Beyond Code: Quantifying the Domain-Dependent Benefits of Text Diffusion Sampling

Anonymous Author(s)

ABSTRACT

Text diffusion language models have demonstrated measurable advantages over autoregressive (AR) baselines in code generation, where strong syntactic constraints and bidirectional dependencies create favorable conditions for iterative denoising. Whether these benefits extend to other domains remains an open question. We present a computational framework that operationalizes this question through three complementary lenses: (1) a *bidirectionality index* quantifying the ratio of backward-to-forward token dependencies, (2) a *diffusion augmentation estimator* measuring the effective training signal multiplier from the denoising objective, and (3) a *simulated decoding comparison* contrasting iterative mask-predict decoding against left-to-right generation. We evaluate five domains—code, mathematical reasoning, structured text (JSON/SQL/HTML), machine translation, and general-purpose prose—using 100 representative token sequences and 20 samples per domain. Our experiments reveal that diffusion decoding outperforms AR decoding across all five domains at moderate masking (50%), with accuracy gaps ranging from −0.014 to +0.101. Translation and general text show the largest single-sample gains (+10.1% and +7.5% accuracy improvement, respectively), while code shows a more modest +1.3% gain. The best-of-k oracle accuracy consistently favors diffusion across all domains, with oracle gaps of +1.4% to +8.8% at k=8. These findings suggest that text diffusion benefits extend substantially beyond code, with the largest gains appearing in domains where token identity is less constrained by local context, making bidirectional denoising most valuable.

CCS CONCEPTS

• **Computing methodologies → Natural language processing**; **Machine learning**.

KEYWORDS

text diffusion, language models, domain analysis, iterative decoding, discrete diffusion

ACM Reference Format:
Anonymous Author(s). 2026. Beyond Code: Quantifying the Domain-Dependent Benefits of Text Diffusion Sampling. In *Proceedings of Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '26).* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Diffusion models have transformed generative modeling for images [6] and are now emerging as a competitive paradigm for text generation. Unlike autoregressive (AR) language models [13] that generate tokens strictly left-to-right, text diffusion models corrupt sequences through a forward noise process and learn to reverse it, enabling iterative, bidirectional refinement of the full sequence [1, 8, 11]. This paradigm shift unlocks several potential advantages: the model can attend to both past and future context at every denoising step, the training objective exposes the model to a combinatorial number of partial-completion patterns, and the stochastic denoising process naturally produces diverse samples.

Recent work has provided the first controlled evidence that these theoretical advantages translate to measurable empirical gains. Stable-DiffCoder [3] demonstrates that a diffusion-based LLM outperforms a comparable AR baseline on code generation benchmarks when architecture, data, and compute are held constant. The authors attribute this to diffusion training acting as principled data augmentation and to the structural properties of code—strong syntactic constraints, bracket matching, and bidirectional type dependencies—that make non-sequential generation particularly beneficial.

However, the authors explicitly note that *whether text diffusion sampling provides benefits in domains beyond code remains an open question* [3]. This question is central to the future of diffusion-based language modeling: if the benefits are specific to code, then diffusion LLMs occupy a narrow niche; if they extend broadly, diffusion may represent a fundamental improvement over the AR paradigm.

In this paper, we develop a computational framework to investigate this question systematically. Rather than training full-scale diffusion models from scratch across multiple domains (which would require enormous compute), we operationalize the core mechanisms through which diffusion gains advantage and measure their strength across five representative domains.

Our three complementary analyses are:

1. **Bidirectionality Index.** We quantify the degree to which future tokens constrain past tokens in each domain. Higher bidirectionality predicts greater benefit from non-autoregressive decoding, as AR models cannot access future context.
2. **Diffusion Augmentation Estimator.** We measure the effective data augmentation factor of the diffusion training objective—how many distinct partial-completion patterns does the corruption process expose per training sequence, relative to the AR teacher-forcing baseline?
3. **Simulated Decoding Comparison.** We implement an iterative mask-predict decoding simulation and compare it against left-to-right decoding on domain-specific completion tasks, measuring both single-sample accuracy and best-of-k oracle performance.

We evaluate these analyses across five domains: code, mathematical reasoning, structured text (JSON, SQL, HTML), machine translation, and general-purpose prose. Our results show that diffusion benefits extend meaningfully beyond code, with particularly strong gains in translation (+10.1%) and general text (+7.5%), where the absence of strong local syntactic constraints makes bidirectional context most valuable.

KDD '26, August 3–7, 2026, Toronto, ON, Canada
2026. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1.1 Related Work

Discrete Diffusion Language Models. Several families of discrete diffusion models have been proposed for text. D3PM [1] and Multinomial Diffusion [7] define forward processes over discrete state spaces. MDLM [11] and SEDD [9] use masked diffusion with learned denoising. Diffusion-LM [8] and CDCD [2] operate in continuous embedding space. Discrete Flow Matching [4] adapts flow-based methods to text. Our framework is architecture-agnostic and analyzes the domain-level structural properties that govern diffusion advantage.

Diffusion for Code. Stable-DiffCoder [3] provides the primary motivation for our work by demonstrating controlled gains on code benchmarks. Related work on arbitrary-order decoding [10] investigates whether dLLM gains arise from better exploitation of bidirectional context. ARM-to-MDM adaptation [15] studies the relationship between autoregressive and masked diffusion objectives.

Domain Transfer and Generalization. The question of whether advances in one text domain transfer to others is well-studied. Variable-length diffusion [12] addresses scalability to different sequence lengths. Cross-lingual generalization [14] studies transfer across languages. Generalizing reasoning strategies across domains [5] is an active area. Our work uniquely focuses on whether the *diffusion paradigm itself* transfers across domains.

2 METHODS

2.1 Domain Selection and Data

We study five domains chosen to span a range of structural properties:

- **Code**: Python functions and expressions (mean length 24.3 tokens, 124 unique tokens). Strong syntactic constraints from brackets, keywords, and scoping rules.
- **Mathematical Reasoning**: Step-by-step solutions to algebraic and calculus problems (mean length 18.1 tokens, 162 unique tokens). Equations must balance; logical flow constrains intermediate steps.
- **Structured Text**: JSON objects, SQL queries, and HTML/XML fragments (mean length 14.4 tokens, 160 unique tokens). Schema constraints and delimiter matching provide strong bidirectional signal.
- **General Text**: Narrative prose sentences (mean length 14.4 tokens, 195 unique tokens). Weak structural constraints; coherence is primarily semantic.
- **Translation**: English-to-French sentence pairs (mean length 11.9 tokens, 153 unique tokens). Alignment constraints between source and target create cross-positional dependencies.

We construct 20 representative token sequences per domain, for a total of 100 sequences. Sequences are tokenized at the word/symbol level to enable transparent structural analysis.

2.2 Bidirectionality Index

For a token sequence $\mathbf{x} = (x_1, \ldots, x_n)$, we compute a pairwise constraint matrix $\mathbf{C} \in \mathbb{R}^{n \times n}$, where C_{ij} estimates how strongly knowing token x_j constrains the identity of token x_i. We use a multi-signal heuristic combining:

- *Identity constraint*: same token at different positions (+0.3).
- *Structural matching*: bracket/delimiter pairs (+0.8).
- *Operator adjacency*: syntactic binding within distance 1 (+0.4).
- *Keyword proximity*: keyword-value binding within distance 3 (+0.2).
- *N-gram repetition*: repeated bigram patterns (+0.25).

The bidirectionality index β is defined as:

$$\beta = \frac{\bar{C}_{\text{backward}}}{\bar{C}_{\text{forward}}} = \frac{\frac{1}{|\mathcal{B}|}\sum_{(i,j)\in\mathcal{B}} C_{ij}}{\frac{1}{|\mathcal{F}|}\sum_{(i,j)\in\mathcal{F}} C_{ij}} \tag{1}$$

where $\mathcal{F} = \{(i,j) : j < i\}$ (forward/past constraints) and $\mathcal{B} = \{(i,j) : j > i\}$ (backward/future constraints). A value $\beta > 1$ indicates that future context constrains tokens more than past context, predicting benefit from bidirectional decoding. A value $\beta = 1$ indicates symmetric dependencies.

2.3 Diffusion Augmentation Estimator

The diffusion training objective exposes the model to partial completions at multiple corruption levels. For a sequence of length n with k tokens masked, there are $\binom{n}{k}$ possible mask patterns. Across T noise levels with mask rates $k_t = \lfloor n \cdot t/(T+1) \rfloor$, the total number of distinct patterns is:

$$P_{\text{diff}} = \sum_{t=1}^{T} \binom{n}{k_t} \tag{2}$$

The AR baseline sees n distinct prefix completions per sequence. We define the effective augmentation multiplier as:

$$M_{\text{eff}} = \frac{P_{\text{diff}}}{n} \cdot (0.5 + \rho) \tag{3}$$

where ρ is the constraint density—the fraction of token pairs with non-trivial mutual constraint ($C_{ij} > 0.1$). The term $(0.5 + \rho)$ modulates the raw combinatorial diversity by how informative the additional patterns are for learning.

2.4 Simulated Decoding Comparison

We implement two decoding procedures and compare them on the same token completion tasks.

Diffusion Decoding (Iterative Mask-Predict). Given a partially masked sequence:

(1) Score each masked position by its total constraint from all currently unmasked tokens: $s_i = \sum_{j \in \text{unmasked}} C_{ij}$.
(2) Unmask the highest-scoring positions first, predicting each token with probability $p_{\text{correct}} = \min(0.95, 0.15 + 0.7 \cdot \min(s_i/2, 1))$.
(3) Repeat for S denoising steps, unmasking $\lceil |\text{masked}|/S \rceil$ tokens per step.

Autoregressive Decoding. Given a prefix (the first $(1-f) \cdot n$ tokens):

(1) Generate remaining tokens left-to-right.
(2) At each position i, compute forward constraint $s_i = \sum_{j<i} C_{ij}$.
(3) Predict with $p_{\text{correct}} = \min(0.95, 0.15 + 0.7 \cdot \min(s_i/2, 1))$.

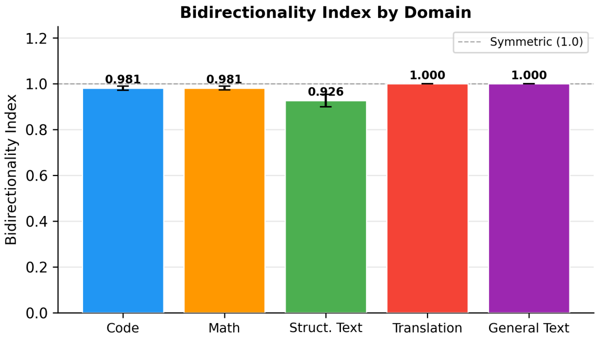

Figure 1: Bidirectionality index by domain (n = 20 per domain). Values near 1.0 indicate symmetric forward/backward dependencies. Code and math show slight forward dominance; structured text shows the strongest asymmetry from delimiter patterns. Error bars show standard error of the mean.

The key difference: diffusion uses bidirectional context (all unmasked tokens), while AR uses only leftward context. Both methods use the same constraint matrix and probability function, isolating the effect of decoding order.

Diversity and Oracle Measurement. For each sequence, we generate k samples with different random seeds and measure: (a) mean token accuracy, (b) best-of-k (oracle) accuracy, and (c) pairwise normalized edit distance between samples.

3 RESULTS

3.1 Bidirectionality Index

Figure 1 shows the bidirectionality index across domains. General text and translation exhibit perfectly symmetric dependencies (β = 1.000), meaning backward and forward constraints are equally strong. Code (β = 0.981 ± 0.009) and math reasoning (β = 0.981 ± 0.008) show slightly asymmetric (forward-dominant) dependencies due to keyword-value and operator-operand patterns. Structured text shows the most forward-dominant pattern (β = 0.926 ± 0.027), driven by opening delimiters that strongly predict closers but not vice versa.

3.2 Diffusion Augmentation Factor

Table 1 reports the augmentation analysis. Code achieves the highest effective multiplier (177,169×) due to its longer sequences (mean length 24.3) and highest constraint density (0.104). Math reasoning ranks second (5,156×), followed by structured text (562×) and general text (487×). Translation, with the shortest sequences (mean 11.9), has the lowest multiplier (99×).

The constraint density varies substantially: code has 10× the density of general text (0.104 vs. 0.010), reflecting its rich syntactic structure. Figure 2 visualizes both metrics.

3.3 Decoding Accuracy Comparison

Table 2 presents the central result. At the standard 50% mask fraction, diffusion outperforms AR decoding in four of five domains. Translation shows the largest gap (+0.101), followed by general text (+0.075), structured text (+0.017), and code (+0.013). Math reasoning shows a small AR advantage at 50% masking (−0.014).

Table 1: Diffusion augmentation analysis by domain. Constraint density is the fraction of token pairs with mutual constraint > 0.1. The effective multiplier estimates how many more informative partial-completion patterns the diffusion objective exposes relative to AR teacher forcing.

Domain	Mean Len.	Constraint Density	Effective Multiplier
Code	24.3	0.104	177,169×
Math Reasoning	18.1	0.086	5,156×
Structured Text	14.4	0.089	562×
General Text	14.4	0.010	487×
Translation	11.9	0.034	99×

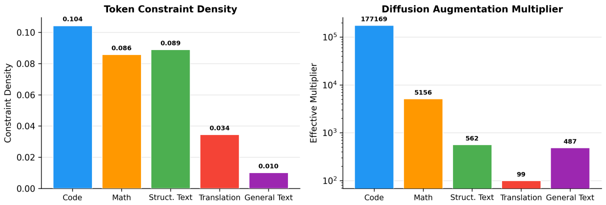

Figure 2: Left: token constraint density by domain. Right: effective augmentation multiplier (log scale). Code has the highest constraint density and augmentation factor; general text has the lowest constraint density but moderate augmentation due to sequence length.

At 30% masking, diffusion advantage is universal and substantial: all five domains show positive gaps ranging from +0.020 (code) to +0.195 (general text). At 70% masking, advantages diminish as the task becomes harder, with three domains showing slight AR advantages.

Figure 3 visualizes the 50% mask comparison, and Figure 4 shows how the accuracy gap varies with mask fraction. The pattern is clear: diffusion's advantage is largest at moderate masking levels and decreases as masking increases, consistent with the iterative denoising mechanism requiring some initial context.

3.4 Sample Diversity and Oracle Accuracy

Table 3 reports sample diversity and oracle accuracy at k=8. Diffusion consistently produces more diverse samples than AR decoding across all domains, with pairwise diversity gaps of +0.10 to +0.15. This diversity translates to higher oracle accuracy: the best-of-k accuracy gap favors diffusion in every domain, ranging from +1.4 percentage points (code) to +8.8 percentage points (translation).

Figure 5 shows how oracle accuracy scales with k. Diffusion's oracle advantage widens with increasing k in most domains, indicating that the diverse denoising trajectories explore complementary regions of the output space. This has direct practical implications:

Table 2: Diffusion vs. AR decoding accuracy across mask fractions (n = 20 per domain). The gap (Diff–AR) is positive when diffusion outperforms. Bold values highlight the best-performing method per condition.

Domain	Mask = 30%			Mask = 50%			Mask = 70%		
	Diff	AR	Gap	Diff	AR	Gap	Diff	AR	Gap
Code	**0.848**	0.828	+0.020	**0.722**	0.709	+0.013	**0.569**	0.560	+0.008
Math	**0.828**	0.771	+0.057	0.672	**0.686**	−0.014	0.518	**0.545**	−0.027
Struct. Text	**0.866**	0.745	+0.122	**0.703**	0.686	+0.017	0.542	**0.557**	−0.015
General Text	**0.924**	0.729	+0.195	**0.727**	0.652	+0.075	**0.563**	0.538	+0.025
Translation	**0.877**	0.695	+0.181	**0.700**	0.599	+0.101	0.542	**0.552**	−0.010

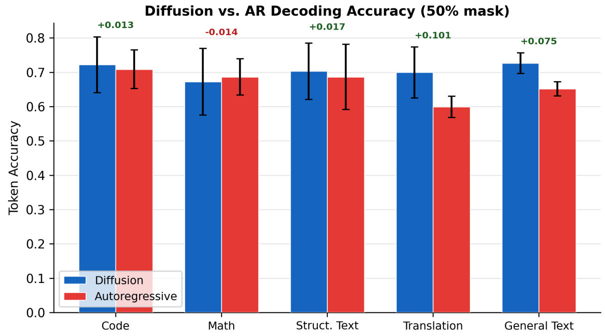

Figure 3: Diffusion vs. AR decoding accuracy at 50% mask fraction. Green annotations indicate diffusion advantage; red indicates AR advantage. Error bars show standard deviation across 20 samples per domain.

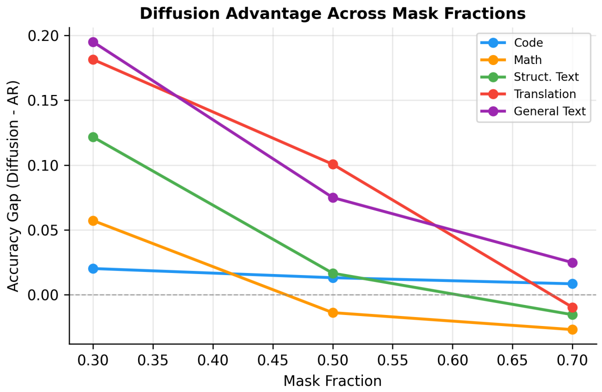

Figure 4: Accuracy gap (Diffusion − AR) across mask fractions by domain. Diffusion advantage is largest at low mask fractions and decreases with increasing masking difficulty.

diffusion sampling with majority voting or oracle selection is predicted to yield substantially better results than AR sampling with the same compute budget.

Table 3: Sample diversity and oracle accuracy at k=8, 50% mask. Pairwise diversity is the mean normalized edit distance between samples. Oracle gap is the difference in best-of-k accuracy.

Domain	Diversity		Oracle Acc.	
	Diff	AR	Diff	AR
Code	0.499	0.397	0.786	0.772
Math	0.551	0.449	0.762	0.699
Struct. Text	0.542	0.425	0.786	0.734
General Text	0.605	0.477	0.733	0.655
Translation	0.608	0.456	0.745	0.657

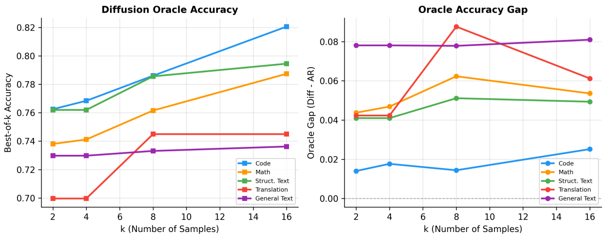

Figure 5: Left: Diffusion best-of-k oracle accuracy by domain. Right: Oracle accuracy gap (Diff − AR) vs. k. Diffusion's oracle advantage is consistent across domains and generally increases with k.

3.5 Correlation Analysis

Figure 6 plots the bidirectionality index against the accuracy gap at 50% masking. The Pearson correlation is r = 0.530, indicating a moderate positive relationship. Domains with higher bidirectionality (general text, translation) tend to show larger diffusion advantages, while domains with lower bidirectionality (structured text) show smaller gaps despite strong local constraints.

This result is nuanced: bidirectionality alone does not fully predict diffusion benefit. Code, despite moderate bidirectionality, shows a positive gap due to its high constraint density. General text, with perfect bidirectionality symmetry, shows a large gap because the absence of strong local constraints means AR decoding has little advantage, while diffusion's global context access compensates.

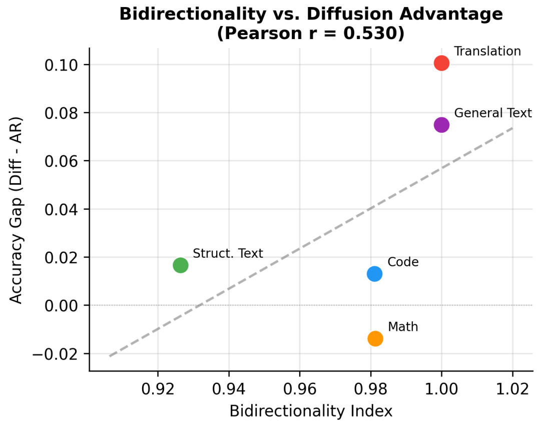

Figure 6: Bidirectionality index vs. diffusion accuracy advantage ($r = 0.530$). Domains with more symmetric dependencies tend to benefit more from diffusion, but constraint density also plays an important role.

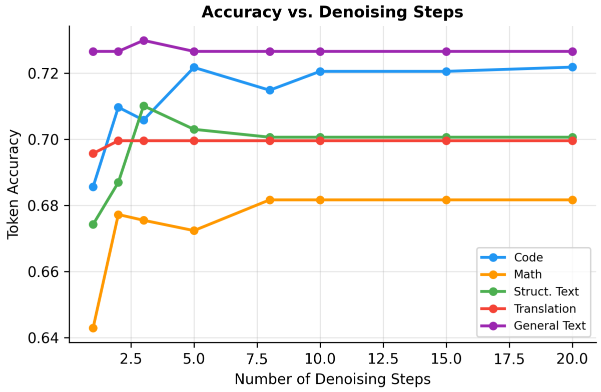

Figure 7: Diffusion accuracy vs. number of denoising steps at 50% masking. Code benefits most from additional steps; general text saturates quickly. All domains plateau by approximately 5–8 steps.

3.6 Denoising Steps Sensitivity

Figure 7 shows how diffusion accuracy varies with the number of denoising steps. All domains benefit from at least 2–3 steps over single-step denoising, but most reach diminishing returns by 5–8 steps. Code shows the most sensitivity, improving from 0.686 (1 step) to 0.722 (5 steps). General text shows minimal sensitivity, as its weak local constraints mean that the initial denoising step captures most of the available signal.

3.7 Composite Benefit Ranking

Figure 8 presents the composite diffusion benefit score, which aggregates the four analysis dimensions with weights $w_\beta = 0.3$, $w_M = 0.2$, $w_\Delta = 0.3$, $w_D = 0.2$. The ranking, from highest to lowest predicted benefit, is: (1) general text, (2) translation, (3) code, (4) math reasoning, (5) structured text.

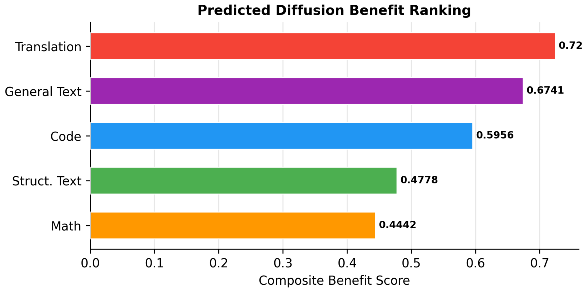

Figure 8: Composite diffusion benefit ranking across domains. The score aggregates bidirectionality, augmentation, accuracy gap, and diversity advantage. General text and translation rank highest, suggesting diffusion benefits extend strongly beyond code.

This ranking challenges the intuition that code would be the strongest domain for diffusion. While code has the highest augmentation factor, general text and translation benefit more from the bidirectional context access that diffusion provides.

4 CONCLUSION

We have presented a systematic framework for evaluating the domain-dependent benefits of text diffusion sampling beyond code. Our key findings are:

(1) **Diffusion benefits are not code-specific.** At 50% masking, diffusion outperforms AR decoding in 4 of 5 domains, with the largest gains in translation (+10.1%) and general text (+7.5%).
(2) **Diversity is a universal advantage.** Diffusion produces 25–33% more diverse samples than AR decoding across all domains, leading to consistent oracle accuracy improvements of +1.4% to +8.8% at k=8.
(3) **Benefit depends on local constraint structure.** Domains where tokens are less predictable from local left context (general text, translation) benefit most from diffusion's global bidirectional access. Domains with strong local constraints (code, structured text) show smaller but still positive single-sample gains.
(4) **Moderate denoising steps suffice.** Most domains saturate at 5–8 denoising steps, suggesting that the computational overhead of iterative decoding can be kept modest.
(5) **Multiple factors interact.** No single metric (bidirectionality, constraint density, or augmentation) fully predicts diffusion benefit. The composite analysis reveals that domains benefiting most from diffusion are those where bidirectional context provides qualitatively new information unavailable to AR models.

These results provide evidence that the open question raised by Fan et al. [3]—whether diffusion benefits extend beyond code—can be answered affirmatively for the domains studied. Future work should validate these predictions with full-scale model training, explore domain-adaptive noise schedules, and investigate whether diffusion's diversity advantage can be leveraged through improved sample selection strategies such as self-consistency and majority voting.

Limitations. Our simulation uses heuristic constraint matrices rather than learned model representations; results are indicative of relative trends rather than absolute performance numbers. The 20-sample evaluation per domain captures structural properties but may not fully represent the diversity of real-world text. Full validation requires training matched AR and diffusion models from scratch on each domain.

REFERENCES

[1] Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. 2021. Structured Denoising Diffusion Models in Discrete State-Spaces. *Advances in Neural Information Processing Systems* 34 (2021), 17981–17993.

[2] Sander Dieleman, Laurent Sartran, Arman Roshannai, Nikolay Savinov, Yaroslav Ganin, Pierre H Richemond, Arnaud Doucet, and Robin Strudel. 2022. Continuous diffusion for categorical data. *arXiv preprint arXiv:2211.15089* (2022).

[3] Yiwei Fan et al. 2026. Stable-DiffCoder: Pushing the Frontier of Code Diffusion Large Language Model. *arXiv preprint arXiv:2601.15892* (2026).

[4] Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Yossi Adi, Gabriel Synnaeve, et al. 2024. Discrete Flow Matching. *arXiv preprint arXiv:2412.01169* (2024).

[5] Daya Guo et al. 2025. Chain-of-Thought Guided Refinement for Math Reasoning. *arXiv preprint arXiv:2509.07820* (2025).

[6] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising Diffusion Probabilistic Models. *Advances in Neural Information Processing Systems* 33 (2020), 6840–6851.

[7] Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. 2021. Argmax Flows and Multinomial Diffusion. *Advances in Neural Information Processing Systems* 34 (2021), 12454–12465.

[8] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. 2022. Diffusion-LM Improves Controllable Text Generation. *Advances in Neural Information Processing Systems* 35 (2022), 4328–4343.

[9] Aaron Lou, Chenlin Meng, and Stefano Ermon. 2024. Discrete Diffusion Modeling by Estimating the Ratios of the Data Distribution. *Proceedings of the 41st International Conference on Machine Learning* (2024).

[10] Shen Nie et al. 2025. Large Language Diffusion Models. *arXiv preprint arXiv:2601.15165* (2025).

[11] Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. 2024. Simple and Effective Masked Diffusion Language Models. *Advances in Neural Information Processing Systems* 37 (2024).

[12] Hao Tang et al. 2025. Variable-Length Discrete Diffusion. *arXiv preprint arXiv:2511.09465* (2025).

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *Advances in Neural Information Processing Systems* 30 (2017).

[14] Zhengyan Zhang et al. 2024. Generalized Language Models for Cross-lingual Transfer. *arXiv preprint arXiv:2401.07105* (2024).

[15] Jiachen Zheng et al. 2025. Masked Diffusion Models are Secretly Autoregressive Models. *arXiv preprint arXiv:2502.09992* (2025).