THE EFFICIENCY GAP IN BYTE MODELING

Celine Lee[♠] **Jing Nathan Yan**[†] **Chen Liang**[†] **Jiaxin Shi**[†] **Yin Zhang**[†]
Jeremiah Liu[†] **Pengcheng Yin**[†] **Ed Chi**[†] **Fernando Pereira**[†] **Derek Cheng**[†]
Alexander M. Rush[♠] **Ruoxi Wang**[†]
[♠] Cornell University [†] Google DeepMind
cl923@cornell.edu

ABSTRACT

Modern language models have historically relied on two dominant design choices: subword tokenization and autoregressive (AR) ordering. Recently, there has been significant research interest in moving toward byte-level modeling to bypass domain-specific vocabularies, as well as masked diffusion models (MDM) to enable parallel non-sequential generation. Intuitively, the intersection of these paradigms represents a generative ideal: a modality-agnostic system capable of fine-grained any-order generation. However, the computational interaction between these granular representations and non-sequential objectives remains under-explored. In this work, we investigate the viability of this combination through a compute-matched scaling study. We observe a structural dichotomy: AR models on bytes effectively amortize the cost of tokenization, naturally rediscovering sub-word segmentation at scale. In contrast, byte-level MDMs demand disproportionately more compute to match their BPE counterparts at the compute scales studied and our isoFLOPs studies suggest that they may reach parity only at much higher compute scales. We attribute this disparity to the masking objective, which shatters the local contiguity required to resolve sub-word semantics from bytes, whereas AR's stable causal history preserves these local dependencies. Our findings inform the community of a critical efficiency tradeoff, suggesting that future modality-agnostic designs should address this context fragility to maintain efficient scaling trajectories.

1 INTRODUCTION

While rapid scaling has unlocked remarkable capabilities in large language models (LLMs) (Anthropic, 2024; OpenAI et al., 2024; Comanici et al., 2025; xAI, 2025), the standard recipe remains anchored to subword tokenization and autoregressive (AR) ordering. These priors impose fundamental constraints: fixed compression via Byte-Pair Encoding (BPE) (Sennrich et al., 2016) limits generalization to non-lexical or out-of-distribution modalities (Xue et al., 2022), while the unidirectional AR objective weakens parallel generation and look-ahead planning (Nie et al., 2025). To overcome this, the field is exploring **byte-level modeling** for universality (YU et al., 2023; Wang et al., 2024; Hwang et al., 2025; Pagnoni et al., 2024), and **masked diffusion models (MDMs)** for order-agnostic inference (Shi et al., 2024; Sahoo et al., 2024).

In this work, we investigate the viability of combining these frontiers through a compute-controlled study across the cross-product of modeling objectives (AR vs. MDM) and tokenization strategies (Byte vs. BPE). We find that byte modeling introduces a overhead distinct from the transformers' quadratic attention cost: the model must expend compute to rediscover the subword structures that BPE provides for free. Our study reveals that this efficiency gap is objective-dependent. We observe a stark disparity: while AR byte models approach performance parity with their BPE counterparts as compute scales, byte-level MDMs exhibit a persistent and significantly steeper efficiency penalty.

We first quantify the efficiency gap and demonstrate that the computational overhead of byte-level modeling is not uniform. AR models on raw bytes are data-efficient, effectively rediscovering BPE-like segmentation through stable causal history. In contrast, MDMs suffer a collapse in efficiency when applied to raw bytes.

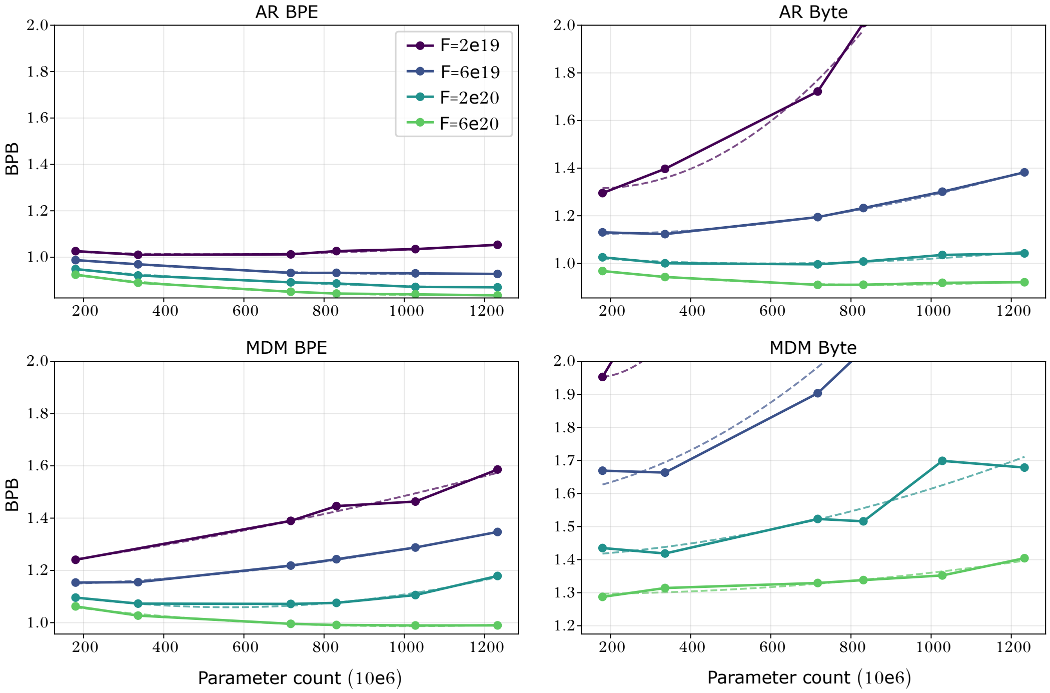


Figure 1: **Scaling Trends across Objectives and Tokenization.** BPB performance for AR and MDM from 180M to 1.2B non-embedding parameters. Curves represent training budgets ($F = 2 \times 10^{19}$ to $F = 6 \times 10^{20}$ FLOPs), with dotted parabolas marking efficiency frontiers. AR Byte models converge toward BPE counterparts at scale, whereas MDM Byte models exhibit a persistent performance offset.

