SCALING UP MASKED DIFFUSION MODELS ON TEXT

Anonymous authors
Paper under double-blind review

ABSTRACT

Masked diffusion models (MDMs) have shown promise in language modeling, yet their scalability and effectiveness in core language tasks, such as conditional generation and language understanding, remain underexplored. This paper establishes the first scaling law for MDMs, demonstrating a scaling rate comparable to autoregressive models (ARMs) and a relatively small compute gap. Motivated by their scalability, we train a family of MDMs with up to 1.1 billion (B) parameters to systematically evaluate their performance against ARMs of comparable or larger sizes. Fully leveraging the probabilistic formulation of MDMs, we propose a simple yet effective *unsupervised classifier-free guidance* that effectively exploits large-scale unpaired data, boosting performance for conditional inference. In language understanding, a 1.1B MDM shows competitive results, outperforming the larger 1.5B GPT-2 model on four out of eight zero-shot benchmarks. In conditional generation, MDMs provide a flexible trade-off compared to ARMs utilizing KV-cache: MDMs match the performance of ARMs while being 1.5 times faster, or achieve higher quality than ARMs at a slightly higher computational cost. Moreover, MDMs address challenging tasks for ARMs by effectively handling bidirectional reasoning and adapting to temporal shifts in data. Notably, a 1.1B MDM breaks the *reverse curse* encountered by much larger ARMs with significantly more data and computation, such as Llama (13B) and GPT-3 (175B).

1 INTRODUCTION

Autoregressive models (ARMs) have long been regarded as the gold standard in probabilistic language modeling. Their ability to predict the next token, grounded in the chain rule, naturally aligns with the sequential nature of language and scales effectively (Radford, 2018; Radford et al., 2019; Brown, 2020; OpenAI, 2022; Achiam et al., 2023; Touvron et al., 2023a;b; Dubey et al., 2024) when integrated with Transformers (Vaswani, 2017). However, ARMs exhibit inherent limitations, particularly in reasoning tasks that require bidirectional context understanding or handling temporal shifts in data. These shortcomings, widely recognized as the *reverse curse* (Berglund et al., 2023) and *temporal quality degradation* (Vela et al., 2022), significantly hinder their applicability in complex language modeling scenarios. Additionally, their linear sampling time growth w.r.t. the output length poses practical challenges for long text generation.

The limitations of ARMs have sparked interest in an alternative approach: masked diffusion models (MDMs) (Austin et al., 2021; Hoogeboom et al., 2021b;a; He et al., 2022; Meng et al., 2022; Sun et al., 2022; Lou et al., 2023; Sahoo et al., 2024; Shi et al., 2024; Ou et al., 2024). MDMs present a promising alternative due to their unique probabilistic framework, which enables flexible bidirectional context modeling by filling in masked positions across a sequence. Recent advances (Lou et al., 2023; Sahoo et al., 2024; Shi et al., 2024; Ou et al., 2024) have shown promise in unconditional text generation and zero-shot perplexity evaluation. Despite recent progress, the scalability of MDMs and their effectiveness in critical language tasks, such as conditional generation and language understanding, remain open questions. Furthermore, it is still unclear whether MDMs can address the inherent limitations of ARMs, such as improving bidirectional reasoning capabilities.

Given that scalability and generality across tasks are core attributes of large language models, advancing MDMs requires not only a focus on algorithm design (Austin et al., 2021; Lou et al., 2023; Sahoo et al., 2024; Shi et al., 2024; Ou et al., 2024) but also attention to an orthogonal dimension: the exploration of scalability and generality. From this perspective, this paper challenges the long-standing dominance of ARMs by presenting a comprehensive study of MDMs regarding key factors in language models: scalability and capabilities for language understanding and conditional generation. To this end, we train a family of MDMs with up to 1.1 billion (B) parameters on a large-scale dataset and establish the first scaling law for MDMs. Leveraging their unique probabilistic framework, we propose a simple yet effective *unsupervised classifier-free guidance (CFG)* mechanism to leverage unsupervised data to enhance inference performance in language tasks involving conditional distributions. Notably, unsupervised CFG does not rely on paired data as standard CFG (Ho & Salimans, 2022) but can still benefit from paired data when available, achieving performance that surpasses standard CFG. Supported by the scaling law and unsupervised CFG, our extensive experiments yield the following key findings:

- **Strong scalability.** As scaling computate budgets from 6×10^{18} to 10^{20} FLOPs, the optimal validation loss of MDMs decreases according to a power law, with a rate matching that of ARMs. While MDMs maintain a constant computation gap of 16 times compared to ARMs, this gap is smaller than the factor of 64 observed in continuous diffusion models (Gulrajani & Hashimoto, 2024) and can be further minimized with future optimizations.
- **Competitive in zero-shot language understanding.** Across eight standard zero-shot benchmarks like *commonsense reasoning* and *reading comprehension*, MDMs outperform not only a same-sized ARM with the same pre-training FLOPs but also a larger 1.5B GPT-2 model on four tasks. Furthermore, when scaled up with 16 times more pre-training time, as suggested by the scaling law, MDMs consistently surpass ARMs across all tasks.
- **Flexible trade-off in conditional generation.** On the standard MT-Bench, a 1.1B MDM matches the performance of a same-sized ARM while achieving a 1.5 times speedup in sampling time. By increasing sampling steps, MDMs can further improve generation quality at the cost of being 1.4 times slower. ARMs are equipped with KV-cache, a technique to speed up sequential sampling while MDMs exploit no system optimization but require 16 times pre-training time.
- **Addressing challenging tasks for ARMs.** MDMs effectively relieve *temporal quality degradation* (Vela et al., 2022) compared to a same-sized ARM and successfully overcome the *reverse curse* (Berglund et al., 2023) encountered by much larger ARMs with significantly more data and computation, such as Llama (13B) and GPT-3 (175B).

2 Masked Diffusion Models on Text

In analogy to continuous diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2020), MDMs (Austin et al., 2021; Lou et al., 2023; Ou et al., 2024) also introduce a forward process that gradually adds noise to the data and learn a corresponding reverse process to generate samples. Our basic approach is built upon Ou et al. (2024), an advanced MDM suitable for scaling up.

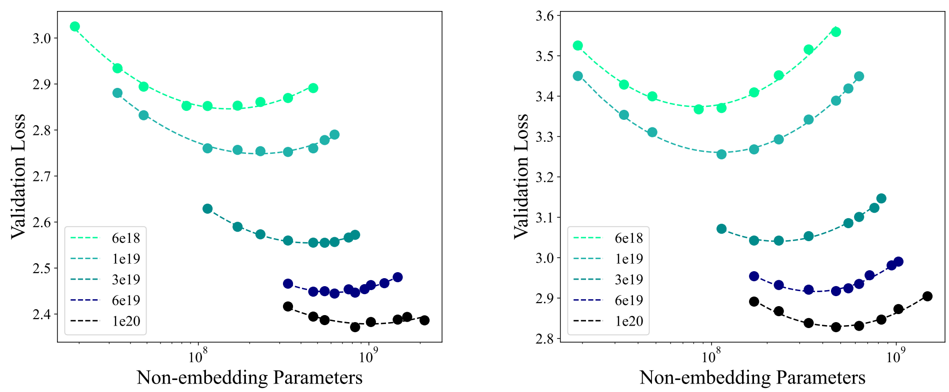
Forward process. Let K and L denote the vocabulary size and sentence length respectively. Given a sentence $\boldsymbol{x}_0 \in \{0, 1, \ldots, K-1\}^L$ and a noise level $t \in [0, 1]$, the forward process in MDMs randomly and independently masks out tokens in the sentence, formulated as follows:

$$q_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0) = \prod_{i=0}^{L-1} q_{t|0}(\boldsymbol{x}_t^i|\boldsymbol{x}_0^i) \quad \text{and} \quad q_{t|0}(\boldsymbol{x}_t^i|\boldsymbol{x}_0^i) = \begin{cases} \alpha_t, & \boldsymbol{x}_t^i = \boldsymbol{x}_0^i, \\ 1-\alpha_t, & \boldsymbol{x}_t^i = m, \end{cases} \tag{1}$$

where $\boldsymbol{x}^i$ denotes the i-th element of $\boldsymbol{x}$, m denotes the mask token (Devlin, 2018), $\boldsymbol{x}_t$ denotes the noisy data at time t and $q_0(\cdot)$ is the data distribution $p_{\text{data}}(\cdot)$. We set the hyperparameter α_t as $1-t$ for the best empirical performance as suggested in previous work (Lou et al., 2023; Sahoo et al., 2024; Shi et al., 2024). We focus on a discrete-time process and refer the readers to prior work (Campbell et al., 2022) for extensions to continuous time.

Reverse process. The reverse process in MDMs iteratively recover values for masked tokens, starting from a mask sequence $\boldsymbol{x}_1$. Let $0 \le s < t \le 1$, the reverse process is characterized by

$$q_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t) = \prod_{i=0}^{L-1} q_{s|t}(\boldsymbol{x}_s^i|\boldsymbol{x}_t) \;\text{ and }\; q_{s|t}(\boldsymbol{x}_s^i|\boldsymbol{x}_t) = \begin{cases} 1, & \boldsymbol{x}_t^i \neq m, \boldsymbol{x}_s^i = \boldsymbol{x}_t^i, \\ \frac{s}{t}, & \boldsymbol{x}_t^i = m, \boldsymbol{x}_s^i = m, \\ \frac{t-s}{t} q_{0|t}(\boldsymbol{x}_s^i|\boldsymbol{x}_t), & \boldsymbol{x}_t^i = m, \boldsymbol{x}_s^i \neq m, \\ 0, & \text{otherwise}. \end{cases} \tag{2}$$



(a) ARMs. (b) MDMs.

Figure 1: **IsoFLOP curves** plot the optimal model size under a fixed computation budget.

Here $q_{0|t}(\cdot|\cdot)$ is the data prediction model (Ho et al., 2020) to be learned. Notably, Ou et al. (2024) revealed an intrinsic property of MDMs that $q_{0|t}(\cdot|\cdot)$ can be represented by conditional distributions on clean data $p_{\text{data}}(\cdot|\cdot)$ independently from the time t, distinct from other diffusion. Formally,

$$q_{0|t}(\boldsymbol{x}_0^i|\boldsymbol{x}_t) = p_{\text{data}}(\boldsymbol{x}_0^i|\boldsymbol{x}_t^{\text{UM}}), \tag{3}$$

where $\boldsymbol{x}_t^{\text{UM}}$ collects all unmasked tokens in noisy data $\boldsymbol{x}_t$ and $p_{\text{data}}(\cdot|\cdot)$ is irrelevant to t.[1]

Training objective. A distribution $p_{\boldsymbol{\theta}}(\boldsymbol{x}_0^i|\boldsymbol{x}_t)$ parameterized by $\boldsymbol{\theta}$ is employed to approximate $p_{\text{data}}(\boldsymbol{x}_0^i|\boldsymbol{x}_t^{\text{UM}})$, optimizing the following upper bound on negative log-likelihood (Ou et al., 2024):

$$-\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0) \leq \int_0^1 \frac{1}{t}\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[\sum_{\{i|\boldsymbol{x}_t^i=m\}} -\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0^i|\boldsymbol{x}_t)\right] dt \triangleq \mathcal{L}. \tag{4}$$

We emphasize that the formulation is particularly suitable for scaling. First, it is among the best MDMs w.r.t. zero-shot perplexity (Ou et al., 2024). Second, it removes the timestep from input and minimally modifies the original Transformers (see Sec. 3). Third, it enables unsupervised classifier-free guidance, which does not rely on paired data yet is effective in language tasks (see Sec. 4).

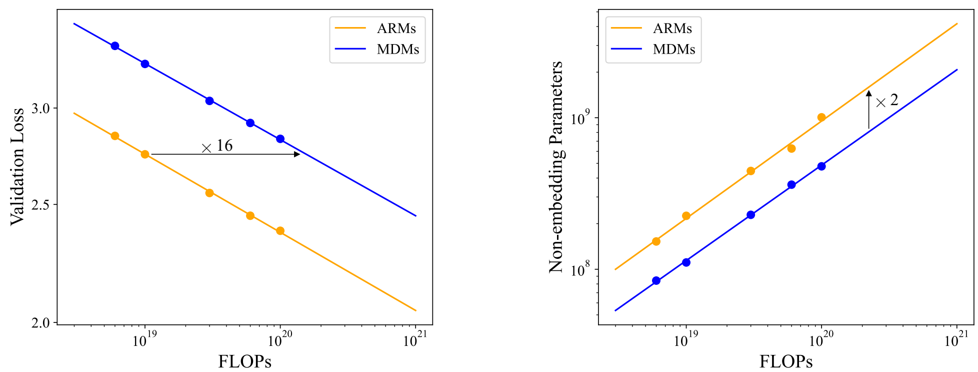
3 Scaling Laws for Masked Diffusion Models

Scaling laws (Kaplan et al., 2020; Hoffmann et al., 2022) characterize the quantitative power-law relationship between model performance and computational resources under constraints, significantly influencing the progress of large ARMs. Previous work Ye et al. (2023) fine-tunes pre-trained XLM-RoBERTa (Goyal et al., 2021; Conneau, 2019) models into MDMs and investigates scaling trends by varying the size of XLM-RoBERTa. However, a detailed exploration of scaling laws for MDMs, along with a fair comparison to ARMs in terms of scalability, remains absent. In this section, we address these two key questions. Our results reveal the strong scalability of MDMs, highlighting their potential as a competitive alternative to ARMs in language modeling.

Model. We employ a Transformer decoder for ARMs and the corresponding Transformer encoder for MDMs (note that it is unnecessary to input timestep t according to Eq. (3)). The main differences between these architectures are: (1) the encoder has an additional dimension in its embedding layer for the mask token, and (2) the encoder's self-attention does not use a causal mask. All other architectural settings (e.g., depth, hidden size, and number of heads) remain consistent in both models.

We further enhance both models with several techniques inspired by advanced language models like Llama (Touvron et al., 2023a;b). Specifically, we adopt Pre-LayerNorm with RMSNorm (Zhang &

[1]For example, if $\boldsymbol{x}_t = [3, 5, m, 2]$, then $\boldsymbol{x}_t^{\text{UM}} = [3, 5, \cdot, 2]$ and $p_{\text{data}}(\cdot|[3, 5, \cdot, 2])$ is irrelevant to t.



(a) Loss-Flops curve.

(b) Parameters-Flops curve.

Figure 2: **Scaling laws for MDMs.** Compared to ARMs, MDMs demonstrate competitive scalability with comparable scaling rates and similar scaling behavior on utilizing the parameter capacity.

Sennrich, 2019) for better stability, use SwiGLU (Shazeer, 2020) as the activation function to enhance non-linearity, and implement RoPE (Su et al., 2024) for more expressive positional encoding.

Data. The well-known Chinchilla scaling law (Hoffmann et al., 2022) utilizes a large dataset with more data than the number of training tokens. Motivated by it, we employ the open-source SlimPajama dataset (Soboleva et al., 2023), a multi-corpora dataset comprising 627 billion tokens, which is sufficiently large for all of our experiments. For simplicity and fairness, we employ the Llama2 tokenizer (Touvron et al., 2023b) for both ARMs and MDMs. Additionally, we set the context length to 2048. Further implementation details are provided in Appendix B.2.

IsoFLOP analysis. We conduct a standard IsoFLOP analysis (Hoffmann et al., 2022) to identify the optimal allocation between the non-embedding parameters N and dataset size D. Specifically, building on prior studies (Kaplan et al., 2020; Hoffmann et al., 2022), we scale the compute budget C from 6×10^{18} to 10^{20} FLOPs. For a fixed C, we vary N and D such that $C = 6ND$, a relationship valid for both ARMs and MDMs. We measure the corresponding validation loss $\mathcal{L}$ in Eq. (4) and fit a quadratic function to capture the relationship between the validation loss $\mathcal{L}$ and the logarithm of the parameter size $\log N$. This regression allows us to determine the optimal parameter size N_C, which corresponds to the minimum validation loss $\mathcal{L}_C$ for a given compute budget. The IsoFLOP analysis results are visualized in Fig. 1.

Scaling laws. After obtaining the optimal validation losses for the corresponding compute budget in $\{C_0, C_1, \ldots, C_{n-1}\}$, we fit the following scaling law to model the relationship between them:

$$\min_{\alpha,\beta} \sum_{i=0}^{n-1} \left(\log \mathcal{L}^*_{C_i} - \alpha \log C_i - \beta\right)^2 . \tag{5}$$

Let α^* and β^* denote the solution of Eq. (5) and the validation loss empirically follows $\mathcal{L} = e^{\beta^*} C^{\alpha^*}$.

As illustrated in Fig. 2a, the validation loss of MDMs decreases according to a power law as the compute budget increases, following a rate similar to that of ARMs. MDMs still require approximately 16 times more computational resources than ARMs to achieve comparable validation losses. There is still potential to narrow this constant since optimizations for MDMs in model, data, and system remain unexplored. Besides, for reference, Gulrajani & Hashimoto (2024) reported that the constant factor between continuous diffusion models (CDMs) and ARMs is 64.

Furthermore, the optimal model size also follows a power-law relationship with the compute budget, as shown in Fig. 2b. Notably, the optimal size of MDMs is approximately half that of ARMs across different computations, reflecting a very similar scaling behavior on utilizing the parameter capacity.

In conclusion, the comparable scaling rates and the relatively small constant factors suggest that MDMs have strong scalability and promising potential as an alternative to ARMs on a large scale.

4 Unsupervised Classifier-free Guidance

We propose a surprisingly simple yet effective approach that leverages unlabeled data to boost performance in various language tasks, dubbed *unsupervised classifier-free guidance (CFG)*.

CFG. CFG (Ho & Salimans, 2022) is an effective and versatile technique widely used in both continuous and discrete diffusion models, with applications spanning image (Ho & Salimans, 2022; Chang et al., 2023) and text generation (Lovelace et al., 2024). Rooted in Bayes' rule, CFG simultaneously trains a conditional and an unconditional diffusion model, introducing a rescaled distribution for inference. Specifically, at a given timestep $t \in [0, 1]$, CFG (Chang et al., 2023) is defined as:

$$\tilde{p}_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{c}, \boldsymbol{x}_t) \propto \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{c}, \boldsymbol{x}_t)^{1+w}}{p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{x}_t)^w}, \tag{6}$$

where $\boldsymbol{c}$ is the condition, w is a hyperparameter that flexibly controls the strength of $\boldsymbol{c}$, and $p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{c}, \boldsymbol{x}_t)$ and $p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{x}_t)$ are the conditional and unconditional models respectively.

Notably, it seems that the conditional model must be trained on paired data before applying CFG. Consequently, to the best of our knowledge, all existing work (Ho & Salimans, 2022; Chang et al., 2023; Lovelace et al., 2024) fall into supervised settings, where paired data $\boldsymbol{c}, \boldsymbol{x}$ is readily available.

Unsupervised CFG. We extend CFG to an unsupervised setting by introducing a new formulation:

$$\tilde{p}_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{c}, \boldsymbol{x}_t) \propto \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{c}, \boldsymbol{x}_t)^{1+w}}{p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{m}, \boldsymbol{x}_t)^w}, \tag{7}$$

where $\boldsymbol{m}$ is a mask sequence of the same length as $\boldsymbol{c}$. Compared to Eq. (6), the dummy variable $\boldsymbol{m}$ translates the unconditional distribution to a conditional format without adding new information. For simplicity, we continue to refer to $p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{m}, \boldsymbol{x}_t)$ as the unconditional distribution in unsupervised CFG throughout this paper.

The core insight is that an MDM already characterizes both distributions employed in Eq. (7) during unsupervised pretraining. Specifically, in language tasks, both $\boldsymbol{c}$ and $\boldsymbol{x}$ can be viewed as segments of a whole sequence, following the same distribution of unsupervised samples for pretraining.[2] After the pretraining on large-scale text data, MDMs can capture the joint distribution of the whole sequence, i.e. $p_{\text{data}}(\boldsymbol{c}, \boldsymbol{x})$. Under the formulation, MDMs simultaneously learn all conditional distributions on clean data induced by $p_{\text{data}}(\boldsymbol{c}, \boldsymbol{x})$ according to Eq. (3). In particular, we have:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{c}, \boldsymbol{x}_t) \approx p_{\text{data}}(\boldsymbol{x}_0|\boldsymbol{c}, \boldsymbol{x}_t^{\text{UM}}) \quad \text{and} \quad p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{m}, \boldsymbol{x}_t) \approx p_{\text{data}}(\boldsymbol{x}_0|\boldsymbol{x}_t^{\text{UM}}), \tag{8}$$

where both distributions are factorized as in Eq.(3), and the approximation error is due to the gap between the model distribution and the true data distribution. Notably, Eq.(8) also implies that the unconditional distribution $p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{x}_t)$ used in standard CFG and the conditional distribution with a dummy variable $p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{m}, \boldsymbol{x}_t)$ are equivalent.

We have explained why unsupervised CFG works without paired data (see experiments in Sec. 5). Moreover, when paired data is available for downstream tasks, simply fine-tuning the conditional distribution in MDMs—similar to the classical approach used for ARMs—not only further improves the performance of unsupervised CFG but also outperforms the standard CFG trained on paired data, demonstrating its superior capability in leveraging large-scale unpaired data (see Sec. 6).

Prior studies Zhao et al. (2021); Holtzman et al. (2021) emphasize the importance of unconditional distributions in large language models. By exploiting the distinctive properties of MDMs, unsupervised CFG introduces a novel approach to estimating the unconditional distribution (i.e., Eq. (8)) and incorporates this estimate through an alternative mechanism (i.e., Eq. (7)), inspired by the standard CFG framework (i.e., Eq. (6)).

5 Zero-shot Language Understanding

We investigate the capabilities of MDMs in zero-shot language understanding, a critical skill for language models that has been largely overlooked in prior studies (Austin et al., 2021; Lou et al.,

[2] E.g., the question "where does the sun rise?" and answer "from the east." is a paired sample but their concatenation "where does the sun rise? from the east." can be modeled by an MDM with unsupervised training.

Table 1: **Ablation of unsupervised CFG without paired data.** Unsupervised CFG significantly improves the performance of MDMs across eight common question-answering tasks.

	ARC-Easy	BoolQ	Hellaswag	OpenBookQA	PIQA	RACE	SIQA	LAMBADA
MDM w/o CFG	37.42	61.50	33.46	27.00	60.34	29.28	36.95	36.00
MDM w/ CFG	**39.02**	**62.17**	**34.10**	**34.20**	**60.39**	**30.81**	**37.41**	**40.99**

2023; Sahoo et al., 2024; Shi et al., 2024; Ou et al., 2024; Gat et al., 2024). Our results show that MDMs are highly competitive to ARMs of similar model sizes and computations.

Benchmarks. To provide a comprehensive evaluation, we assess MDMs on eight widely used benchmarks involving *commonsense reasoning* and *reading comprehension*: Hellaswag (Zellers et al., 2019), ARC-Easy (Clark et al., 2018), BoolQ (Clark et al., 2019), PIQA (Bisk et al., 2020), SIQA (Sap et al., 2019), OpenBookQA (Mihaylov et al., 2018), RACE (Lai et al., 2017), and LAMBADA (Paperno et al., 2016). For a detailed description of these benchmarks, see Appendix D.

On certain challenging benchmarks such as ARC-Challenge (Clark et al., 2018), WinoGrande (Sakaguchi et al., 2021), and MMLU (Hendrycks et al., 2020), both ARMs and MDMs pre-trained in Sec. 3 perform similarly to random guessing. This is consistent with findings from Wei et al. (2022a), which showed that only ARMs with more than 10^{22} training FLOPs can surpass random guessing on MMLU, a phenomenon known as the emergence of new capabilities in large language models. We leave the exploration of their potential emergent abilities at a larger scale as future work.

Evaluation. We employ the popular Language Model Evaluation Harness framework (Gao et al., 2024) to evaluate both ARMs and MDMs. For the LAMBADA dataset, given a prompt, we apply greedy sampling to generate responses from each model and calculate the matching accuracy against the ground truth (please refer to Appendix A for the greedy sampling algorithm of MDMs). For other tasks, we report the accuracy of each model that selects the correct answer from the provided options based on the given context. Specifically, we compute the likelihood of each option given the prompt and choose the answer with the highest likelihood.

Fixing the train-test discrepancy. Due to employing a bidirectional Transformer encoder, MDMs face a train-test discrepancy in context lengths, negatively impacting model performance. Specifically, the training context length is fixed at 2048 tokens, while the testing context length is variable and often shorter. To address this issue, we propose two mitigation strategies: (1) allocate a portion of training data with variable sequence lengths $L \sim \mathcal{U}[1, 2048]$, where $\mathcal{U}[\cdot]$ denotes the uniform distribution; (2) pad sentences with mask tokens to reach 2048 tokens during evaluation.

As present in Appendix C.1, both strategies effectively reduce the train-test discrepancy, and only a small proportion (e.g., 1%) of variable-length training data is sufficient to activate the capability to handle variable length inputs. Given its superior inference efficiency (e.g., 20 times faster than method (2) on the Hellaswag dataset), we employ method (1) in subsequent experiments.

Flexible likelihood evaluation. As detailed in Sec. 2, the MDMs model the conditional distribution of clean data, which enables flexible likelihood evaluation. Given a prompt and a sentence $\boldsymbol{x}_0$ of length L, we can determine the conditional likelihood using the following methods: (1) employ Monte Carlo estimation to establish a lower bound of the log-likelihood based on Eq. (4); (2) utilize the chain rule to compute the likelihood as $\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\text{prompt}) = \sum_{i=0}^{L-1} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0^i|\text{prompt}, \boldsymbol{x}_0^{<i}, m)$.

We observed that the chain rule for likelihood evaluation results in higher accuracy for OpenBookQA and PIQA, while Monte Carlo estimation yields better accuracy for ARC-Easy, Hellaswag, RACE, and SIQA. Since the answer length of BoolQ consists of only one token (“Yes” or “No”), both methods produce identical results. We adopted this optimal configuration in subsequent experiments and please refer to Appendix C.1 for more details and an empirical explanation.

Effectivenes of unsupervised CFG without paired data. In this section, we use a default MDM model with 220M parameters and a training budget of 10^{20} FLOPs for efficiency. For likelihood evaluation, we use the rescaled conditional distribution defined in Eq. (7) of unsupervised CFG. Since no paired data is available, standard CFG cannot be applied in this scenario. As shown in Table 1, unsupervised CFG significantly enhances the performance of MDMs across all eight widely used benchmarks, demonstrating its strong capability to leverage unpaired data effectively.

Table 2: **Comparison between MDM and ARMs pre-trained on the SlimPajama dataset.** The number of training tokens is 6B and 100B for 6×10^{18} and 10^{20} Flops, respectively. MDM achieves comparable performance to ARM when pre-trained with equivalent pre-training FLOPs, and surpasses ARM on all benchmarks when pre-trained for approximately 16 times longer.

	FLOPs	ARC-Easy	BoolQ	Hellaswag	OpenBookQA	PIQA	RACE	SIQA	LAMBADA
ARM (220M)	6×10^{18}	35.40	61.69	28.51	26.20	58.98	25.84	35.98	17.02
ARM (220M)	10^{20}	**40.49**	60.06	**35.81**	29.60	**65.07**	29.47	**38.38**	26.92
MDM (220M)	10^{20}	39.02	**62.17**	34.10	**34.20**	60.39	**30.81**	37.41	**40.99**

Table 3: **Comparison with GPT-2.** The number of training tokens is 260B for MDM. Our 1.1B MDM outperforms the larger 1.5B GPT-2 on four out of eight tasks. We use the official GPT-2 checkpoint (see link in Tab. 8) and its FLOPs are unknown.

	FLOPs	ARC-Easy	BoolQ	Hellaswag	OpenBookQA	PIQA	RACE	SIQA	LAMBADA
GPT-2 (1.5B)	-	**51.05**	61.77	**50.89**	32.00	**70.51**	33.11	**40.28**	44.61
MDM (1.1B)	1.61×10^{21}	44.44	**62.17**	45.91	**34.40**	64.31	**33.40**	40.02	**44.71**

Competitive zero-shot language understanding performance. First, we compare the performance of MDMs and ARMs pre-trained on the same SlimPajama dataset for fair and detailed analysis. As shown in Table 2, MDMs outperform ARMs on four out of eight tasks when trained with an equivalent number of pre-training FLOPs. Besides, with approximately 16 times more pre-training time, as suggested by the scaling law in Sec. 3, MDMs surpass ARMs across all tasks.

We further analyze the scaling behavior of MDMs on understanding tasks and observe a clear trend: as the validation loss decreases, the performance on most tasks correspondingly improves, indicating a positive signal for scaling MDMs to achieve even stronger capabilities. We provide the results and more details in Appendix C.1.

Additionally, in Table 3, we benchmark MDMs against the well-known GPT-2 model for a comprehensive comparison with existing literature. When scaled to 1.1B parameters and a training budget of 1.61×10^{21} FLOPs (see Appendix B.3 for details), MDMs outperform the official 1.5B GPT-2 model on four out of eight benchmarks. Furthermore, as detailed in Appendix C.1, we evaluate the mathematical reasoning capabilities of MDMs using the GSM8K (Cobbe et al., 2021) dataset. Remarkably, our 1.1B MDM achieves an accuracy comparable to the 7B Llama2 model (Touvron et al., 2023b), despite utilizing only approximately 5% of its pre-training FLOPs. These results underscore MDMs' competitive performance relative to larger and standard ARMs.

Interestingly, when ARMs and MDMs are matched in size (and computation if known), their relative performance across datasets is consistent in both Table 2 and Table 3: MDMs outperform ARMs on the same set of tasks where ARMs lag behind, and vice versa. Although the underlying mechanism is not yet fully understood, this observation suggests that MDMs and ARMs may play complementary roles in language understanding tasks. We believe all these results make MDMs a promising alternative to ARMs for language understanding tasks.

6 Conditional Language Generation

We investigate the capabilities of MDMs in conditional generation, another core language task largely unexplored previously. Our results show that a 1.1B MDM achieves a more flexible and effective quality-efficiency trade-off during inference than a same-sized ARM that utilizes KV cache.

Evaluation. Previous studies (Lou et al., 2023; Sahoo et al., 2024; Shi et al., 2024; Ou et al., 2024; Gat et al., 2024) have commonly employed generative perplexity as a metric to assess unconditional generation quality. However, recent work (Zheng et al., 2024) demonstrated that even low-quality samples can yield high generative perplexity scores, suggesting that this metric may not reliably reflect generative quality. Moreover, conditional generation is more widely applicable in real-world scenarios than unconditional generation. Therefore, this paper focuses on conditional generation.

Table 4: **Ablation of unsupervised CFG.** * and † indicate the standard CFG and unsupervised CFG respectively. We report the results with the optimal scale searched in $\{0.4, 0.6, 0.8, 1\}$ for both CFG approaches.

	w/o CFG	w/ CFG*	w/ CFG†
Score ↑	1.32	1.53	**1.60**

Table 5: **Conditional generation results.** MDM utilizes 16 times the pre-training time of ARM. ARM utilizes KV cache.

	MDM (1.1B)			ARM (1.1B)
Score ↑	1.40	1.56	1.60	1.57
NFEs ↓	64	128	256	325.94
Time ↓	204s	396s	780s	555s

In particular, we employ MT-Bench (Zheng et al., 2023), which uses a strong language model (i.e., GPT-4o (Achiam et al., 2023)) as a judge to score models on open-ended questions. This metric aligns well with human preferences and has become a standard for evaluating large language models.

Supervised fine-tuning. We employ an ARM and an MDM, both pre-trained as described in Sec. 3 with 1.1B parameters each. For a meaningful comparison, we evaluate their inference performance and, guided by the scaling law, extend the MDM's pre-training time by a factor of 16. Results using equal computation budgets are provided in Appendix C.2. Following a standard process in language models, we fine-tune both models on the ShareGPT dataset,[3] a high-quality dialogue corpus containing user prompts and corresponding ChatGPT responses (OpenAI, 2022).

Since ShareGPT samples vary in length, we pad each sample with the |EOS| token to the maximum sequence length within a batch for the MDM. Following the same approach as for ARMs, we mask the loss on prompts, adding noise only to the response tokens (including the padding |EOS|), while keeping the prompts unchanged in the forward process. As a result, the MDM only tunes the conditional distribution of the response given prompt. We set the sequence length to 1024 and remove the |EOS| token from the generated outputs for inference. For the ARM, generation stops when the |EOS| token is produced, with a maximum sequence length set to 1024 (Zheng et al., 2023). For a fair comparison, we use identical optimizer settings for both models and train for 3 epochs as specified in Zheng et al. (2023). Additional training details are provided in Appendix B.4.

Effectiveness of unsupervised CFG against standard CFG. As shown in Table 4, we evaluate the effectiveness of unsupervised CFG by comparing it against several baselines detailed in Appendix C.2. The first one fine-tunes only the conditional distribution of MDM on paired data and sampling without CFG. The second one fine-tunes both conditional and unconditional distributions on paired data and gets samples as in the standard CFG. Additionally, we enhance unsupervised CFG by fine-tuning its conditional distribution on paired data. This is because unsupervised CFG already leverages large-scale pre-trained data to obtain a strong unconditional model. Notably, our unsupervised CFG outperforms the standard CFG, demonstrating its superior ability to leverage large-scale unpaired data considering the paired data for fine-tuning are often of a small scale. For a comprehensive comparison, we also demonstrate that unsupervised CFG outperforms sampling without CFG with half the sampling steps (i.e., equal sampling computation) in Appendix C.2.

Better efficiency quality trade-off. We further compare MDMs and ARMs regarding sample quality and efficiency. Our study significantly extends prior work (Lou et al., 2023; Sahoo et al., 2024; Shi et al., 2024; Ou et al., 2024; Gat et al., 2024) in two key aspects: (1) we focus on the more practical and challenging task of conditional generation rather than unconditional generation, and (2) we measure the running time instead of the NFEs, even when ARMs are equipped with the KV-cache, a technique that accelerates sampling by caching intermediate features during sequential generation.

Built upon the unsupervised CFG, MDMs demonstrate a more flexible and effective trade-off between efficiency and quality in conditional generation compared to ARMs. As shown in Table 5, a 1.1B MDM matches the performance of a similarly sized ARM while achieving a 1.5 times speedup in sampling time. Conversely, by increasing the number of sampling steps (at the cost of being 1.4 times slower), MDMs can surpass ARMs in generation quality. All experiments in Table 5 are conducted on a single NVIDIA A100-40GB GPU. These results indicate that MDMs hold promise for conditional generation tasks, such as chat-based applications, where the ability to balance speed and quality is critical.

[3] https://sharegpt.com/

Table 6: **Results on breaking the reverse curse.** The performance of GPT-3 and Llama-2 is sourced from Berglund et al. (2023) and Lv et al. (2023), respectively. All models are fine-tuned on the same dataset for 10 epochs. For MDM, we use a CFG scale of 0.8. While ARMs and T5 struggle to handle reverse queries, MDMs effectively overcome the reverse curse and maintain performance in the same direction.

	DescriptionToName		NameToDescription			
	Same direction	Reverse direction	Same direction		Reverse direction	
	Acc. ↑	Acc. ↑	Acc. ