DO WE NEED DEPENDENCY-READY DECODING FOR MASKED DIFFUSION LANGUAGE MODELS?

Anonymous authors
Paper under double-blind review

ABSTRACT

Masked diffusion models (MDMs) have shown strong performance on structured reasoning tasks such as Sudoku, Trip Planning, and Countdown. MDMs introduce a freedom in the order in which tokens may be unmasked, and to exercise this freedom for good accuracy, early decoding strategies relied on signals of confidence or the downstream impact of decoding a token. However, simple examples seem to illustrate that resolving a token's dependencies before unmasking it should be important. Therefore, we examine whether this principle, which we call *dependency-ready decoding*, would contribute to better accuracy, from a practical and a theoretical perspective. We examine a recent decoder that implements dependency-readiness through attention, and generally find that when it performs better than confidence-based or future utility baselines, its decoding order is collapsing to left-to-right decoding. For theoretical analysis, we establish a definition of dependency without resorting to attention, measured by an information-theoretic score of how much revealing one masked token would shift the model's prediction for another. We prove that when the margin is large enough, no single revelation can change the top prediction, so in this regime max-margin decoding already commits dependency-ready tokens. These findings leave only a narrow scope for future dependency-ready decoders: their potential benefit is limited to positions with small margins, and reliance on attention should be questioned.

1 INTRODUCTION

Diffusion models have achieved remarkable success in image generation, and recent work has extended them to language. Masked diffusion language models (MDMs) (Austin et al., 2021a; Lou et al., 2024; Sahoo et al., 2024; Shi et al., 2024), the dominant approach in this setting, generate text by iteratively revealing tokens from a fully masked sequence, rather than following the fixed left-to-right factorization of autoregressive models. At inference time, this non-causal generation process offers two key advantages. First, multiple tokens can be committed in parallel during each denoising step, substantially improving decoding throughput (Khanna et al., 2025; Google DeepMind, 2025; Wu et al., 2025; Arriola et al., 2025; Ben-Hamu et al., 2025; Kim et al., 2026a). Second, the model is free to choose the order in which tokens are revealed. MDMs have shown strong performance on code generation (Gong et al., 2025; Xie et al., 2025) and, most notably, on structured reasoning tasks such as Sudoku (Shah et al., 2024; Kim et al., 2025a), Trip Planning (Ye et al., 2025b), and Countdown (Ye et al., 2025a). Prior work attributes this success to their flexible generation order (Kim et al., 2025a; Ye et al., 2025a), which may allow decoding to better accommodate the underlying dependency structure of these tasks (Ye et al., 2025a). With the emergence of large instruction-tuned MDMs such as LLaDA (Nie et al., 2025), iLLaDA (Nie et al., 2026), and Dream (Ye et al., 2025b), the decoding policy has become a central component of inference.

MDMs generate text through iterative denoising: at each step, the model predicts all masked tokens, and a decoding method selects a subset to reveal, which then serves as context for later steps. While the first MDMs revealed positions at random (Sahoo et al., 2024), later methods relied on confidence-based criteria such as maximum probability (Nie et al., 2025), maximum margin (Kim et al., 2025a), or minimum entropy (Ye et al., 2025b). More recent methods, such as the Info-Gain Sampler (Yang et al., 2026), SOAR (Cao et al., 2026), and DAPD (Kim et al., 2026a), instead focus on *future utility*, that is, how much revealing a token helps predict the tokens that are still masked.

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.

Considering simple puzzle examples such as Sudoku suggests that resolving token dependencies before unmasking a token might be important, an intuition which we explore in Section 3. This principle, which we call *dependency-ready decoding*, at first appears to be different from the confidence or future utility considerations. The recent Dependency-Oriented Sampler (DOS) (Zhou et al., 2026) follows this principle using attention as a proxy for dependency, following DAPD (Kim et al., 2026a) and ADAS (Sahin et al., 2026), which had used attention to avoid parallel decoding conflicts. At each step, DOS reveals the masked positions with the highest *groundedness*, that is, the ones that attend most to the tokens already revealed.

In this paper, we examine the value of dependency-ready decoding, both in practice in its attention-based DOS implementation, and in theory.

From a practical perspective, we reimplement DOS on LLaDA and Dream and add a left-to-right baseline that prior work did not report. We find that generally, when DOS performs better than confidence-based or future utility baselines, its decoding order is collapsing to left-to-right decoding. Its previously reported advantage over confidence-based methods shrinks or disappears once the end-of-sequence (EOS) fix of Nie et al. (2025) is applied.

From a theoretical perspective, we show that large margins already certify dependency readiness. Without operationalizing dependency through attention, we give a theoretical definition based on how much revealing a single missing token would change the model's prediction. We capture this with the Pre-Commit score, an information-theoretic measure of how far the prediction for a masked token can shift when any single other masked token is revealed at its ground-truth value. Under this definition, we prove that when the margin is large relative to this score, no single revelation can change the top prediction, so maximum-margin decoding already commits dependency-ready tokens whenever its margin is large enough.

Together, these results suggest that dependency-ready strategies should reconsider their reliance on attention, and that their benefit is limited to decoding token positions with small margins.

In summary, our contributions are:

1. We highlight and explain a collapse of attention based dependency-ready decoding to left-to-right decoding.
2. We show that an EOS handling issue conflates previous comparisons of dependency-ready decoding strategies against confidence decoding strategies.
3. We establish a new definition of dependency ready token selection in an oracle setting that does not use attention.
4. We show that dependency readiness already impacts maximum margin token selection.