Second, we provide a structural mechanism analysis to identify the root cause of the masked diffusion modeling failure: the masking objective inherently destroys the local contiguity essential for resolving ambiguity in raw byte streams. Our analysis shows that resolving sub-word semantics from raw bytes requires stable local context. The MDM masking objective shatters local contiguity, depriving the model of the structural cues necessary to form semantic chunks. Through permutation experiments, we confirm that while byte models are uniquely fragile when deprived of the non-causal context typical of MDM masking.

Finally, we analyze the scaling trajectories and find that the additional modeling demands for AR models is eventually amortized by scaling. Conversely, the scaling requirement for byte-level MDMs is significantly steeper. This reveals a critical efficiency tradeoff: while AR ordering naturally compensates for the lack of tokenization, standard MDMs fails to resolve granular byte dependencies. These findings provide a structural signal to the community: achieving viable scaling in the raw byte regime requires moving beyond simple compute increases toward architectural adaptations.

2 Related Work

Byte-Level and Diffusion Language Models. Universal, tokenizer-free language modeling seeks to bypass domain-specific vocabularies using raw UTF-8 bytes. Prior research addresses the extreme sequence lengths of byte streams through architectural innovations such as hierarchical patching (Pagnoni et al., 2024; YU et al., 2023), linear-time backbones (Wang et al., 2024), or hybrid approaches (Hwang et al., 2025). While such specialized architectures are designed to handle raw byte streams, we utilize a standard Transformer backbone to isolate the interaction between modeling objective and data representation without introducing confounding architectural variables. Conversely, discrete diffusion models (Austin et al., 2023) like MDLM (Sahoo et al., 2024) and MD4 (Shi et al., 2024) offer order-agnostic generation but rely heavily on subword tokenization. While small-scale character-level experiments exist (e.g. on text8 (Austin et al., 2023)), they lack the systematic scaling analysis required to understand how non-causal objectives resolve dependencies in raw byte streams.

We demonstrate that the viability of bypassing tokenization is fundamentally contingent on the modeling objective itself.

Scaling and Vocabulary. Performance characterization via compute-optimal scaling laws is well-established (Hoffmann et al., 2022). While recent works show MDMs improve on perplexity at a rate comparable to AR models (Nie et al., 2025), they maintain a persistent computational gap of approximately $16\times$ that of AR models, and may require higher parameters-to-data ratios at scale (von Rütte et al., 2025). Scaling laws for vocabulary (Tao et al., 2024) further suggest that larger semantic units optimizer performance in larger models. Smaller vocabularies have been noted to increase MDM denoising complexity due to the resulting increase in long-range dependencies (Sahoo et al., 2024). Our study reveals that token representation interacts with the objective beyond simple context length: the order-agnostic nature of MDMs makes them uniquely dependent on the pre-composed semantic scaffolding provided by subword tokens.

3 Background

3.1 Language Modeling Objectives

Language modeling is the task of learning the probability distribution of sequences $x = (x_1, x_2, ...x_L)$. The choice of factorization and generation order distinguishes autoregressive (AR) models from masked diffusion models (MDMs).

Autoregressive models factorize the joint probability as a product of conditional probabilities using a fixed left-to-right order: $p_\theta(x) = \prod_{i=1}^{L} p_\theta(x_i|x_{<i})$, parameterized by model θ. This formulation enforces a strict sequential dependency, where the model attends only to past tokens when predicting the next token.

Unlike AR models, **masked diffusion models** are inherently bidirectional and order-agnostic. We follow the continuous-time discrete diffusion framework (Austin et al., 2023; Shi et al., 2024; Sahoo et al., 2024), where a forward process independently replaces tokens in x_0 with `[MASK]` based on a retention schedule α_t. The model learns to reverse this by minimizing the weighted objective $\mathcal{L}_{\text{MDM}} = \mathbb{E}_{t,x_0,x_t}[\frac{\alpha'_t}{1-\alpha_t} \sum_{i:x_t^{(i)}=\texttt{[MASK]}} \log p_\theta(x_0^{(i)}|x_t, t)]$, which effectively prioritizes mostly-unmasked states. Inference entails iterative unmasking.

Crucially, whereas AR models rely on a stable causal history, MDMs must resolve semantics from unstable contexts across combinatorial orderings. This distinction is central to our investigation into how byte-level representations, which lack pre-composed BPE units, affect scaling across paradigms. More details can be found in Appendix A.

3.2 BPE Compression.

Byte-Pair Encoding (BPE) (Gage, 1994; Sennrich et al., 2016) iteratively merges the most frequent adjacent pairs of bytes into new tokens, building a vocabulary V of subword units. This process compresses the raw data: a sequence of bytes is represented by fewer BPE tokens.

This compression has two primary effects: (1) **computational efficiency**, as the attention mechanism scales quadratically with sequence length L and (2) **semantic density**. BPE tokens correspond to common sub-word structures, meaning each individual token carries higher information content. In contrast, byte-level modeling ($V = 256$) involves no compression, resulting in longer sequences where individual units (bytes) carry minimal semantic value in isolation.

