]. We can see from the gray line of Fig. 2a that the performance degrades consistently as the evidence becomes more distant from the masked tokens. This suggests that, although diffusion LLMs can access the entire context during denoising, they are biased toward local context and underutilize global context.

3.2.2 Head-level analysis

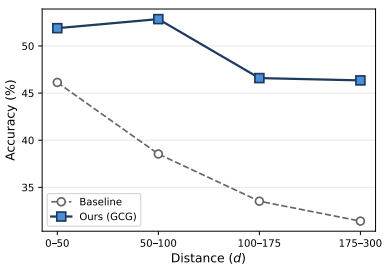
We next analyze attention patterns of individual heads using the center of mass (COM) of their attention distributions. Let $\alpha_h^m(i, j, \tau)$ denote the attention weight from query token i to key token j at block m, head h, and denoising step τ. For each head, we compute the average distance between query and key tokens, where the distance $|i - j|$ is weighted by the corresponding attention weight $\alpha_h^m(i, j, \tau)$:

$$\frac{1}{T}\sum_{\tau=1}^{T}\left(\frac{1}{N}\sum_{i}\sum_{j}\alpha_h^m(i,j,\tau)|i-j|\right), \tag{3}$$

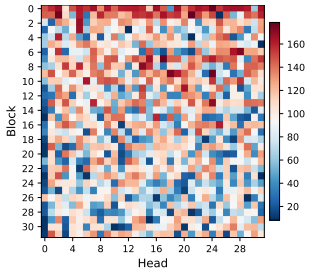
where T is the number of denoising steps and N is the number of query tokens. We refer to this value as the COM of head h in block m. Larger COM scores indicate that the head attends more strongly to distant positions on average, and vice versa.

As shown in Fig. 2b, attention heads exhibit substantial variation in their average COM values. Some heads have relatively small COM values and primarily attend to nearby positions, whereas others have much larger values and attend more strongly to distant positions. To examine whether this tendency is preserved across samples, we compute head rankings induced by COM for each sample and measure their pairwise Spearman correlation. The resulting rankings are highly consistent, with a mean correlation of 0.94 across 700 samples.

We then examine whether heads with high COM values are important for leveraging distant context. For this analysis, we rank heads in each block by their COM values, and refer to the top-25% and bottom-25% heads as global and local heads, respectively. We prune either global or local heads



(a) Accuracy vs. Evidence-Mask Distance.



(b) Average COM of attention heads across blocks.

Figure 2: (a) Accuracy as a function of the average evidence–mask distance. The baseline performance consistently degrades as the distance increases, indicating that diffusion LLMs rely more heavily on local context than on global context during denoising (gray). GCG mitigates this issue by enhancing global-context utilization, yielding larger gains at longer distances (blue). (b) COM heatmap across all blocks and heads. Heads with high COM values (red) attend more strongly to global context, whereas heads with low COM values (blue) mainly focus on local context.

by zeroing the outputs of the corresponding heads during inference, and measure the resulting performance change on the synthetic infilling benchmark. As shown in Fig. 3a, pruning global heads causes substantially larger performance degradation than pruning local heads, and this gap becomes more pronounced as the evidence–mask distance increases. This indicates that global heads play a more important role in leveraging distant context during denoising.

To further analyze whether all global heads are equally important across blocks, we perform a block-wise ablation study, where global heads are removed from one block at a time, while keeping those in all other blocks unchanged. As shown in Fig. 3b, the impact varies substantially across blocks, indicating that the contribution of global heads is not uniform across blocks. Additional analyses on Dream-7B [6] and LLaDA-1.5 [24] are provided in Appendix C, showing similar trends.

3.3 Global Context Guidance (GCG)

Based on the above observations, we propose GCG, a feature-space guidance method for diffusion LLMs that enhances global-context utilization during denoising. GCG identifies target blocks for global-context utilization, using COM analysis and block-wise ablation, and applies guidance to global heads in these blocks. For each selected global head, GCG projects its output onto the subspace spanned by outputs of local heads in the same block. It then amplifies the residual between the outputs of global heads and their projection, thereby strengthening global-context features during denoising.

We identify global and local heads and target blocks for GCG using a training subset of LongBench [25][2]. For each model, we first compute COM values on this subset and rank heads within each block, and designate top-$p\%$ heads as global heads and the bottom-$p\%$ heads as local heads. We then perform block-wise ablation, pruning global heads in each block and select the top-$q\%$ blocks with the largest performance drops as target blocks. The selected heads and target blocks are fixed for all evaluation benchmarks.

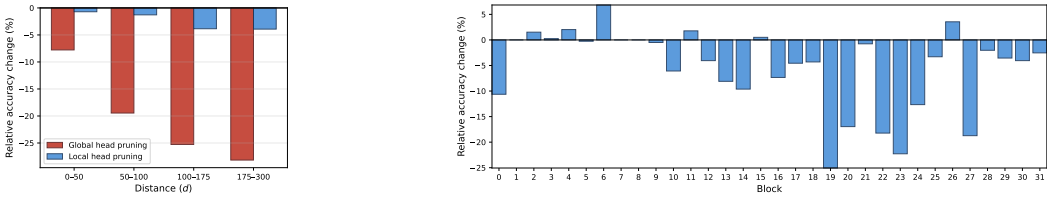
Let $\mathcal{M}$ denote the set of target blocks. For each target block $m \in \mathcal{M}$, let $\mathcal{G}^m = \{g_1, \ldots, g_k\}$ and $\mathcal{L}^m = \{j_1, \ldots, j_k\}$ denote the sets of selected global and local heads, respectively, where k is the number of heads in each set. We denote by $\mathbf{o}_h^m(i,\tau) \in \mathbb{R}^{D_h}$ the output of head h at token position i and denoising step τ in block m, where D_h is the head dimension. We construct the local space from the outputs of local heads:

$$\mathbf{A}^m(i,\tau) = \left[\mathbf{o}_{j_1}^m(i,\tau)^\top; \mathbf{o}_{j_2}^m(i,\tau)^\top; \ldots; \mathbf{o}_{j_k}^m(i,\tau)^\top\right] \in \mathbb{R}^{k \times D_h}. \tag{4}$$

The row space of $\mathbf{A}^m(i,\tau)$ defines a local space at token position i and denoising step τ. For each selected global head $g_l \in \mathcal{G}^m$, we compute a projection of its output onto the local space [26]:

$$\mathrm{Proj}_{\mathcal{L}^m}\left(\mathbf{o}_{g_l}^m(i,\tau)\right) = \mathbf{A}^m(i,\tau)^\top \left(\mathbf{A}^m(i,\tau)\mathbf{A}^m(i,\tau)^\top + \lambda \mathbf{I}\right)^{-1} \mathbf{A}^m(i,\tau)\mathbf{o}_{g_l}^m(i,\tau), \tag{5}$$

[2]We use a LongBench subset rather than the synthetic benchmark for head and block selection to reflect more realistic generation settings. To build the subset, we randomly sample 3 examples from each of the 21 tasks, including question answering, summarization, code, and dialogue, resulting in a total of 63 samples.



(a) Global vs. local head pruning.

(b) Block-wise pruning of global heads.

Figure 3: (a) Relative accuracy change after pruning global heads (red) or local heads (blue) across evidence–mask distances. Pruning global heads leads to larger degradation, especially at longer distances. (b) Block-wise relative accuracy change after removing global heads from each block individually. The impact varies substantially across blocks, indicating that a subset of blocks is particularly more critical for global-context utilization.

where λ is a small constant for numerical stability, which we set to 10^{-4} in our experiments. This projection is efficient, since it can be computed by solving the corresponding $k \times k$ linear system, instead of inverting the matrix in Eq. (5) [26]. Since k is the number of local heads in the block, which is typically small (*e.g.*, 8 in LLaDA-8B [5]), this adds negligible overhead during inference. Note that the residual

$$\mathbf{o}_{gl}^{m}(i,\tau) - \mathrm{Proj}_{\mathcal{L}^m}\left(\mathbf{o}_{gl}^{m}(i,\tau)\right) \tag{6}$$

removes the component of the global-head output aligned with local-head features, capturing the features that are more related to global context. GCG amplifies this residual, and updates the output of each selected global head:

$$\tilde{\mathbf{o}}_{gl}^{m}(i,\tau) = \mathbf{o}_{gl}^{m}(i,\tau) + s\left(\mathbf{o}_{gl}^{m}(i,\tau) - \mathrm{Proj}_{\mathcal{L}^m}\left(\mathbf{o}_{gl}^{m}(i,\tau)\right)\right), \tag{7}$$

where s is a guidance scale shared across all target heads and blocks. Outputs of all other heads are left unchanged. This update can be viewed as an implicit guidance mechanism without an explicit weak model. Rather than contrasting outputs from separate models, GCG directly modifies head-wise features within the same model, requiring only a single forward pass, while enhancing the performance significantly (blue line of Fig. 2a). GCG is also compatible with FlashAttention [27], since it operates on the features after attention computation.

4 Experiments

In this section, we first describe the experimental settings (Sec. 4.1). We then provide a quantitative comparison of GCG with existing guidance methods across various models and benchmarks (Sec. 4.2), and present an empirical analysis of GCG (Sec. 4.3).

4.1 Experimental settings

Models and benchmarks. We evaluate GCG on three representative diffusion LLMs, including LLaDA-8B [5], Dream-7B [6], and LLaDA-1.5 [24]. We use a diverse set of benchmarks covering scientific reasoning (ARC [16]), physical commonsense reasoning (PIQA [28]), mathematical reasoning (GSM8K-CoT [18]), code generation (HumanEval [17] and MBPP [29]), and long-context understanding (Ruler [30] and MiniLongBench [31]). Following the standard evaluation protocols of each benchmark, we report accuracy for ARC, PIQA, and GSM8K-CoT, and pass@1 for HumanEval and MBPP. For Ruler and MiniLongBench, we report the average score over all subtasks. We compare GCG with representative guidance methods for diffusion LLMs, including U-CFG [11] and A-CFG [12], which we reproduce based on the descriptions in their papers.

Implementation details. As described in Sec. 3.3, we use a training subset of LongBench [25] to select global and local heads and target blocks. Specifically, heads in each block are ranked by COM scores, where the top-p% and bottom-p% heads are selected as global and local heads, respectively. Target blocks are selected as the top-q% blocks with the largest performance drops after performing block-wise pruning of global heads, and these heads and blocks are fixed for each model. Unless otherwise specified, we set both p and q to 25, and the guidance scale s in Eq. (7) to 0.3. All experiments are performed on 8 A6000 GPUs. Further implementation details, including decoding settings, are provided in Appendix B.

Table 1: Quantitative comparison with existing guidance methods across models and benchmarks. We evaluate LLaDA-8B [5], Dream-7B [6], and LLaDA-1.5 [24] on ARC [16], PIQA [28], HumanEval [17], MBPP [29], GSM8K-CoT [18], Ruler [30], and MiniLongBench [31]. We also report the inference overhead of each method in terms of FLOPs, measured for a single denoising step with a context length of 1K tokens. Bold numbers indicate the best performance within each model.

Method	ARC	PIQA	HumanEval	MBPP	GSM8K-CoT	Ruler	MiniLongBench	FLOPs
LLaDA-8B [5]	43.3	72.4	12.2	16.0	46.7	26.2	24.8	**15.9T**
+ U-CFG [11]	42.4	74.1	7.9	14.8	36.7	28.9	22.8	31.8T
+ A-CFG [12]	45.7	73.8	11.6	16.6	34.7	27.5	24.4	31.8T
+ GCG	**46.2**	**74.4**	**12.8**	**18.4**	**48.4**	**29.5**	**25.9**	**15.9T**
Dream-7B [6]	54.2	75.5	1.8	15.0	3.9	28.0	26.5	**14.9T**
+ U-CFG [11]	55.6	74.8	3.1	14.6	4.3	28.3	26.7	29.8T
+ A-CFG [12]	56.1	74.7	**3.7**	12.6	5.2	27.1	26.8	29.8T
+ GCG	**58.4**	**75.6**	3.1	**17.0**	**5.4**	**28.5**	**27.1**	**14.9T**
LLaDA-1.5 [24]	51.4	73.5	15.9	17.2	37.8	37.6	31.6	**15.9T**
+ U-CFG [11]	53.8	73.2	18.1	**19.6**	38.2	38.2	29.2	31.8T
+ A-CFG [12]	**54.5**	73.8	17.4	18.4	37.9	38.4	29.4	31.8T
+ GCG	**54.5**	**73.9**	**18.9**	19.0	**38.5**	**39.3**	**32.2**	**15.9T**

4.2 Results

We show in Table 1 quantitative comparisons between GCG and existing guidance methods. We summarize our findings as follows: (1) GCG consistently improves baselines across various models and benchmarks. For example, on LLaDA-8B [5], GCG improves GSM8K-CoT [18] from 46.7 to 48.4 and Ruler from 26.2 to 29.5. This suggests that enhancing global-context utilization during denoising is broadly effective, rather than being specific to a particular model or task. (2) GCG achieves stronger overall performance than existing guidance methods. While U-CFG [11] and A-CFG [12] improve the baseline in some cases, their gains are less consistent, and they can even degrade performance on several tasks, such as GSM8K-CoT on LLaDA-8B. Note that these methods construct a weak model by replacing prompt or generated tokens with masked tokens, analogous to using a null prompt in image diffusion models [10]. However, masked tokens in diffusion LLMs are prediction targets rather than null conditions, making the role of the weak model less clear. GCG instead builds on our analysis of context utilization in diffusion LLMs and applies guidance to components that are important for leveraging global context, leading to more consistent improvements across tasks. (3) GCG is more efficient than existing guidance methods. Since U-CFG and A-CFG require an additional forward pass for the weak model, they substantially increase the inference cost. GCG avoids this overhead by modifying head features within a single forward pass, which requires only negligible overhead (*e.g.*, 0.003% FLOPs increase for LLaDA-8B). These results demonstrate that GCG is both effective and efficient compared to existing guidance methods for diffusion LLMs.

4.3 Analysis

Design choices of GCG. We analyze in Table 2 two design choices of GCG. We first analyze whether guidance should enhance global-context features rather than local-context features. To examine this, we construct local context guidance (LCG), a variant that reverses the direction of GCG. LCG projects outputs of local heads onto the global-head subspace and amplifies the difference between the original and projected outputs. We can see that LCG shows worse performance than GCG. This is particularly evident on benchmarks that require chain-of-thought reasoning (*e.g.*, GSM8K-CoT), or long-context understanding (*e.g.*, Ruler and MiniLongBench), which require more global-context utilization. This is consistent with our analysis in Sec. 3.2, where diffusion LLMs already rely heavily on local context during denoising. Further strengthening local-context features can therefore reinforce this bias, reducing the benefit of guidance. We further analyze the effect of target block selection. Instead of selecting the 25% blocks with the largest performance drops after block-wise pruning of global heads as in GCG, we apply the same global-head guidance operation to the 25% blocks with the smallest performance drops, which we refer to this variant as Least-Sensitive Blocks (LSB). Similar to LCG, LSB also shows worse performance than GCG, especially on benchmarks that

Table 2: Quantitative comparison on the design choices of GCG across models and benchmarks. LCG replaces global-head guidance with local-head guidance, while LSB applies GCG to the least sensitive blocks instead of the selected target blocks. Bold numbers indicate the best performance within each model. Please refer to the main text for details.

Method	ARC	PIQA	HumanEval	MBPP	GSM8K-CoT	Ruler	MiniLongBench
LLaDA-8B [5]	43.3	72.4	12.2	16.0	46.7	26.2	24.8
+ LCG	45.2	73.7	11.0	16.8	47.6	26.0	25.0
+ LSB	45.3	73.9	11.0	16.6	44.8	23.4	24.6
+ GCG	**46.2**	**74.4**	**12.8**	**18.4**	**48.4**	**29.5**	**25.9**
Dream-7B [6]	54.2	75.5	1.8	15.0	3.9	28.0	26.5
+ LCG	57.3	75.4	1.8	13.8	2.7	22.4	25.6
+ LSB	58.0	75.5	2.4	14.4	3.3	25.9	26.1
+ GCG	**58.4**	**75.6**	**3.1**	**17.0**	**5.4**	**28.5**	**27.1**
LLaDA-1.5 [24]	51.4	73.5	15.9	17.2	37.8	37.6	31.6
+ LCG	51.0	73.6	15.2	16.0	29.8	37.2	31.4
+ LSB	52.0	73.6	16.5	17.2	31.2	37.8	31.4
+ GCG	**54.5**	**73.9**	**18.9**	**19.0**	**38.5**	**39.3**	**32.2**

Table 3: Quantitative analysis of GCG with different ratios for selecting global and local heads (p) and target blocks (q) on LLaDA-8B [5]. Note that it is equivalent to the baseline without guidance when both p and q are set to 0. Best scores in each column are in bold.

p	q	ARC	PIQA	HumanEval	MBPP	GSM8K-CoT	Ruler	MiniLongBench
0.0	0.0	43.3	72.4	12.2	16.0	46.7	26.2	24.8
12.5	12.5	45.7	73.9	**13.4**	17.8	**49.5**	28.1	25.6
	25.0	44.9	73.9	12.2	**18.8**	49.1	28.3	25.0
	50.0	44.7	74.3	11.6	17.6	49.3	26.0	26.6
25.0	12.5	45.7	73.7	**13.4**	18.2	46.8	30.2	25.9
	25.0	**46.2**	**74.4**	12.8	18.4	48.4	29.5	25.9
	50.0	45.0	74.1	12.8	16.6	47.3	29.1	27.0
50.0	12.5	45.5	73.9	12.2	17.4	47.5	30.8	25.2
	25.0	45.1	74.0	12.2	18.0	47.3	**30.9**	24.8
	50.0	44.1	73.9	12.8	17.0	47.7	29.2	**27.7**

require more global-context utilization. This suggests that global-head guidance is effective when applied to blocks that are important for global-context utilization.

Hyperparameters. We analyze the sensitivity of GCG to the hyperparameters used for selecting heads and blocks and for controlling the guidance strength. We show in Table 3 the results obtained by sweeping $p, q \in \{12.5, 25, 50\}$, where p and q denote the percentages of selected heads and target blocks, respectively. We also show in Table 4 the results obtained by sweeping the guidance scale $s \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$. We can see that GCG consistently improves the baseline across a wide range of settings, indicating that it is robust to the choice of hyperparameters. This suggests that the effectiveness of GCG does not rely on a carefully tuned configuration, but rather comes from enhancing global-context features during denoising. Please refer to Appendix C for additional results on Dream-7B [6] and LLaDA-1.5 [24].

Qualitative analysis. We show in Fig. 4 a qualitative example on LLaDA-1.5 [24] using the synthetic infilling dataset, illustrated in Fig. 1. In this dataset, the model needs to recover the masked tokens by identifying the relevant evidence in the document. We can see that the baseline fails on recovering several masked tokens. Specifically, the baseline either relies on nearby distractor tokens (*e.g.*, predicting Project Radiant for `[MASK_1]`) or incorrectly uses irrelevant distant evidence (*e.g.*, predicting Approver-D for `[MASK_3]`). This suggests that the baseline can be misled by both local distractors and irrelevant global context. In contrast, GCG correctly recovers all masked tokens, indicating that it identifies and uses the relevant contextual information more effectively.

Calibration data. GCG selects global and local heads and target blocks using a subset of LongBench [25] (Sec. 3.3). To analyze whether GCG depends on this choice, we perform the selection

Table 4: Quantitative analysis of GCG with different guidance scale s on LLaDA-8B [5]. Note that it is equivalent to the baseline without guidance when s is set to 0. Best scores in each column are in bold.

s	ARC	PIQA	HumanEval	MBPP	GSM8K-CoT	Ruler	MiniLongBench
0.0	43.3	72.4	12.2	16.0	46.7	26.2	24.8
0.1	46.0	74.1	**12.8**	18.0	47.0	28.5	24.9
0.2	45.7	74.3	**12.8**	18.8	**50.6**	28.6	25.6
0.3	**46.2**	74.4	**12.8**	18.4	48.4	**29.5**	**25.9**
0.4	**46.2**	74.4	12.2	18.0	48.1	27.6	25.0
0.5	46.0	**74.5**	12.2	**19.4**	48.1	25.7	24.5

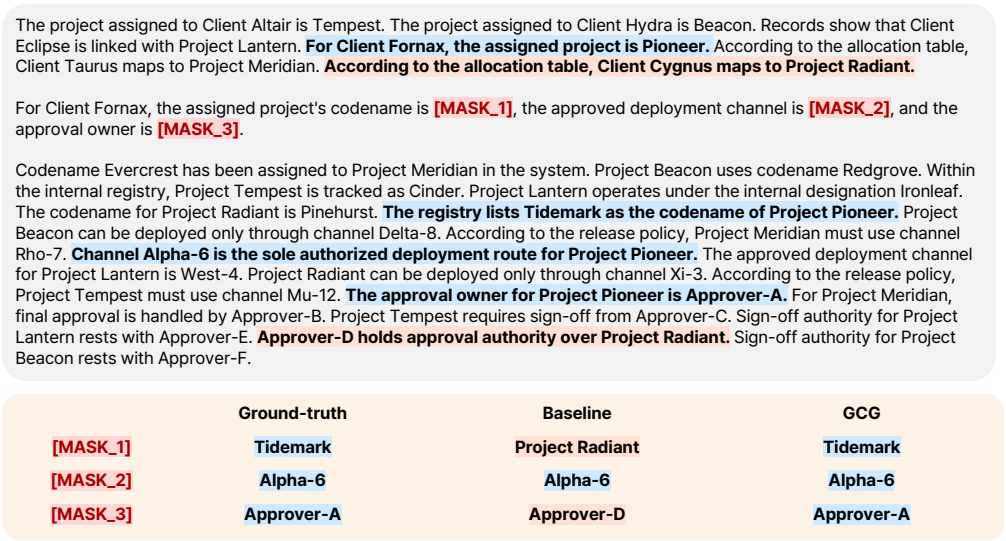


Figure 4: Qualitative example on LLaDA-1.5 [24] using the synthetic infilling benchmark. Blue highlights indicate supporting evidence for the correct predictions, orange highlights indicate distractor tokens used by the baseline.

process using different calibration sets, including GSM8K-CoT [18], HumanEval [17], and Ruler [30], while keeping all other settings fixed. We show in Table 5 the results on LLaDA-8B [5]. We can see that GCG consistently outperforms the baseline for all calibration sets, and the results obtained with different calibration sets are comparable to each other. This suggests that GCG is robust to the choice of calibration data.

Applicability to other transformers. GCG operates on the outputs of attention heads, and it does not depend on the training or inference procedure of diffusion LLMs. It can thus be applied to other transformer-based language models, such as autoregressive LLMs and bidirectional masked language models. We conjecture, however, that GCG is particularly beneficial for diffusion LLMs. At early denoising steps, many tokens near a masked position remain masked, and the local context provides limited information for the prediction. The prediction should thus rely more on distant evidence, making the bias toward local context more harmful in diffusion LLMs. In contrast, autoregressive LLMs predict each token from a fully observed prefix, and masked language models typically predict masked tokens in a single step with most surrounding tokens observed. To verify this, we apply GCG to Llama-3.2-3B [1] and RoBERTa-base [32], and show in Table 6 the results with different guidance scales. We can see that GCG improves the performance on several tasks for both models, while the gains are smaller and less consistent than those for diffusion LLMs. This suggests that GCG is applicable to general transformers, while more effective for diffusion LLMs, where the local context is sparse during denoising.

Table 5: Quantitative results of GCG on LLaDA-8B [5] with different calibration sets. Each row denotes the dataset used to select global heads, local heads, and target blocks. We report accuracy for ARC [16], PIQA [28], and GSM8K-CoT [18], pass@1 for HumanEval [17] and MBPP [29], and average scores for Ruler [30] and MiniLongBench [31]. Numbers in bold indicate the best performance.

Calibration set	ARC	PIQA	HumanEval	MBPP	GSM8K-CoT	Ruler	MiniLongBench
LLaDA-8B [5]	43.3	72.4	12.2	16.0	46.7	26.2	24.8
GSM8K-CoT [18]	45.7	**74.6**	12.8	18.0	**48.6**	28.7	25.4
HumanEval [17]	45.6	74.1	**13.4**	18.2	47.9	28.5	**26.1**
Ruler [30]	45.8	74.0	12.8	18.2	48.0	29.2	25.7
LongBench [25] (default)	**46.2**	74.4	12.8	**18.4**	48.4	**29.5**	25.9

Table 6: GCG on Llama-3.2-3B [1] and RoBERTa-base [32] across guidance scales s. For RoBERTa-base, we report development-set F1 on SQuAD v2 [33] and the average development-set score on GLUE [34]. Best scores in each column are in bold.

s	Llama-3.2-3B				RoBERTa-base	
	ARC-c	HumanEval	MBPP	MiniLongBench	SQuAD v2	GLUE
0.0	46.1	51.2	52.4	21.8	82.92	**84.70**
0.1	46.0	**51.8**	52.0	21.1	83.20	84.39
0.2	**46.6**	**51.8**	**53.0**	21.2	83.36	84.36
0.3	46.0	**51.8**	52.0	21.4	**83.44**	84.66
0.4	46.2	**51.8**	50.6	**21.9**	83.37	84.45
0.5	45.9	50.6	52.0	21.6	82.96	84.00

5 Limitations and Future Work

GCG requires a separate selection process to identify global and local heads and target blocks. In our experiments, this process takes about 2–3 hours per model on 8 NVIDIA A6000 GPUs. Note that the selection is performed only once for each model, and the selected heads and blocks are fixed across all benchmarks. Nevertheless, improving this selection process would further enhance the practicality of GCG, especially for larger diffusion LLMs. Future work could reduce the required number of samples for selection or replace block-wise ablation with lightweight proxy metrics for identifying target blocks.

6 Conclusion

We have presented an empirical analysis of how diffusion LLMs use contextual information during denoising. Our analysis shows that diffusion LLMs tend to rely more heavily on local context, while often underutilizing global context. Based on this observation, we have introduced GCG, a feature-space guidance method that enhances global-context utilization without constructing an explicit weak model. GCG identifies global heads and target blocks, and strengthens global-context features directly within a single forward pass. Extensive experiments on various diffusion LLMs and benchmarks demonstrate that GCG consistently improves performance over existing guidance methods, while incurring much lower inference overhead.

Acknowledgements

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No. RS-2022-00143524, Development of Fundamental Technology and Integrated Solution for Next-Generation Automatic Artificial Intelligence System, and No. RS-2026-25519622).

References

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

[2] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[3] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[4] Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *NeurIPS*, 2022.

[5] 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. *arXiv preprint arXiv:2502.09992*, 2025.

[6] Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models. *arXiv preprint arXiv:2508.15487*, 2025.

[7] 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. *ICLR*, 2026.

[8] Jinsong Li, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Jiaqi Wang, and Dahua Lin. Beyond fixed: Training-free variable-length denoising for diffusion large language models. *ICLR*, 2026.

[9] Xinyin Ma, Runpeng Yu, Gongfan Fang, and Xinchao Wang. dkv-cache: The cache for diffusion language models. *NeurIPS*, 2025.

[10] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *NeurIPSW*, 2021.

[11] Shen Nie, Fengqi Zhu, Chao Du, Tianyu Pang, Qian Liu, Guangtao Zeng, Min Lin, and Chongxuan Li. Scaling up masked diffusion models on text. *arXiv preprint arXiv:2410.18514*, 2024.

[12] Pengxiang Li, Shilin Yan, Joey Tsai, Renrui Zhang, Ruichuan An, Ziyu Guo, and Xiaowei Gao. Adaptive classifier-free guidance via dynamic low-confidence masking. *NeurIPS*, 2025.

[13] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *NeurIPS*, 2021.

[14] Linrui Ma, Yufei Cui, Kai Han, and Yunhe Wang. Diffusion in diffusion: Reclaiming global coherence in semi-autoregressive diffusion. *arXiv*, 2026.

[15] Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and LingPeng Kong. Diffuseq: Sequence to sequence text generation with diffusion models. *arXiv preprint arXiv:2210.08933*, 2022.

[16] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv*, 2018.

[17] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv*, 2021.

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

[19] Guanghan Wang, Yair Schiff, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Remasking discrete diffusion models with inference-time scaling. *NeurIPS*, 2025.

[20] Feng Hong, Geng Yu, Yushi Ye, Haicheng Huang, Huangjie Zheng, Ya Zhang, Yanfeng Wang, and Jiangchao Yao. Wide-in, narrow-out: Revokable decoding for efficient and effective dllms. *ICLR*, 2026.

[21] Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. *NeurIPS*, 2024.

[22] Donghoon Ahn, Hyoungwon Cho, Jaewon Min, Wooseok Jang, Jungwoo Kim, SeonHwa Kim, Hyun Hee Park, Kyong Hwan Jin, and Seungryong Kim. Self-rectifying diffusion sampling with perturbed-attention guidance. In *ECCV*, 2024.

[23] Chubin Chen, Jiashu Zhu, Xiaokun Feng, Nisha Huang, Meiqi Wu, Fangyuan Mao, Jiahong Wu, Xiangxiang Chu, and Xiu Li. S ^{2}-guidance: Stochastic self guidance for training-free enhancement of diffusion models. *ICLR*, 2026.

[24] Fengqi Zhu, Rongzhen Wang, Shen Nie, Xiaolu Zhang, Chunwei Wu, Jun Hu, Jun Zhou, Jianfei Chen, Yankai Lin, Ji-Rong Wen, et al. Llada 1.5: Variance-reduced preference optimization for large language diffusion models. *arXiv preprint arXiv:2505.19223*, 2025.

[25] Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, et al. Longbench: A bilingual, multitask benchmark for long context understanding. In *ACL*, 2024.

[26] Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013.

[27] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *NeurIPS*, 2022.

[28] Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *AAAI*, 2020.

[29] Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv*, 2021.

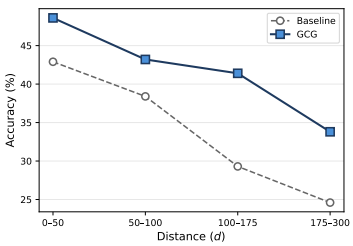
[30] Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. Ruler: What's the real context size of your long-context language models? In *COLM*, 2024.

[31] Zhongzhan Huang, Guoming Ling, Shanshan Zhong, Hefeng Wu, and Liang Lin. Minilongbench: The low-cost long context understanding benchmark for large language models. In *ACL*, 2025.

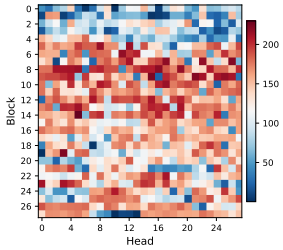
[32] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[33] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2018.

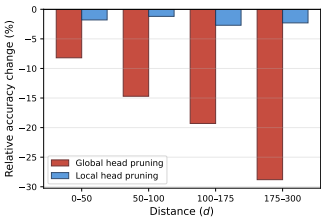
[34] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *ICLR*, 2019.



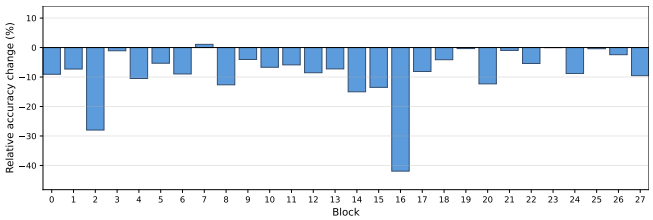
(a) Accuracy vs. evidence–mask distance.



(b) COM of attention heads across blocks.



(c) Global vs. local head pruning.



(d) Block-wise pruning of global heads.

Figure 5: Additional analyses of Dream-7B [6] using the synthetic infilling benchmark. (a) Accuracy as a function of the average evidence–mask distance. (b) COM heatmap across all blocks and heads. (c) Relative accuracy change after pruning global heads (red) or local heads (blue) across evidence–mask distances. (d) Block-wise relative accuracy change after removing global heads from each block individually.

A Synthetic infilling benchmark

We provide additional details on the synthetic infilling benchmark used in our analysis. Each sample is generated from predefined pools of synthetic entities, including 24 clients, 24 projects, 24 project codenames, 24 deployment channels, and 768 approval owners. The owner pool is constructed by combining 32 first-name identifiers with 24 last-name identifiers. Each pool contains synthetic names such as Orion, Nova, Vega, and Altair for clients; Zephyr, Meridian, Aurora, and Nimbus for projects; Aster, Northwind, Bluehaven, and Ironleaf for codenames; and North-7, East-2, West-4, and South-9 for deployment channels. We generate total 700 samples.

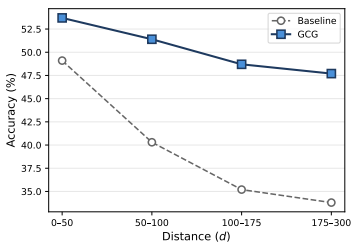
For each sample, we randomly sample a subset of entities and define mappings from clients to projects and from projects to their attributes. We then choose a target client and construct a document with masked tokens in the middle. The masked tokens correspond to the project codename, deployment channel, and approval owner of the project assigned to the target client. Solving the task therefore requires the model to first identify the project associated with the target client and then retrieve the attributes of that project.

We use a bidirectional evidence structure. The client-to-project evidence is placed before the masked tokens, while the project-attribute evidence is placed after them. The document also includes filler text and distractor statements involving non-target clients and projects, so that the answer cannot be inferred from the surrounding template or the target client alone.

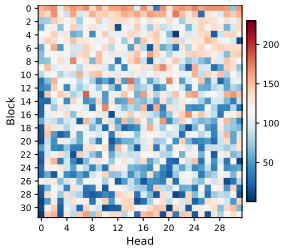
To study the effect of contextual distance, we randomize the amount of intervening context between the masked tokens and the supporting evidence. By mixing evidence sentences with filler text and distractor statements, the generated examples cover a broad range of realized evidence–mask distances, from approximately 5 to 300 tokens. This design allows us to compare local-context cases, where the evidence appears near the masked tokens, with global-context cases, where the evidence is separated by substantially longer context.

B More implementation details

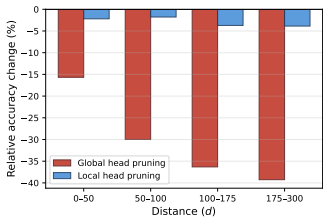
Calibration procedure. We describe the procedure for selecting global heads, local heads, and target blocks in Sec. 3.3. Note that the synthetic infilling benchmark in Sec. 3.2 is used only to analyze



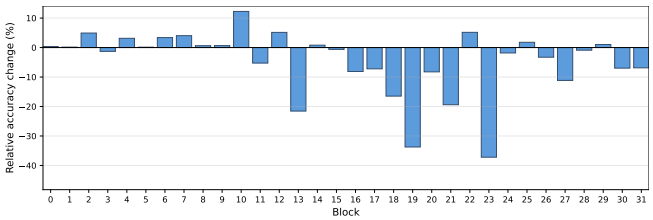
(a) Accuracy vs. evidence–mask distance.



(b) COM of attention heads across blocks.



(c) Global vs. local head pruning.



(d) Block-wise pruning of global heads.

Figure 6: Additional analyses of LLaDA-1.5 [24] using the synthetic infilling benchmark. (a) Accuracy as a function of the average evidence–mask distance. (b) COM heatmap across all blocks and heads. (c) Relative accuracy change after pruning global heads (red) or local heads (blue) across evidence–mask distances. (d) Block-wise relative accuracy change after removing global heads from each block individually.

the use of contextual information under controlled evidence–mask distances, and it is not used for the selection. Instead, we use natural examples from LongBench [25]. Specifically, we randomly sample three examples from each of the 21 tasks in LongBench, resulting in 63 calibration examples that cover question answering, summarization, code, and dialogue. The selection process consists of three steps: (1) We run the standard denoising process on each calibration example, and compute the COM score of each head h in block m using Eq. (3). We then average the COM scores over the calibration examples. (2) We rank the heads in each block by their average COM scores, and select the top-$p\%$ and bottom-$p\%$ heads as global heads $\mathcal{G}^m$ and local heads $\mathcal{L}^m$, respectively. (3) We prune the global heads in a single block by zeroing their outputs, and measure the resulting performance drop on the calibration examples. We repeat this for each block, and select the top-$q\%$ blocks with the largest performance drops as the set of target blocks $\mathcal{M}$. We set both p and q to 25 by default. This process is performed once for each model. The selected $\mathcal{G}^m$, $\mathcal{L}^m$, and $\mathcal{M}$ are then fixed across all evaluation benchmarks.

Decoding procedure. We provide additional details on the evaluation settings used in our experiments. For multiple choice benchmarks, including ARC [16] and PIQA [28], we follow the common evaluation protocol for diffusion LLMs, scoring each answer candidate using Monte Carlo likelihood estimation. Specifically, given a prompt and a candidate answer, we repeatedly sample a subset of answer tokens, replace them with masked tokens, and compute the log probability assigned to the original tokens at the masked positions. Each estimate is reweighted by the ratio between the answer length and the number of masked tokens. We average the resulting scores over 128 Monte Carlo samples and select the answer candidate with the highest estimated likelihood.

For generative benchmarks [17, 29, 18, 30, 31], we set the generation length and denoising steps according to the expected output length of each task. For HumanEval [17] and MBPP [29], we generate up to 256 tokens with 64 denoising steps. For GSM8K-CoT [18], we generate up to 512 tokens with 128 denoising steps. We set the number of few-shot examples to 0, 3, and 8 for HumanEval [17], MBPP [29], and GSM8K-CoT [18], respectively. For long context benchmarks, we set the prompt length to 8K tokens for both Ruler and MiniLongBench. We use 64 generation tokens and 16 denoising steps for Ruler, and 512 generation tokens and 128 denoising steps for MiniLongBench.

Table 7: Quantitative analysis of GCG with different head and block selection percentages on Dream-7B [6]. The parameters p and q control the percentages of selected heads and target blocks, respectively. Best scores in each column are in bold.

p	q	ARC	PIQA	HumanEval	MBPP	GSM8K-CoT	Ruler	MiniLongBench
0.0	0.0	54.2	75.5	1.8	15.0	3.9	28.0	26.5
12.5	12.5	58.4	75.7	1.8	15.0	5.1	29.8	26.8
	25.0	58.3	75.5	3.0	15.8	4.5	28.0	26.5
	50.0	58.8	75.7	3.0	15.4	5.2	30.8	**27.9**
25.0	12.5	58.8	75.8	1.8	15.8	4.9	29.5	26.8
	25.0	58.4	75.6	3.1	**17.0**	5.4	28.5	27.1
	50.0	58.9	75.8	3.7	14.4	4.2	30.4	26.6
50.0	12.5	58.5	75.6	3.7	15.0	5.2	29.4	26.6
	25.0	58.4	75.5	**4.3**	16.0	**6.2**	28.5	26.9
	50.0	**59.0**	**75.9**	1.8	16.6	5.4	**32.2**	26.7

We follow the official implementation of each model for decoding. For LLaDA [5] and LLaDA-1.5 [24], we unmask tokens in descending order of prediction confidence, so that positions with higher confidence are filled earlier during denoising. For Dream-7B [6], we follow its default random unmasking strategy, where token positions are selected randomly at each denoising step.

Table 8: Quantitative analysis of GCG with different head and block selection percentages on LLaDA-1.5 [24]. The parameters p and q control the percentages of selected heads and target blocks, respectively. Best scores in each column are in bold.

p	q	ARC	PIQA	HumanEval	MBPP	GSM8K-CoT	Ruler	MiniLongBench
0.0	0.0	51.4	73.5	15.9	17.2	37.8	37.6	31.6
12.5	12.5	54.8	73.6	17.1	18.8	37.1	37.4	**32.8**
	25.0	54.7	73.9	17.1	18.8	38.9	37.6	32.2
	50.0	**55.2**	74.0	18.3	18.2	38.5	38.8	32.6
25.0	12.5	54.4	73.9	17.7	19.0	39.4	37.9	32.5
	25.0	54.5	73.9	**18.9**	19.0	38.5	39.3	32.2
	50.0	54.8	**74.1**	18.3	18.0	**40.9**	37.4	32.7
50.0	12.5	54.6	73.8	18.3	17.8	37.5	41.6	32.0
	25.0	54.6	73.9	17.1	18.0	38.8	**42.3**	32.3
	50.0	55.1	73.9	**18.9**	**19.8**	39.3	39.7	31.9

C Additional analyses

We provide additional analyses on Dream-7B [6] and LLaDA-1.5 [24].

Analysis on context utilization. We provide additional analyses on context utilization for Dream-7B [6] and LLaDA-1.5 [24] in Figs. 5 and 6, respectively. We follow the same analysis protocol as in Sec. 3.2 using the synthetic infilling benchmark. Similar to the results on LLaDA-8B, both models show a consistent drop in accuracy as the average evidence–mask distance increases, indicating that they rely more heavily on local context and underutilize global context during denoising. The COM heatmaps also show that attention heads exhibit clear variation in their average attention distances, suggesting the existence of heads that attend more strongly to distant positions. We further observe that pruning global heads leads to larger performance degradation than pruning local heads, especially when the supporting evidence is distant. Finally, the block-wise pruning results show that the effect of removing global heads varies substantially across blocks. These results confirm that the main observations in Sec. 3.2 hold across different diffusion LLMs, where diffusion LLMs are biased toward local context, and a subset of global heads and blocks plays an important role in leveraging global context.

Table 9: Quantitative analysis of GCG with different guidance scales on Dream-7B [6]. The setting $s = 0$ corresponds to the baseline without guidance. Best scores in each column are in bold.

s	ARC	PIQA	HumanEval	MBPP	GSM8K-CoT	Ruler	MiniLongBench
0.0	54.2	75.5	1.8	15.0	3.9	28.0	26.5
0.1	58.5	75.5	2.4	**17.6**	5.2	28.7	**27.4**
0.2	58.3	75.6	2.4	15.8	5.1	**28.9**	**27.4**
0.3	58.4	75.6	3.1	17.0	5.4	28.5	27.1
0.4	58.4	75.6	3.1	14.6	**5.5**	28.1	26.5
0.5	**58.6**	**76.4**	**4.3**	16.6	**5.5**	28.1	26.9

Table 10: Quantitative analysis of GCG with different guidance scales on LLaDA-1.5 [24]. The setting $s = 0$ corresponds to the baseline without guidance. Best scores in each column are in bold.

s	ARC	PIQA	HumanEval	MBPP	GSM8K-CoT	Ruler	MiniLongBench
0.0	51.4	73.5	15.9	17.2	37.8	37.6	31.6
0.1	54.4	73.8	17.1	18.6	38.4	38.0	31.8
0.2	54.3	73.8	**18.9**	18.8	**39.0**	39.3	32.5
0.3	54.5	**73.9**	**18.9**	**19.0**	38.5	39.3	32.2
0.4	**54.6**	**73.9**	**18.9**	16.8	38.4	**39.5**	32.6
0.5	**54.6**	73.8	18.3	**19.0**	**39.0**	38.5	**33.5**

Analysis on hyperparameters. We further evaluate the sensitivity of GCG to the head and block selection ratios, p and q, and the guidance scale s on Dream-7B and LLaDA-1.5. Tables 7 and 8 show the results obtained by varying p and q, while Tables 9 and 10 show the results obtained by varying s. We can see that GCG generally improves the baseline across a wide range of settings for both models. These results suggest that GCG is not highly sensitive to a specific hyperparameter configuration, and that its gains do not rely on careful hyperparameter tuning.

D Broader impacts

This work aims to improve diffusion large language models (LLMs) by making guidance more effective and efficient. A potential positive impact is that GCG reduces the inference overhead of guided generation by avoiding the additional forward pass required by existing guidance methods. This can make diffusion LLMs more computationally efficient, potentially reducing energy consumption and improving accessibility in resource-constrained settings.

However, improvements in language generation may also have negative societal impacts. More efficient and capable LLMs could be misused to generate misinformation, spam, phishing messages, or other harmful content. Moreover, since GCG operates on existing diffusion LLMs, it may inherit biases, privacy risks, hallucinations, and other limitations of the underlying models. Although our work is foundational and does not deploy a user-facing system, we encourage appropriate safeguards when applying GCG, such as safety evaluation, content moderation, abuse monitoring, and human oversight in high-stakes applications.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the main claims, contributions, and scope of the paper, including the analysis of contextual-information use in diffusion LLMs and the proposed GCG method.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations of the proposed method in Sec. 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper does not present formal theoretical results requiring assumptions or proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper provides the experimental settings, model details, benchmark information, implementation details, and evaluation protocols needed to reproduce the main results in Sec. 4.1 and Appendix B.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code is available at https://cvlab.yonsei.ac.kr/projects/GCG/.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the models, datasets, baselines, benchmarks, evaluation metrics, and key hyperparameters used in the experiments in Sec. 4.1 and Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We do not report error bars. However, we conduct extensive evaluations across multiple diffusion LLMs and diverse benchmarks, where GCG shows consistent improvements in most settings, supporting its effectiveness and generality.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports the computational resources and inference cost used for the experiments, including hardware information and FLOPs comparisons, in Sec. 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The research conforms to the NeurIPS Code of Ethics. The paper does not involve human subjects, private data, or unsafe data collection.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses potential broader impacts of improving diffusion LLMs, including efficiency benefits and possible risks associated with improved language generation, in Appendix D.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release new pretrained models or high-risk datasets requiring additional safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper properly cites the existing models, datasets, and benchmarks used in the experiments.

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: Our paper does not have new assets to release.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects and therefore does not require IRB approval or equivalent review.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The paper studies diffusion LLMs as target models, but LLMs were not used as a non-standard component of the research methodology. Any LLM assistance was limited to minor writing or editing and did not affect the scientific content of the paper.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.