2 Background

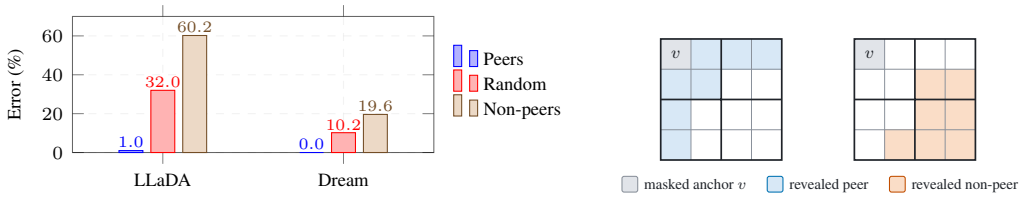
Masked diffusion language models. MDMs (Austin et al., 2021a; Lou et al., 2024; Sahoo et al., 2024; Shi et al., 2024) generate a sequence by iteratively replacing [MASK] tokens with vocabulary symbols. Let $\mathbf{r}$ denote the prompt and $\mathbf{x}^{(t)} \in (\mathcal{V} \cup \{\mathbf{m}\})^L$ the partially decoded sequence at reverse step t. The remaining masked positions are $\mathcal{F}_t = \{v : x_v^{(t)} = \mathbf{m}\}$, while the revealed context consists of the prompt together with every committed token, $\mathcal{R}_t = (\mathbf{r}, \{x_v^{(t)}\}_{v \notin \mathcal{F}_t})$. At every reverse step the model predicts a categorical distribution $p_\theta(X_v \mid \mathcal{R}_t)$ for every remaining masked position $v \in \mathcal{F}_t$, from which the decoder selects positions to reveal.

Decoding as an ordering problem. Unlike autoregressive models, which fix a left-to-right factorization, an MDM is free to choose which positions to commit at each reverse step. A decoding policy therefore maps the current state to a subset of masked positions, $\pi : (\mathcal{R}_t, \mathcal{F}_t) \to S_t \subseteq \mathcal{F}_t$. When $|S_t| = 1$, the policy reduces to a permutation of the generation positions. We refer to this permutation as the *decoding order*. Existing decoders differ almost entirely in how they choose this order. Practical MDMs often commit multiple positions simultaneously for efficiency. This approximates the joint predictive distribution as

$$p_\theta(X_{S_t} \mid \mathcal{R}_t) \approx \prod\nolimits_{v \in S_t} p_\theta(X_v \mid \mathcal{R}_t), \tag{1}$$

which is exact only if the selected positions are conditionally independent. Consequently, decoding quality depends not only on prediction accuracy but also on whether positions are revealed in an order consistent with their underlying dependencies.

3 Why Use Dependency Information in Decoding?



(a) (b)

Figure 1: (a) **Context relevance matters.** Prediction error at a masked target cell after revealing seven of its peers, seven random cells, or seven non-peers, over 120 4×4 Sudoku puzzles. (b) The seven peers (left) and seven non-peers (right) of a target cell v.

In the 4×4 Sudoku task of Ye et al. (2025a), each puzzle has a unique solution, and the rules make explicit which cells depend on which. For a target cell v, we call the cells in its row, column, or 2×2 subgrid its *peers* (Figure 1b). Peers directly constrain the value of v, while the remaining cells, the *non-peers*, do not. To separate which cells are revealed from how many are revealed, we start from the same partially revealed puzzle, reveal seven cells at their ground-truth values, and then measure the prediction error at a fixed target cell. The seven cells are all peers, all non-peers, or chosen at random. Some of them may already be revealed, in which case fewer than seven are new. Figure 1a shows the results for LLaDA and Dream. With the peers revealed, both models make almost no errors on the target (1.0% for LLaDA and 0.0% for Dream). With the non-peers revealed, the error reaches 60.2% and 19.6%, and random cells fall in between. The same amount of context therefore helps much more when it contains the target's peers.

We observe the same phenomenon in individual GSM8K (Cobbe et al., 2021) examples (Figure 2). In one example, computing a total cost requires intermediate products such as 3×16.50 and 3×22.50. Under maximum-probability decoding, the model commits the result tokens before the operand tokens are revealed, so it must estimate the intermediate values from incomplete context. This gives 49 instead of 49.50 and 67 instead of 67.50, and a wrong total of 242. We then repeat the decoding but forbid selecting a result token while any of its operand tokens is still masked. With this constraint, the model produces the exact intermediate values 49.50, 67.50, and 126, and the correct total of 243. The model, the maximum-probability rule, and the decoding budget stay the same, so only the decoding order differs.

Q: Mishka buys 3 pairs of shorts that cost \$16.50 each, 3 pairs of pants that cost \$22.50 each, and 3 pairs of shoes that cost \$42 each. How much does she spend in total?

MDM											
	max-prob	3 ×		= 49	3 ×		= 67	3 ×		= 126	total **242** ✗
	operands first	3 ×	16.50	= 49.50	3 ×	22.50	= 67.50	3 ×	42	= 126	total **243** ✓

Figure 2: GSM8K example: the same MDM decoded in two orders.

In both cases, the model predicts a token more accurately once specific other tokens are revealed, namely the peers of a Sudoku cell or the operands of a GSM8K result. This motivates dependency-ready decoding, which delays committing a token until most of the tokens that affect its prediction are revealed, a notion we make precise in Section 6. A task-agnostic decoder, however, does not know these tokens in advance and has to estimate them from the model. In the Section 5, we study the use of attention to estimate these tokens, as proposed in DOS.

4 Related Work

Confidence-based decoding. Early MDMs selected decoding positions uniformly at random, mirroring the masking process used during training (Austin et al., 2021a; Lou et al., 2024; Sahoo et al.,

2024). As larger MDMs emerged, confidence-based scheduling became the dominant decoding strategy. Variations on confidence-based scheduling rank masked positions according to maximum token probability (Nie et al., 2025), maximum margin (Kim et al., 2025a), or predictive entropy (Ben-Hamu et al., 2025). The most common scores are

$$s_v^{\text{conf}} = \max_w p_\theta(x_v = w \mid \mathcal{R}_t), \qquad s_v^{\text{margin}} = p_\theta(x_v = w_1 \mid \mathcal{R}_t) - p_\theta(x_v = w_2 \mid \mathcal{R}_t),$$

where s_v denotes the confidence score assigned to the masked position v, and w ranges over vocabulary tokens. In the margin score, w_1 and w_2 denote the most likely and second-most-likely tokens, respectively. Although these heuristics differ in how confidence is measured, they all operate on the same marginal predictive distribution $p_\theta(X_v \mid \mathcal{R}_t)$. Consequently, they estimate how certain the model is about its current prediction, but not whether that prediction still depends on information contained in unrevealed tokens.

Beyond confidence-based decoding. Recent work has explored decoding strategies that move beyond purely greedy confidence-based scheduling. We broadly divide these approaches into three categories. The first introduces additional information or supervision to learn or construct a desirable decoding trajectory. For example, Asano et al. (2026) learns an unmasking policy from an oracle trajectory, while Huang et al. (2026) uses an auxiliary mechanism to guide unmasking. Such approaches can provide richer trajectory-level information, but introduce an additional source of supervision or computation.

The second category takes a *utilitarian* approach, evaluating how a current unmasking decision may affect future decoding. Information-Gain Sampler (Yang et al., 2026) selects tokens based on their expected reduction of marginal entropy in the remaining masked positions, while SOAR (Cao et al., 2026) searches over alternative unmasking orders when confidence is insufficient. Dependency-Aware Parallel Decoding (DAPD) (Kim et al., 2026a), despite the word dependency, shares the same principle. At each step, it selects one or more tokens with the *highest impact* on other masked tokens (as measured by symmetrized attention), and decodes them.

A contrary principle is to prioritize the *readiness* of the decoded tokens, rather than future utility. DOS (Zhou et al., 2026) examines whether a token to be decoded has unresolved dependencies according to the attention matrix, *i.e.*, whether it is *ready* to be decoded. It is the focus of our analysis. For a masked target position $v \in F_t$, it defines the dependency score as groundedness:

$$Gr(v) = \sum_{u \in \mathcal{R}_t} \text{Attn}(v, u), \tag{2}$$

measuring the extent to which the token at position v attends to already *unmasked* positions u. In the sequential variant, each iteration step decodes at the position with the maximum groundedness score. Because it is score-based, this strategy is compatible with parallel decoding methods.

Other methods, including KLASS (Kim et al., 2025b), Fast-dLLM (Wu et al., 2025), and Prophet (Li et al., 2026), primarily target inference acceleration through more aggressive parallel decoding. These methods reduce the number of denoising steps but do not directly address which positions are safest to commit. In the parallel setting, DAPD avoids simultaneously decoding tokens with strong dependencies to each other, as measured by symmetrized self-attention. The Entropy-Bound Sampler (Ben-Hamu et al., 2025) can be applied with any score-based decoding strategy to choose multiple tokens to decode at each step, up to an entropy budget. In this way, the parallel decoding issue may be considered as orthogonal to a score which prioritizes tokens to be decoded.

Block diffusion (Arriola et al., 2025) is another way of achieving parallelization while avoiding the dangers of predicting faraway tokens. It constrains blocks to be predicted from left to right, with multiple tokens selected at each step within a single block until the block is completed. In this way, it provides an autoregressive heuristic to decoding. Because, as we will show, DOS already mostly follows a left-to-right order when given complete freedom, there is no need for us to require block structure, a hyperparameter setting which would confound a comparison between methods.

5 Dependency-Ready Decoding With Attention

Section 3 shows that a token is easier to predict once the tokens that constrain it are revealed. We now test whether a decoder that uses attention to find tokens with resolved dependencies improves

accuracy in practice. We focus on DOS described in Section 4, which decodes the masked position that attends most to already revealed tokens, and compare it with confidence-based decoders, decoders based on future utility (Info-Gain and DAPD), and a left-to-right baseline. For each method, we measure accuracy and how far its decoding order is from the left-to-right order. Because DOS was originally compared without the EOS fix described below, we report results with and without it.

5.1 Experimental Setup

Unless otherwise stated, all experiments use LLaDA-8B-Instruct (Nie et al., 2025) or Dream-v0-Instruct-7B (Ye et al., 2025b) with the same prompts, denoising schedule, and decoding budget across all methods. The only difference between policies is the order in which masked positions are committed; token predictions are always obtained from the same model without retraining or additional forward passes. All experiments were conducted on a compute server equipped with eight NVIDIA A100-PCIE GPUs (40 GB memory each) running CUDA 12.2.

The EOS Fix. The LLaDA paper (Nie et al., 2025) suggested a measure to prevent the premature generation of EOS tokens. When uncontrolled, models may tend to fill the tail of their generation context with EOS tokens before committing any meaningful tokens, ultimately not leaving enough room for meaningful content (Kim et al., 2026b). This measure, which we refer to as the "EOS Fix," stops the selection of positions where EOS is the maximum-probability token candidate, unless EOS is the maximum-probability token candidate at every masked position. Effectively, it is a filter on the masked positions that may be filled by a decoding strategy at a given step. Our results are reported for decoding methods with the EOS Fix either on or off, as indicated.

Tasks. Tables 1 and 2 report results on GSM8K (Cobbe et al., 2021) and MATH500 (Hendrycks et al., 2021), covering diverse multi-step reasoning and structured prediction tasks, on the coding tasks HumanEval (Chen et al., 2021) and MBPP (Austin et al., 2021b), and on the puzzle tasks of Sudoku 9x9 (Shah et al., 2024), Trip Planning (Zheng et al., 2024), and Countdown with three numbers (Ye et al., 2025a). For Sudoku 9x9, we measure per-token accuracy. All remaining benchmarks are evaluated using the official evaluation suite released with Dream (Ye et al., 2025b), keeping its datasets, prompt templates, in-context exemplars and scoring scripts unchanged. In all runs, the number of decoding steps equals the generation length. Following Dream's evaluation scripts, we use 256 tokens for GSM8K, 512 for MATH500, 256 for Trip Planning and 24 for Countdown. For HumanEval and MBPP, we replace Dream's 768 and 1024 tokens with LLaDA's 512 and 256 (Nie et al., 2025), closer to the 256 used by DOS (Zhou et al., 2026) and DAPD (Kim et al., 2026a). For Sudoku 9x9, we use 81 tokens, one per cell, as in Ye et al. (2025a).

Decoding strategies. We evaluate maximum margin, maximum probability ("standard"), minimum entropy, DOS, DAPD, and Info-Gain Sampler against a baseline of left to right decoding. To give the sampler full freedom to decode any token and avoid confounding issues of parallelism, no blocks are used, and each method is evaluated in a setting where a single token is decoded at each step. The DOS authors found that different layers performed optimally for different tasks, but that the first layer was a consistently good choice (Figure 4 and Appendix B.3, Zhou et al. (2026)). We evaluate this as "DOS (1)." DAPD reports that upper-layer attention best recovers a known synthetic Markov Random Field structure, and therefore averages attention over the final 30% of layers in large models, so we implement DOS with these final layers as well for comparison ("DOS (late)").

5.2 Results

For LLaDA and Dream, in the chain-of-thought tasks (Tables 1 and 2, left sides), left-to-right decoding and DOS have an advantage. However, confidence based methods dominate for code infilling and flexible generation tasks (Table 1, middle and right) for LLaDA. For Dream, the future utility-maximizing Info-Gain may beat the confidence methods for flexible generation (Table 2, right side) and left-to-right is best even for code infilling (Table 2, middle). Apart from a marginal 0.23% gain for DOS (late) on GSM8K, none of the results show a superiority for the dependency-ready DOS method on any benchmark except when left-to-right decoding is best.

Method	EOS fix	GSM8K-CoT	MATH500	HumanEval	MBPP	Trip	Sudoku	cd3
Max Prob	Off	50.49	19.20	15.24	22.60	10.12	37.06	45.36
Max Margin	Off	50.34	19.40	15.85	25.40	11.75	38.15	44.76
Min Entropy	Off	52.39	20.00	23.17	16.80	6.44	35.32	40.73
Max Prob	On	63.84	27.20	28.05	39.00	10.38	**39.60**	36.29
Max Margin	On	64.37	28.20	29.88	**42.80**	9.62	39.34	37.40
Min Entropy	On	63.47	25.80	**37.20**	40.60	9.75	39.51	36.90
DOS (1)	On	76.95	**37.00**	36.59	37.80	5.81	21.67	32.36
DOS (late)	On	**77.18**	36.40	35.37	37.60	6.56	21.71	31.05
Info-Gain	On	65.88	27.40	29.27	39.60	10.25	37.90	34.38
DAPD	On	65.05	24.40	26.83	37.40	8.69	34.58	36.39
DOS (1)	Off	76.95	**37.00**	36.59	37.80	6.69	21.71	32.36
DOS (late)	Off	74.45	36.20	35.37	37.80	6.62	21.40	33.27
Info-Gain	Off	52.08	18.80	17.68	27.00	**16.25**	36.58	44.46
DAPD	Off	53.30	21.80	11.59	11.60	4.81	32.04	**47.37**
Left to Right		76.95	**37.00**	36.59	37.80	6.69	21.71	32.36

Table 1: Evaluation of decoding methods on LLaDA Instruct, with $\mathrm{tmp} = 0$ and no Block Diffusion, on GSM8K-CoT(8-shot), MATH500(4-shot), HumanEval(0-shot), MBPP(0-shot), Trip Planning (2-shot), 9x9 Sudoku (5-shot), and 3-number Countdown (8-shot).

Method	EOS fix	GSM8K-CoT	MATH500	HumanEval	MBPP	Trip	Sudoku	cd3
Max Prob	Off	37.83	7.60	26.83	45.40	14.81	71.89	53.53
Max Margin	Off	39.73	8.20	28.66	47.40	14.25	**72.21**	53.12
Min Entropy	Off	35.63	7.00	21.95	37.00	15.75	69.55	53.02
Max Prob	On	80.59	37.20	53.66	56.80	15.94	70.11	54.74
Max Margin	On	81.12	36.80	46.95	55.80	15.88	70.27	52.62
Min Entropy	On	**82.64**	40.60	55.49	56.80	15.69	69.03	53.02
DOS (1)	On	81.05	**41.60**	**58.54**	**57.60**	11.88	49.40	45.26
DOS (late)	On	77.86	33.20	53.66	56.80	13.31	49.40	45.26
Info-Gain	On	81.35	40.80	**58.54**	56.20	15.94	70.11	54.74
DAPD	On	73.69	32.40	45.12	57.00	10.12	69.59	52.22
DOS (1)	Off	81.05	**41.60**	**58.54**	**57.60**	11.94	49.40	45.26
DOS (late)	Off	58.61	18.00	17.07	37.20	6.00	49.40	45.26
Info-Gain	Off	45.26	10.80	32.32	50.80	**17.00**	70.96	**55.24**
DAPD	Off	35.94	6.40	15.85	54.00	0.06	66.23	51.81
Left to Right		81.05	**41.60**	**58.54**	**57.60**	11.94	49.40	45.26

Table 2: Evaluation of decoding methods on Dream-v0-Instruct-7B on GSM8K-CoT, MATH500, HumanEval, MBPP, Trip Planning, 9x9 Sudoku, and 3-number Countdown.

5.3 Why DOS Decodes Left-to-Right

DOS maximizes $Gr(j) = \sum_{i=1}^{M} Attn(j, i)$. At least at the first layer, RoPE leads to "decaying inter-token dependency with increasing relative distances" in the attention scores (see Section 3.4.3 of Su et al. (2024)). This is a property of an upper bound rather than strict monotonicity, but if it were strictly monotone, it would imply that $Gr(j) > Gr(k)$ when $M < j < k$. In this case, $Gr(j)$ would be maximized at $j = M + 1$, and the next token to the right would be decoded by DOS. Thus, DOS at the first layer would never escape autoregressive generation.

Figure 3 displays the normalized Kendall τ distance (Kendall, 1938) between the left to right decoding order and the decoding orders chosen by DOS and other baselines. This metric counts the fraction of pairs of position indices (i, j) with $i < j$ such that position i is decoded after position j (*i.e.*, in a different relative order than left-to-right would decode). At the first layer, DOS stays so close to the left-to-right order that its τ distance rounds to .00 on every task. Applied to LLaDA, even at a later layer, DOS remains much closer to left-to-right order than the confidence-based base-

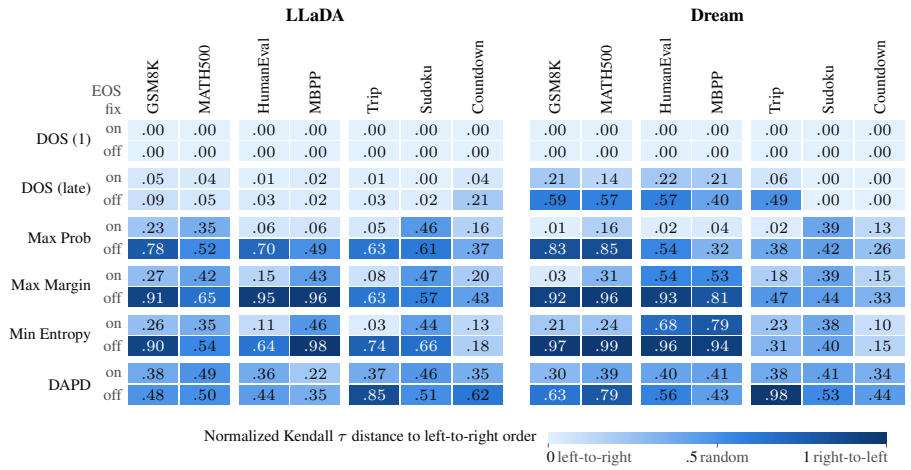


Figure 3: Normalized Kendall τ distance between the decoding order of each method and the left-to-right order.

lines. Although DOS escapes left-to-right behavior more on Dream late layers, it does so without gaining accuracy on any dataset over using the first layer. Even with late layers, DOS has nearly zero Kendall τ on the Sudoku task where dependencies are not expected to be left-to-right.

For the other methods, keeping the EOS fix on generally reduces the τ distance, consistent with the observation that the EOS is often filled in from the end when these methods are uncontrolled (Kim et al., 2026b). With the EOS fix on, the confidence methods generally have highest τ for Sudoku, where left-to-right got lower accuracy, and where the notion of peers suggests a better dependency ready decoding order.

5.4 Why Confidence Methods Appeared Not to Work

Zhou et al. (2026) reported substantial gains for DOS over maximum probability, maximum margin, and minimum entropy, but using smaller generation lengths and not applying an EOS fix. It is unlikely that EOS is ever the top candidate at the next masked position when decoding from left to right until the response is finished, and we indeed observe that the EOS fix has no impact on the DOS results at the first layer. However, the EOS fix is a very necessary repair for confidence-based strategies, as EOS predictions near the end of the generation limit may be considered confident without any careful analysis. In our results, we find that simply applying this fix gives some confidence-based methods a double-digit increase in performance, making them outperform DOS with LLaDA on the coding tasks of HumanEval and MBPP where they otherwise would trail it.

6 Dependency Ready Decoding Without Attention

Our motivation (Section 3) suggests that dependency-ready decoding should be useful, but in our experiments, the attention-based DOS fails to gain a consistent lead over confidence-based methods or left-to-right decoding, depending on the task. In this section, without operationalizing dependency through attention, we establish a theoretical definition based on how much revealing a single missing token would change the model's prediction. In this sense, we can examine whether the confidence-based decoding strategies already consider some form of dependency readiness.

6.1 Defining unresolved dependency

Let F_t denote the set of positions that remain masked at denoising step t, and let R_t denote the currently revealed context. For a masked target position $v \in F_t$, the model predicts the categorical distribution $p_\theta(X_v \mid R_t)$. Consider another masked position $u \in F_t$, $u \neq v$, and let $x_u^\star$ denote its ground-truth value. We perform an oracle intervention: only the missing value at u is revealed, while all other masked positions remain unchanged. We then measure how much the prediction at v

changes using the Kullback–Leibler divergence

$$\Delta_t(v \leftarrow u) = D_{\mathrm{KL}}\left(p_\theta(X_v \mid X_u = x_u^\star, R_t) \,\|\, p_\theta(X_v \mid R_t)\right).$$

A large value of $\Delta_t(v \leftarrow u)$ indicates that the prediction at v is sensitive to the information at u: revealing u substantially changes the model's belief about v, possibly enough to introduce a *top-1 prediction flip*, as in Figure 4. Conversely, a small value indicates that revealing u has little effect on the current prediction.

A target prediction may depend on multiple unresolved positions. We therefore summarize its strongest unresolved dependency as

$$\mu_t(v) = \max_{\substack{u \in F_t \\ u \neq v}} \Delta_t(v \leftarrow u).$$

We refer to $\mu_t(v)$ as the *Pre-Commit score*. A small $\mu_t(v)$ means that the prediction at v is relatively stable to future revelations, whereas a large value indicates that the prediction remains exposed to information that has not yet been resolved.

The ground-truth intervention is deliberate. Revealing the model's own current prediction at u would measure the effect of confirming its existing belief, rather than the effect of receiving the information that is actually missing. In particular, if the model were confidently wrong about u, self-conditioning could reinforce the incorrect hypothesis and conceal a genuine dependency. Ground-truth conditioning instead measures the change in the prediction at v once the correct missing information becomes available.

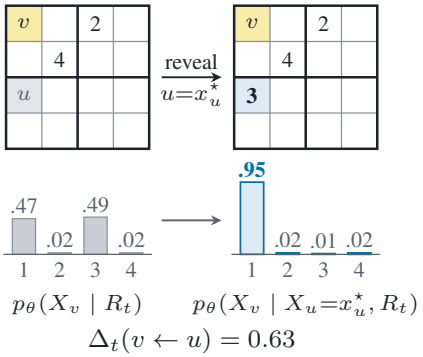


Figure 4: Revealing u at $x_u^\star = 3$ rules out 3 for v, so the top-1 prediction at v flips from 3 to 1. Probabilities are illustrative.

The Pre-Commit score is an oracle quantity: it requires the ground-truth values of masked tokens, which are unavailable at inference time. Its role is to give a principled, model-based definition of unresolved dependency, against which practical dependency estimators can be evaluated. It also defines an *oracle dependency-ready decoding* strategy, which at each denoising step selects the masked position that is least sensitive to any single unresolved token:

$$v_t^\star = \arg\min_{v \in F_t} \mu_t(v).$$

6.2 Large margins can certify dependency-readiness

Although μ_t does not provide a decoding strategy that can be used at test time without token labels, if it is taken as an ideal dependency-ready decoder, it can be useful for analysis. Here, we use it to establish a relation with maximum margin decoding.

Let $q_{t,v}(x)$ denote the probability with which token x is predicted to fill position v at time slot t, and $p_{t,v}^{(u)}(x)$ denote the same probability if the ground truth token at u is also revealed. In this notation,

$$\mu_t(v) = \max_{\substack{u \in F_t \\ u \neq v}} D_{KL}(p_{t,v}^{(u)} || q_{t,v}). \tag{3}$$

Let w_1 and w_2 be the top two tokens predicted to fill position v by $q_{t,v}$, and $m_t(v) = q_{t,v}(w_1) - q_{t,v}(w_2)$ be the margin. We can show that if $m_t(v)$ is sufficiently large, then the decoding of w_1 by maximum margin is not premature, in the sense that if another token were revealed, w_1 would be selected by maximum margin anyway:

Proposition. If $m_t(v) > \sqrt{2\mu_t(v)}$, then $p_{t,v}^{(u)}(w_1) \geq p_{t,v}^{(u)}(x)$ for any masked u and any $x \neq w_1$.

Proof. Let $||p_{t,v}^{(u)} - q_{t,v}||_\infty = \sup_A |p_{t,v}^{(u)}(A) - q_{t,v}(A)|$, where A denotes a measurable event. Then

$$\begin{aligned} p_{t,v}^{(u)}(w_1) - p_{t,v}^{(u)}(x) &= q_{t,v}(w_1) - q_{t,v}(x) + \left(p_{t,v}^{(u)}(w_1) - q_{t,v}(w_1)\right) - \left(p_{t,v}^{(u)}(x) - q_{t,v}(x)\right) \\ &\geq m_t(v) - 2||p_{t,v}^{(u)} - q_{t,v}||_\infty \;\geq\; m_t(v) - \sqrt{2\mu_t(v)} \;\geq\; 0, \end{aligned} \tag{4}$$

where the second inequality is Pinsker's inequality (Gilardoni, 2010), $||p_{t,v}^{(u)} - q_{t,v}||_\infty \leq \sqrt{\mu_t(v)/2}$, and the last follows from our hypothesis. □

In this way, a large margin already implies that the revelation of another masked token will not affect the token selection.

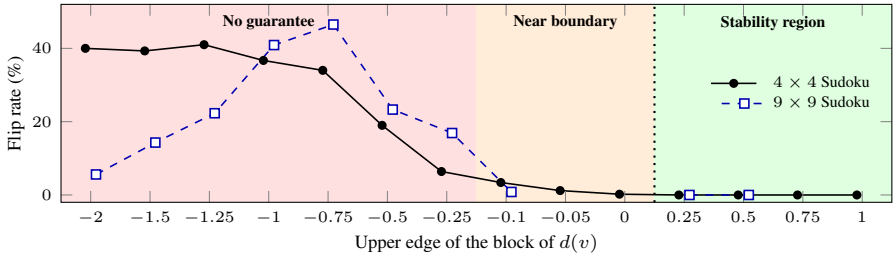


Figure 5: Top-1 flip rate per block of $d(v) = m(v) - \sqrt{2\mu(v)}$ (`max-margin`, exact sweeps). The dotted line marks $d(v) = 0$. On 4×4, the rate falls from about 40% to 0.21% just below the boundary. On 9×9, it peaks at 46.5%. Neither board has a flip above the boundary (0 in 3,777 and 467 reveals). Gaps are blocks with no positions.

Empirical validation. We test this stability condition empirically on 4×4 and 9×9 Sudoku with the LLaDA Instruct model. At every denoising step t and for every masked position v, we compute the margin $m_t(v)$ and $\mu_t(v)$. To compute $\mu_t(v)$, we reveal each other masked position u at its ground-truth token, one at a time, and check whether the top-1 prediction at v changes. As guaranteed by the proposition, the top-1 prediction never changes when the condition holds (0 of 3,777 revelations on 4×4 and 0 of 467 on 9×9). Far below the stability boundary, by contrast, top-1 changes are frequent (Figure 5): for all bins with $m_t(v) - \sqrt{2\mu_t(v)} \leq -0.75$, 34–41% of revelations change the top-1 prediction on 4×4, while on 9×9 the rate reaches 46.5%, nearly half. We note that these low-margin token positions may be unlikely to be chosen by max-margin decoding. However, the results suggest that the proposition leaves room for future work on a non-oracle dependency-ready decoding method, likely one that does not rely on attention, to identify dependency-ready positions that maximum margin may fail to select when the margin is small relative to $\sqrt{2\mu_t(v)}$.

7 Conclusion

Our results cast substantial doubt on the necessity of dependency-ready decoding as a strategy beyond left-to-right, future utility, and confidence-based methods.

Empirically, we generally did not observe that an attention implementation of dependency-ready decoding (DOS) achieved gains over confidence-based or future utility strategies, except on benchmarks where left-to-right decoding was beneficial. We were further discouraged by the left-to-right behavior which resulted from the use of attention as a proxy for dependency at the first layer, even on benchmarks like Sudoku where other kinds of dependencies were expected.

We further gave a theoretical definition of what dependency-ready decoding should ideally do, given oracle labels for each token, without resorting to attention as a proxy for readiness. We showed that maximum-margin decoding already follows this principle whenever the margin is large enough.

Assuming that dependency-ready decoding would even be beneficial for accuracy, these findings leave only a narrow scope for future dependency-ready decoders: their potential benefit is limited to positions with small margins, and reliance on attention should be questioned.

8 Reproducibility Statement

We made several efforts to ensure our results can be reproduced. Our experimental setup describes the models, benchmarks, evaluation harness, EOS Fix, and hardware, as well as the configuration of each decoding strategy, including the attention layers used for DOS (1) and DOS (late). The prompts follow those provided in the public Dream repository[1], and the few-shot settings and generation lengths are reported in the setup and with the main results. We ran DAPD and Info-Gain Sampler with the official code released by their authors, and since no official code was available for DOS, we reimplemented it following Zhou et al. (2026), using the groundedness score defined earlier. All methods share the same prompts, model weights, and decoding budget and reveal one token per step without blocks, so they differ only in how they choose which positions to reveal. LLaDA is decoded with temperature 0, and Dream with temperature 0.1, following the LLaDA and Dream papers. We also give the full setup of our motivating experiment and the definition of the Kendall τ distance we use to compare decoding orders. Finally, we define the Pre-Commit score and oracle dependency decoding, state the assumptions of the Proposition with its complete proof, and describe the stability check we ran.

9 Ethics Statement

All authors adhere to the ICLR Code of Ethics. We study decoding strategies for public MDMs on public benchmarks, involving no human subjects or personal data. All models and datasets follow their original licenses. No new capabilities or risks beyond those of the underlying models and pre-existing decoding methods are introduced.

10 AI Usage Statement

In this work, we used generative AI tools to help refine our hypotheses and experimental design and to help implement our methods, including the decoding strategies and the Pre-Commit score computations. We have not used generative AI tools to interpret results or clean and reformat datasets. Generating synthetic data sets, translation, and qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to help write and edit other code, such as evaluation and plotting scripts, to help create figures, to draft and edit parts of the paper, and to search for relevant literature. We mainly used Claude Code to refine code, OpenAI's GPT models to rephrase and edit text and to search for paper. All AI-assisted code was checked and verified by the authors. The definitions of Pre-Commit score and oracle dependency decoding were developed without AI. The Proposition was suggested by GPT-6 Astra but its proof was written by the authors. All citations were written and checked manually by the authors.

References

Marianne Arriola, Aaron Gokaslan, Justin T. Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=tyEyYT267x.

Hikaru Asano, Tadashi Kozuno, Kuniaki Saito, and Yukino Baba. Where-to-unmask: Ground-truth-guided unmasking order learning for masked diffusion language models. *CoRR*, abs/2602.09501, 2026. doi: 10.48550/ARXIV.2602.09501. URL https://doi.org/10.48550/arXiv.2602.09501.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 17981–

[1] https://github.com/DreamLM/Dream

17993, 2021a. URL https://proceedings.neurips.cc/paper/2021/hash/958c530554f78bcd8e97125b70e6973d-Abstract.html.

Jacob Austin, Augustus Odena, Maxwell I. Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie J. Cai, Michael Terry, Quoc V. Le, and Charles Sutton. Program synthesis with large language models. *CoRR*, abs/2108.07732, 2021b. URL https://arxiv.org/abs/2108.07732.

Heli Ben-Hamu, Itai Gat, Daniel Severo, Niklas Nolte, and Brian Karrer. Accelerated sampling from masked diffusion models via entropy bounded unmasking. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/510e22f4a2da5212a64f1591736e2eaf-Abstract-Conference.html.

Mingyu Cao, Alvaro H. C. Correia, Christos Louizos, Shiwei Liu, and Lu Yin. Search or accelerate: Confidence-switched position beam search for diffusion language models. *CoRR*, abs/2602.10953, 2026. doi: 10.48550/ARXIV.2602.10953. URL https://doi.org/10.48550/arXiv.2602.10953.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *CoRR*, abs/2107.03374, 2021. URL https://arxiv.org/abs/2107.03374.

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. *CoRR*, abs/2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

Gustavo L. Gilardoni. On Pinsker's and Vajda's type inequalities for Csiszár's f -divergences. *IEEE Transactions on Information Theory*, 56(11):5377–5386, 2010. doi: 10.1109/TIT.2010.2068710.

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. *CoRR*, abs/2506.20639, 2025. doi: 10.48550/ARXIV.2506.20639. URL https://doi.org/10.48550/arXiv.2506.20639.

Google DeepMind. Gemini diffusion, 2025. URL https://deepmind.google/models/gemini-diffusion/.

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. In Joaquin Vanschoren and Sai-Kit Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/be83ab3ecd0db773eb2dc1b0a17836a1-Abstract-round2.html.

Pengcheng Huang, Tianming Liu, Zhenghao Liu, Yukun Yan, Shuo Wang, Tong Xiao, Zulong Chen, and Maosong Sun. Empirical analysis of decoding biases in masked 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), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pp. 6853–6876. Association for Computational Linguistics, 2026. doi: 10.18653/V1/2026.ACL-LONG.311. URL https://doi.org/10.18653/v1/2026.acl-long.311.

M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938. ISSN 00063444. URL http://www.jstor.org/stable/2332226.

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. *CoRR*, abs/2506.17298, 2025. doi: 10.48550/ARXIV.2506.17298. URL https://doi.org/10.48550/arXiv.2506.17298.

BumJun Kim, Dongjae Jeon, Moongyu Jeon, and Albert No. DAPD: Dependency-aware parallel decoding via attention for diffusion LLMs. In *Forty-third International Conference on Machine Learning*, 2026a. URL https://openreview.net/forum?id=LZro8iqjiA.

BumJun Kim, Dongjae Jeon, Dueun Kim, Wonje Jeung, and Albert No. Rainbow padding: Mitigating early termination in instruction-tuned diffusion LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=cznTlh7Msz.

Jaeyeon Kim, Kulin Shah, Vasilis Kontonis, Sham M. Kakade, and Sitan Chen. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025a. URL https://proceedings.mlr.press/v267/kim25ah.html.

Seo Hyun Kim, Sunwoo Hong, Hojung Jung, Youngrok Park, and Se-Young Yun. KLASS: kl-guided fast inference in masked diffusion models. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025b. URL http://papers.nips.cc/paper_files/paper/2025/hash/8514c1f12dcdd061ac8521feb78f28ca-Abstract-Conference.html.

Pengxiang Li, Yefan Zhou, Dilxat Muhtar, Lu Yin, Shilin Yan, Li Shen, Yi Liang, Soroush Vosoughi, and Shiwei Liu. Diffusion language model knows the answer before it decodes. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=g88nt4ieTG.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32819–32848. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/lou24a.html.

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 Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/48b383b24230e0e6e649d9c98dae4d8c-Abstract-Conference.html.

Shen Nie, Qiyang Min, Shaoxuan Xu, Zihao Huang, Yuxuan Song, Yong Shan, Yankai Lin, Wayne Xin Zhao, Chongxuan Li, and Ji-Rong Wen. Improved large language diffusion models. *CoRR*, abs/2606.25331, 2026. doi: 10.48550/ARXIV.2606.25331. URL https://doi.org/10.48550/arXiv.2606.25331.

Yusuf Sahin, Ahmed Rockey Saikia, Volkan Cevher, and Paolo Favaro. Attention-discounted adaptive sampler for masked diffusion language models. *CoRR*, abs/2606.10829, 2026. doi: 10.48550/ARXIV.2606.10829. URL https://doi.org/10.48550/arXiv.2606.10829.

Subham S. Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander M. Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/eb0b13cc515724ab8015bc978fdde0ad-Abstract-Conference.html.

Kulin Shah, Nishanth Dikkala, Xin Wang, and Rina Panigrahy. Causal language modeling can elicit search and reasoning capabilities on logic puzzles. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/67b31ca159553d5593e62d7b998d63ea-Abstract-Conference.html.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/bad233b9849f019aead5e5cc60cef70f-Abstract-Conference.html.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2023.127063. URL https://www.sciencedirect.com/science/article/pii/S0925231223011864.

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. *CoRR*, abs/2505.22618, 2025. doi: 10.48550/ARXIV.2505.22618. URL https://doi.org/10.48550/arXiv.2505.22618.

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. *CoRR*, abs/2509.01142, 2025. doi: 10.48550/ARXIV.2509.01142. URL https://doi.org/10.48550/arXiv.2509.01142.

Kaisen Yang, Jayden Teoh, Kaicheng Yang, Yitong Zhang, and Alex Lamb. Improving sampling for masked diffusion models via information gain. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=1HPTFpwktA.

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 *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025a. URL https://openreview.net/forum?id=NRYgUzSPZz.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models. *CoRR*, abs/2508.15487, 2025b. doi: 10.48550/ARXIV.2508.15487. URL https://doi.org/10.48550/arXiv.2508.15487.

Huaixiu Steven Zheng, Swaroop Mishra, Hugh Zhang, Xinyun Chen, Minmin Chen, Azade Nova, Le Hou, Heng-Tze Cheng, Quoc V. Le, Ed H. Chi, and Denny Zhou. NATURAL PLAN: benchmarking llms on natural language planning. *CoRR*, abs/2406.04520, 2024. doi: 10.48550/ARXIV.2406.04520. URL https://doi.org/10.48550/arXiv.2406.04520.

Xueyu Zhou, Yangrong Hu, and Jian Huang. DOS: dependency-oriented sampler for masked diffusion language models. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pp. 17404–17419. Association for Computational Linguistics, 2026. doi: 10.18653/V1/2026.FINDINGS-ACL.861. URL https://doi.org/10.18653/v1/2026.findings-acl.861.