4 Experimental Setup

To isolate the interaction between data representation (Byte vs. BPE) and modeling objective (AR vs. MDM), we employ a compute-matched evaluation protocol (Hoffmann et al., 2022): that ensures that comparisons are grounded in total floating-point operations (FLOPs) expended, accounting for the inherent computational imbalances between byte-level and subword-level processing.

4.1 Data

All models are trained on the Slimpajama-627B dataset (Soboleva et al., 2023). We compare two distinct input representations: **(1) Byte-Level:** we bypass standard text tokenizers entirely, mapping raw UTF-8 bytes directly to a vocabulary of size $V = 256$; **(2) BPE-Level:** we use the standard Llama 2 BPE tokenizer (Touvron et al., 2023), which has a vocabulary size of $V = 32k$.

To ensure both model types process the same volume of information per optimization step, we normalize the information content of the context window. Byte models are trained with a context window of $L_{byte} = 8192$ raw bytes, and BPE models with $L_{BPE} = 1792$ tokens, applying the best practice from Hwang et al. (2025) for a fair comparison.

4.2 Model Training.

We train models across a parameter sweep from 180M to 1.23B non-embedding parameters, employing a standard decoder-only Transformer architecture (Vaswani et al., 2017). We adopt modern architectural best practices, including pre-normalization, SwiGLU activation functions (Ramachandran et al., 2017), and Rotary Positional Embeddings (RoPE) (Su et al., 2024). While autoregressive models utilize standard causal masking, Masked Diffusion Models (MDMs) share the identical backbone but omit the causal mask. This enables bidirectional attention, allowing for global context reasoning during the denoising process. All models are trained using mixed precision on NVIDIA H100 GPUs.

Models are trained using the AdamW optimizer (Loshchilov & Hutter, 2019) ($\beta_1 = 0.9$, $\beta_2 = 0.95$) with a global batch size of $B = 1152$. With our sequence length normalization (Section 3.1), this ensures that the total volume of raw data (in bytes) processed per optimization step is approximately constant across all experiments. Learning rate is swept logarithmically from $1\text{e}{-4}$ to $3\text{e}{-3}$ with a 1% linear warm-up followed by cosine decay to minimum learning rate $2\text{e}{-4}$. We swept gradient clipping between 0.25 and 1.0 while maintaining a constant weight decay of 0.1.

MDMs are trained with a linear masking schedule ($\alpha_t = 1 - t$; $\frac{\alpha_t'}{1-\alpha_t} = -\frac{1}{t}$) and evaluated with a cosine masking schedule ($\alpha_t = 1 - cos(\frac{\pi}{2}(1-t))$; $\frac{\alpha_t'}{1-\alpha_t} = -\frac{\pi}{2}tan(\frac{\pi}{2}(1-t))$), in accordance with best practices from Shi et al. (2024).

4.3 Evaluation

To compare across divergent vocabulary sizes on a unified scale, we report **Bits-Per-Byte (BPB)**. BPB normalizes the total log-likelihood by the raw size of the dataset in bytes, effectively decoupling predictive performance from the discretization strategy. Formally:

$$\text{BPB} = \frac{\text{NLL}(D;\theta)/|D|_{\text{bytes}}}{ln(2)}$$

where $\text{NLL}(D;\theta) = -\sum_{x\in D}\log p_\theta(x)$ represents the total negative log-likelihood of the *tokenized* dataset and $|D|_{\text{bytes}}$ is byte count of the raw data.

Models are compared with equivalent total training FLOPs ($F \in \{2e19, 6e19, 2e20, 6e20, 2e21\}$). For each model parameterization and context length, we adjust the data budget to to achieve the target FLOPs. Detailed FLOPs computations are provided in Appendix B.

Downstream Task Evaluation We also evaluate models on downstream reasoning benchmarks: ARC-Easy (Clark et al., 2018), BoolQ (Clark et al., 2019), HellaSwag (Zellers et al., 2019), OBQA (Mihaylov et al., 2018), PIQA (Bisk et al., 2020), RACE (Lai et al., 2017), and SIQA (Sap et al., 2019). We define two matching protocols:

Compute match compares models with equivalent total training FLOPs ($F \approx 2 \times 10^{20}$), which typically pairs a lower-capacity Byte model against higher-capacity BPE baseline due to the quadratic cost of the $4\times$ longer byte sequences. **Capacity match** isolates representational differences by holding non-embedding parameter counts and data volumes constant. This highlights the performance deficit inherent to the byte-level objective when parameter count and data volume are held constant, independent of compute-balancing.

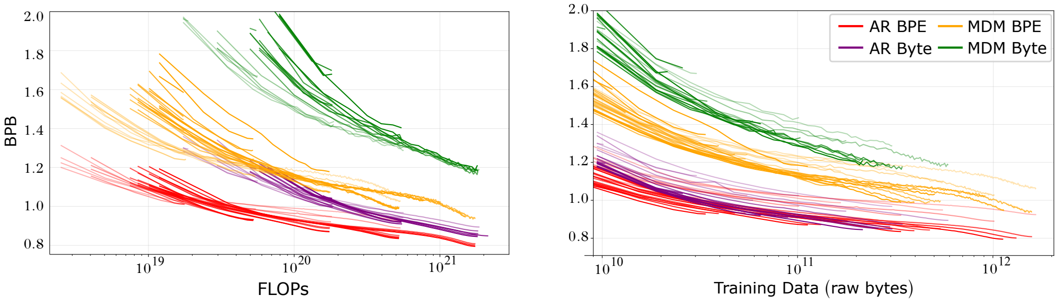


Figure 2: **Compute and Sample Efficiency. FLOPs (Left):** AR Byte and BPE models converge to a shared efficiency frontier, while MDM Byte models maintain a significant FLOPs penalty. **Data (Right):**AR Byte and BPE models show nearly identical sample efficiency; in contrast, MDM Byte models exhibit a persistent gap. Darker shades indicate larger model sizes.

Following Nie et al. (2025), to determine the conditional log-likelihood of a sequence x_0 for downstream tasks, we employ the chain rule decomposition $\log p_\theta(x_0|\text{prompt}) = \sum_{i=0}^{L-1} \log p_\theta(x_0^{(i)}|\text{prompt}, x_0^{(<i)}, m)$. This allows for direct comparison between the AR models and the naturally bidirectional masked diffusion models on a unified sequential likelihood basis.

5 Quantifying the Efficiency Frontier

5.1 Scaling Trajectories and BPB Convergence

Figure 2 reveals a fundamental divergence in how modeling objectives navigate the transition from compressed to raw byte inputs.

Autoregressive models achieve near-parity at scale. As compute approaches 10^{21} FLOPs, the gap between AR Byte and AR BPE models narrow, suggesting that for causal objectives, compute is an effective substitute for pre-computed discretization. In contrast, **MDM Byte models suffer a substantial and sustained performance penalty compared to MDM BPE** in the same compute scale, suggesting that the non-causal objective faces a fundamental structural difficulty in modeling sequences without the guidance of a pre-tokenized vocabulary.

The right side of Figure 2 plots performance against the raw volume of training data seen. It tells a similar story: AR BPE and Byte models show comparable usage of training data, with Byte models even frequently outperforming their BPE counterparts at higher compute budgets. MDM Byte models show more relative sample inefficiency: to match the perplexity of a BPE counterpart, a Byte MDM requires much larger data volumes.

Downstream Task Performance Zero-shot evaluation verifies that these BPB trends translate to semantic capabilities. As shown in Table 3, AR Byte models narrow the gap to their BPE counterparts as capacity increases from compute-matched (180M) to parameter-matched (717M) settings. Conversely, MDM Byte models stagnate, showing no meaningful scaling benefits at these budgets. While the steeper slope of the MDM Byte BPB curves suggests they might eventually match BPE counterparts at extreme scales, these performance gains have not yet materialized in the zero-shot reasoning tasks tested. Per-task performance breakdown is shared in Table 1 in the Appendix. All models exhibited trivial performance on complex tasks (HumanEval (Chen et al., 2021), MBPP (Austin et al., 2021), and BBH (Srivastava et al., 2022; Suzgun et al., 2022)), so we omit these numbers to focus on the tasks where meaningful signals were observed.

Model	**Params**	**Avg.**
Baseline (45×10^9 *BPE tokens)*		
AR BPE	717M	**46.23**
MDM BPE	717M	40.70
Compute Match ($F \approx 2 \times 10^{20}$*)*		
AR Byte	180M	41.71
MDM Byte	180M	37.73
Capacity Match (188×10^9 *Bytes)*		
AR Byte	717M	44.23
MDM Byte	717M	36.76

Figure 3: Task accuracy reflects the relative advantage of AR BPE models over Byte and MDM counterparts.

5.2 The Efficiency Gap of Byte Modeling.

To quantify the performance disparity between data representations, we conduct a scaling analysis following the protocol established by Hoffmann et al. (2022). Rather than identifying a static penalty, we characterize the computational investment required for models to resolve underlying data structures in the absence of a pre-computed tokenizer.

IsoFLOPs Curvature and Extrapolation For each fixed compute budget, we evaluate a sweep of parameter counts to identify the efficiency frontier. We fit parabolas to the BPB-parameter coordinates for each budget. In high-compute regimes where sampled model sizes were limited, these fitted parabolas allow us to characterize the shifting optimal parameter-to-data ratio on the compute budget frontier.

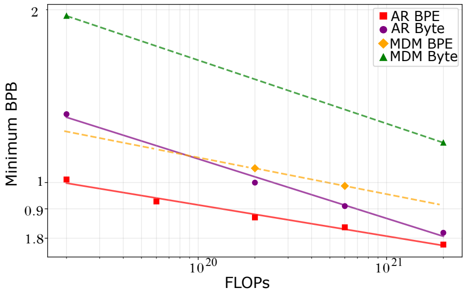


Figure 4: A power law is fit to extrapolated isoFLOPs minima. AR Byte and BPE are predicted to reach parity several orders of magnitude before MDM Byte and BPE do.

Divergent Scaling Trajectories To quantify the performance disparity between these paradigms, we observe the factor of additional training compute required for a byte-level model to achieve the same predictive performance as its BPE counterpart. Visually, this ratio corresponds to the horizontal distance between the BPE and Byte frontiers on Figure 4.

We observe that this efficiency ratio is not a static penalty but a dynamic value that evolves with scale. For the autoregressive objective (red and purple lines), the horizontal gap narrows as compute increases, indicating that the initial computational overhead of byte modeling is amortized over scale. In contrast, the gap between the MDM BPE and MDM Byte frontiers (yellow and green lines) is wider in magnitude and more persistent across tested FLOPs scales. While scaling provides some evident narrowing of the gap, the FLOPs penalty for MDMs remaining significantly higher than that of AR models across the observed range. This difference in performance-matching compute ratios suggests that the order-agnostic nature of the diffusion objective interacts poorly with granular byte-level representations at lower compute scales, a phenomenon we investigate mechanistically in the following sections.

6 Emergent Segmentation

The narrowing of the efficiency gap for autoregressive models suggests that they eventually develop internal structures that approximate the advantages of the pre-computed tokenizer. We hypothesize that because BPE is a pre-computation of frequent byte patterns, a byte-level model must expend computational resources to resolve these same statistical regularities.

6.1 Entropy as a Proxy for Segmentation

To investigate this, we analyze the predictive entropy of trained AR byte models across diverse text samples. We find that the predictive uncertainty is highly non-uniform. In fact, it exhibits high entropy at the start of frequent sub-word units and low entropy for the more predictable byte transitions within those units.

Figure 5 illustrates the peaking of AR byte predictive entropy at the initial bytes of frequent byte patterns. The black-outlined boxes denote the first non-space byte of the corresponding BPE token from the Llama 2 tokenizer. Regions of high entropy align with these BPE start boundaries.

To quantify the alignment between predictive uncertainty and the structural units constructed by BPE, we frame the task as a binary classification problem. We utilize the scalar entropy at each byte position as a score to predict the presence of a BPE start boundary (1 if start of token, 0 otherwise). The resulting ROC AUC of 0.829 demonstrates that the autoregressive model's uncertainty is highly non-uniform and demonstrates a strong statistical alignment with the probabilistic structures established by the tokenizer.

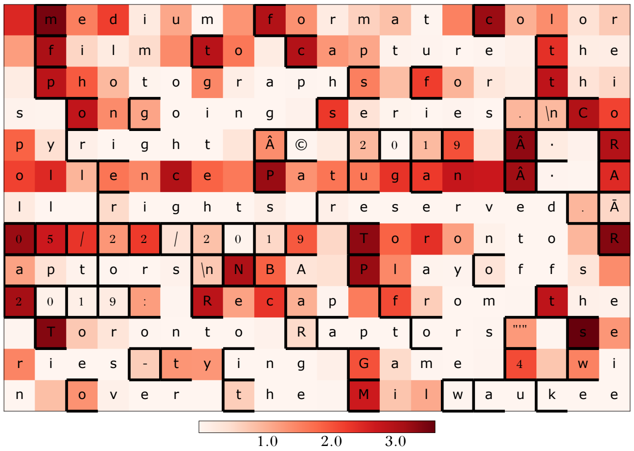


Figure 5: High-entropy bytes for an AR model align with the first non-space byte of corresponding BPE tokens (black outlines), achieving an ROC AUC of 0.829 for predicting BPE start boundaries.

This indicates that the model's output distribution reflects the underlying statistical regularities of the data, even in the absence of an explicit tokenizer.

6.2 Implicit Segmentation as a Structural Requirement

The behavioral alignment between predictive entropy and sub-word boundaries suggests that the predictive efficiency of autoregressive byte models is supported by their sensitivity to local statistical dependencies. The sharp transition in entropy at these boundaries indicates that the model successfully leverages a stable causal history to resolve high-frequency transitions within frequent byte patterns.

Conversely, the persistent efficiency gap in MDM suggests that order-agnostic objectives may struggle to establish this same degree of predictive regularity. If the utilization of local contiguity is a primary driver of efficiency, then the stochastic masking inherent in MDM, which disrupts these probabilistic patterns, may fundamentally hinder the development of this structural awareness. To test this, we move from observing emergent structures to a series of controlled experiments designed to scramble this local context and quantify the resulting fragility of the byte representation.

7 Mechanism of Failure: Context Fragility

To investigate the potential drivers of MDM's scaling overhead, we use AR permutation experiments as a proxy to isolate the impact of context destruction. We hypothesize that context fragility, the loss of local contiguity and global order, is a primary structural mechanism for the observed overhead. While a BPE token often corresponds to a subword unit with intrinsic semantic meaning, a single byte is semantically vacuous in isolation. It relies entirely on neighboring bytes, combined with its global position, to resolve its meaning. Consequently, byte-based sequences are more susceptible to losing informational value when their sequential integrity is compromised, forcing the model to resolve dependencies across a more challenging, low-signal input.

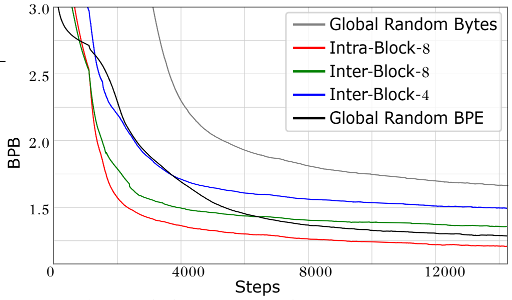
Experimental Setup: Permutation as Corruption To test this, we simulate the context destruction inherent in MDM within a controlled AR framework. We apply a static permutation π to sequences and their corresponding position IDs. We compare three permutation strategies (visualized in Figure 6): (1) **global random**: all token positions are randomly shuffled, destroying local and global structure; (2) **inter-block-N random**: blocks of size N bytes are shuffled while internal byte order is preserved, destroying global order while preserving local contiguity; (3) **intra-block-N random**: bytes within each block of size N are shuffled, destroying local order while preserving global order. For this study, we test $N = \{4, 8\}$ bytes.

To provide a model-agnostic measure of the structural information contained within different sequence representations, we utilize average loss in compressibility under the DEFLATE algorithm (Deutsch, 1996) as a model-free proxy for probabilistic structure of sequence data. DEFLATE is a combination of prefix matching with Huffman coding, letting us quantify the statistical regularities and repetitive patterns preserved under various tokenization and permutation strategies, offering a quantitative baseline for how much structural scaffolding is destroyed during context corruption.

The Primacy of Causal History. Our results in Section 6 show that AR models achieve an ROC AUC of 0.829 against BPE boundaries, proving that these structures emerge out of training for AR

π **Strategy**	**Visualization**	**Cmpr.**
Original BPE	`[diffusion][_models]`	–
Global BPE ■	`[_models][diffusion]`	-7%
Original Byte	`diff usio n_mo dels`	–
Global Byte ▨	`_mus isde ofno dlif`	-9%
Inter-4 🟦	`n_mo diff dels usio`	-14%
Inter-8 🟩	`n_models diffusio`	-9%
Intra-8 🟥	`ifdfsiuo noeld_ms`	-6%

(a) Permutation Strategies



(b) Training Dynamics (AR Byte)

Figure 6: **Context Fragility under Permutation.** (a) Strategies used to corrupt sequence integrity. Average loss in compressibility (%) serves as a model-free proxy for data structure. (b) AR Byte models suffer under global shuffling but recover performance when local contiguity (Inter-Block) or global causal order (Intra-Block) is preserved, highlighting that a stable causal history is a more powerful inductive bias than local predictability alone.

Transformer models. However, Byte MDMs fail to achieve similar efficiency despite using the same Transformer backbone.

Training dynamics in Figure 6 reveal three key observations:

Byte models are less robust to global shuffling . The AR Byte model suffers a much sharper performance drop under global random permutation compared to AR BPE. This confirms that BPE tokens provide a stronger independent learning signal when context is destroyed. This is also reflected in a higher average decrease in compressibility for shuffled bytes (-9%) versus BPE (-7%).

Local contiguity is a helpful inductive bias . Under inter-block-N permutation (preserving local chunks), byte models recover performance. This mirrors the compressibility trend, where preserving chunks increases average compressibility from total random permutation by approximately $5-10\%$.

Global context compensates for local noise . Under intra-block-N permutation (preserving global order), the byte model outperforms the globally-permuted BPE. This occurs even though the data is technically less compressible than inter-block sequences, suggesting that a stable causal history is a more powerful inductive bias for modeling than local predictability alone.

Implication for MDM. These findings identify a paradigm structural mismatch in the Byte MDM paradigm. BPE inherently encapsulates local contiguity within compressed units, and AR objectives preserve it via a stable causal history. The MDM objective, however, is doubly destructive: it operates on granular units (no encapsulation) while simultaneously scattering the context (no causal history). Our results suggest that by simultaneously shattering global sequential ordering and corrupting local structure, the masking process deprives the model of the dependencies required to efficiently resolve sub-semantic units. Deprived of this structural scaffolding, the diffusion objective must resolve dependencies across a combinatorial landscape of possible orderings, leading to the efficiency collapse observed in our scaling study.

Vocabulary Sensitivity Given the persistent gap between Byte and BPE MDMs, we further investigate whether increasing vocabulary size yields diminishing returns. In a sweep across GPT-2 ($V \approx 50k$) and Llama-3 ($V \approx 128k$) tokenizers, we find that while larger vocabularies offer higher compression, the optimal vocabulary size for masked diffusion is not static and depends heavily on model capacity and compute budget. Detailed results on this "upper limit" to vocabulary efficiency are provided in Appendix D

8 Conclusion

We quantified the efficiency gap between subword and byte-level representations, revealing a fundamental dichotomy: while autoregressive models effectively amortize tokenization costs at scale, byte-level MDMs suffer a persistent performance deficit within the comput budgets studied. This behavior likely stems from context fragility; our proxy experiments suggest that the MDM objective shatters the local contiguity and global context required to efficiently resolve sub-word semantics from raw byte streams. While our isoFLOPs extrapolation indicates that MDM byte models may eventually reach parity at extreme scales, our findings demonstrate that standard order-agnostic objectives require specific inductive biases to remain computationally viable in the raw byte regime. Future research should explore such possibilities, including hierarchical masking to preserve local contiguity, alternate backbones that prioritize granular dependencies, and the identification of optimal diffusion vocabularies that balance semantic density against classification difficulty.

REFERENCES

Anthropic. The Claude 3 model family: Opus, Sonnet, Haiku, 2024. URL https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf.

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

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces, 2023. URL https://arxiv.org/abs/2107.03006.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde 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, Josh 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. 2021.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions, 2019. URL https://arxiv.org/abs/1905.10044.

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, 2018. URL https://arxiv.org/abs/1803.05457.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

P. Deutsch. Deflate compressed data format specification version 1.3. RFC 1951, IETF, May 1996. URL http://www.ietf.org/rfc/rfc1951.txt.

Philip Gage. A new algorithm for data compression. *C Users Journal*, 12(2):23–38, 1994.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models, 2022. URL https://arxiv.org/abs/2203.15556.

Sukjun Hwang, Brandon Wang, and Albert Gu. Dynamic chunking for end-to-end hierarchical sequence modeling. *arXiv preprint arXiv:2507.07955*, 2025.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering, 2018. URL https://arxiv.org/abs/1809.02789.

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, 2025. URL https://arxiv.org/abs/2502.09992.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

Artidoro Pagnoni, Ram Pasunuru, Pedro Rodriguez, John Nguyen, Benjamin Muller, Margaret Li, Chunting Zhou, Lili Yu, Jason Weston, Luke Zettlemoyer, Gargi Ghosh, Mike Lewis, Ari Holtzman, and Srinivasan Iyer. Byte latent transformer: Patches scale better than tokens, 2024. URL https://arxiv.org/abs/2412.09871.

Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

Subham Sekhar Sahoo, Marianne Arriola, Aaron Gokaslan, Edgar Mariano Marroquin, Alexander M Rush, Yair Schiff, Justin T Chiu, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=L4uaAR4ArM.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. Social iqa: Commonsense reasoning about social interactions. In *EMNLP*, 2019. URL https://www.aclweb.org/anthology/D19-1454.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units, 2016. URL https://arxiv.org/abs/1508.07909.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data. In *Advances in Neural Information Processing Systems*, 2024.

Daria Soboleva, Faisal Al-Khateeb, Robert Myers, Jacob R Steeves, Joel Hestness, and Nolan Dey. SlimPajama: A 627B token cleaned and deduplicated version of RedPajama. https://cerebras.ai/blog/slimpajama-a-627b-token-cleaned-and-deduplicated-version-of-redpajama, 2023. URL https://huggingface.co/datasets/cerebras/SlimPajama-627B.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*, 2022.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, , and Jason Wei. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*, 2022.

Chaofan Tao, Qian Liu, Longxu Dou, Niklas Muennighoff, Zhongwei Wan, Ping Luo, Min Lin, and Ngai Wong. Scaling laws with vocabulary: Larger models deserve larger vocabularies, 2024. URL https://arxiv.org/abs/2407.13623.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. URL https://arxiv.org/abs/2307.09288.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Dimitri von Rütte, Janis Fluri, Omead Pooladzandi, Bernhard Schölkopf, Thomas Hofmann, and Antonio Orvieto. Scaling behavior of discrete diffusion language models, 2025. URL https://arxiv.org/abs/2512.10858.

Junxiong Wang, Tushaar Gangavarapu, Jing Nathan Yan, and Alexander M Rush. Mambabyte: Token-free selective state space model. *arXiv preprint arXiv:2401.13660*, 2024.

xAI. Grok 4.1 model card, 2025. URL https://data.x.ai/2025-11-17-grok-4-1-model-card.pdf.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306, 2022. doi: 10.1162/tacl_a_00461. URL https://aclanthology.org/2022.tacl-1.17/.

LILI YU, Daniel Simig, Colin Flaherty, Armen Aghajanyan, Luke Zettlemoyer, and Mike Lewis. MEGABYTE: Predicting million-byte sequences with multiscale transformers. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=JTmO2V9Xpz.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

Impact Statement

Our findings suggest several avenues for future investigation into the interaction between data representation and modeling objectives. By establishing compute-matched baselines, this work provides the open-source community with a structural roadmap for selecting data representations that align with specific modeling objectives, helping to prevent the misallocation of limited computational resources toward inefficient scaling trajectories.

This paper presents work whose goal is to advance the field of Machine Learning by quantifying and analyzing the efficiency of different data representations. The compute-controlled pre-training required for this study involves significant computational resources and associated energy consumption, but our findings identify critical structural inefficiencies in current modeling paradigms, providing the foundation for more compute-efficient architectures, potentially reducing the long-term environmental impact of training large-scale, modality-agnostic models.

A Discretization of Continuous-Time Diffusion

We follow the continuous-time discrete diffusion framework presented by Shi et al. (Shi et al., 2024) (MD4), which generalizes the discrete diffusion models of Austin et al. (Austin et al., 2023) (D3PM). Here we detail the forward process, training objective, and sampling strategy used in our experiments.

Forward Process (Corruption). We consider a sequence x_0 of length L tokens, where each token $x_0^{(i)}$ belongs to a finite vocabulary comprised of the model vocabulary plus a special mask token $\mathcal{V} \cup \{\texttt{[MASK]}\}$. The forward process is characterized as a continuous-time Markov chain over time interval $t \in [0, 1]$ applied independently to each token.

At any time t, the marginal distribution of the noisy sequence x_t factorizes as $q(x_t|x_0) = \prod_{i=1}^{L} q(x_t^{(i)}|x_0^{(i)})$, where:

$$q(x_t^{(i)}|x_0^{(i)}) = \begin{cases} \alpha_t & \text{if } x_t^{(i)} = x_0^{(i)} \\ 1 - \alpha_t & \text{if } x_t^{(i)} = \texttt{[MASK]} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Here, α_t is a monotonically decreasing masking schedule from $\alpha_0 \approx 1$ (no masking) to $\alpha_1 \approx 0$ (fully masked).

Training Objective (Continuous-Time ELBO) The generative process is parameterized by a neural network $p_\theta(x_0|x_t, t)$ trained to predict the original uncorrupted sequence x_0 from the noisy state x_t. Following Shi et al. (Shi et al., 2024), we minimize the simplified continuous-time variational lower bound (ELBO), which reduces to a weighted cross-entropy over the masked tokens:

$$\mathcal{L}_{\text{MDM}}(\theta) = \mathbb{E}_{t \sim \mathcal{U}(0,1)} \left[\underbrace{\frac{\alpha_t'}{1 - \alpha_t}}_{w(t)} \sum_{i \in \{i : x_t^{(i)} = \texttt{[MASK]}\}} \log p_\theta(x_0^{(i)}|x_t, t) \right] \tag{2}$$

Note that because α_t is decreasing over t, the derivative α_t' is negative, making $w(t)$ negative, which offsets the negative value produced by the $\log p_\theta$.

Sampling: Reverse Process (Denoising). During inference, we simulate the reverse process by discretizing time into T steps: $1 = t_T > \cdots > t_0 = 0$. At each step $t \to s$ (where $s < t$), we update the sequence based on the model's prediction. Tokens that are already unmasked are kept fixed. For tokens that are still masked at time t, we sample their value at the next step s according to:

$$x_s^{(i)} \sim \begin{cases} \text{Cat}\left(\frac{\alpha_s - \alpha_t}{1 - \alpha_t} \sigma(f_\theta(x_t, t))^{(i)} + \frac{1 - \alpha_s}{1 - \alpha_t} e_m\right) & \text{if } x_t^{(i)} = \texttt{[MASK]} \\ x_t^{(i)} & \text{else} \end{cases} \tag{3}$$

Implementation and Schedules. We approximate the expectation over time by sampling $t \sim \mathcal{U}(0, 1)$ for each batch. We use a linear masking schedule during training and cosine schedule during evaluation and sampling.

B FLOPs Computation

We calculate the forward pass FLOPs for our Transformer architecture based on the specific operations of our decoder-only backbone. We largely follow the methodology of Hoffmann et al. (2022), with modifications to account for the SwiGLU activation functions used in the Transformer architecture.

For a single forward pass, the FLOPs are computed as follows:

- **Embeddings**: $2 \times L \times V \times d_{\text{model}}$
- **Attention Layer** (per layer):
  - **QKV Projections**: $3 \times 2 \times L \times d_{\text{model}} \times (n_{\text{heads}} \times d_{\text{head}})$
  - **Attention Logit Calculation** (QK^T): $2 \times L^2 \times (n_{\text{heads}} \times d_{\text{head}})$
  - **Attention Softmax Weighting**: $2 \times L^2 \times (n_{\text{heads}} \times d_{\text{head}})$
  - **Output Projection**: $2 \times L \times (n_{\text{heads}} \times d_{\text{head}}) \times d_{\text{model}}$
- **MLP Block (SwiGLU)** (per layer):
  - **Up-Projections and Gating**: $2 \times (2 \times L \times d_{\text{model}} \times d_{\text{ff}})$
  - **Down-Projection**: $2 \times L \times d_{\text{ff}} \times d_{\text{model}}$
- **Language Modeling Head**: $2 \times L \times d_{\text{model}} \times V$

The total training compute (F) is then calculated by multiplying the per-step forward pass FLOPs by a factor of 3 to account for the backward pass and gradient computation, and then scaling by the total number of tokens processed.

C Full Task Suite Performance

Model	Params	ARC-E	BoolQ	HellaS	OBQA	PIQA	RACE	SIQA	Avg
Baseline ($D = 45 \times 10^9$ BPE tokens)									
AR BPE	717M	49.49	59.45	41.93	36.40	67.08	30.91	38.33	**46.23**
MDM BPE	717M	35.69	59.14	31.93	32.40	59.58	29.67	36.49	40.70
Compute Match ($D = 188 \times 10^9$ Byte tokens, N =180M)									
AR Byte	180M	43.01	54.62	35.74	30.80	63.11	28.42	36.28	41.71
MDM Byte	180M	26.68	60.64	29.03	31.00	55.28	27.85	33.62	37.73
Capacity Match ($D = 188 \times 10^9$ Byte tokens)									
AR Byte	717M	47.60	56.29	39.29	33.00	66.32	30.14	37.00	44.23
MDM Byte	717M	30.60	44.86	30.33	31.60	56.75	27.18	36.03	36.76

Table 1: Comparison of Byte vs. BPE models. AR Byte models remain relatively competitive with BPE baselines, whereas MDM Byte models significantly underperform, even when parameter counts are matched. (MDM likelihoods computed via chain rule decomposition). Last column shows the average of zero-shot accuracy scores across tasks.

D An Upper Limit to Vocabulary Efficiency

To test if larger vocabularies yield further gains, we extended our sweep to include GPT-2 ($V \approx 50k$) and Llama-3 ($V \approx 128k$) tokenizers. We find that BPE MDM models consistently outperform Byte counterparts regardless of vocabulary size. Our results indicate that the optimal vocabulary size for

Tokenizer	Vocab (V)	Avg. Bytes
Byte	256	1.00
Llama-2	$32,000$	3.74
GPT-2	$50,257$	4.16
Llama-3	$128,000$	4.36

Table 2: Tokenizer vocabulary size and compression rates.

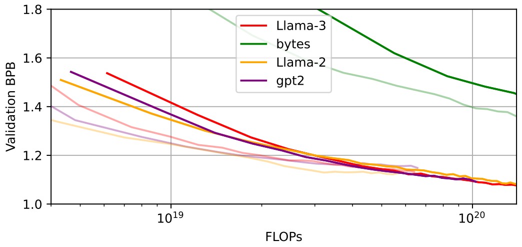


Figure 7: Iso-FLOPs curves for 180M and 717M models. Larger vocabularies offer higher compression.

MDM is not static; it requires balancing semantic density against the increased classification difficulty of massive vocabularies.

Over the compute settings sizes described in Section 4 ($N = 180$M, 717M), we find, consistent with the main results of this paper, that BPE MDM models consistently outperform their Byte counterparts regardless of vocabulary size.

As shown in Figure 7, we observe that the best-performing vocabulary size depends on model capacity and compute budget. Observe, for example, that for the smaller 180M model, the Llama-2 tokenizer ($V = 32k$) fares the best. But as the FLOPs budget increases to 10^{20} for the larger 717M model, the larger BPE tokenizers (GPT-2, Llama-3) close the gap, even overtaking the smaller-vocabulary Llama-2 tokenizer baseline. These results indicate that optimal vocabulary size for masked diffusion is not static; it requires balancing the benefits of semantic density against the token sparsity and classification difficulty of massive vocabularies.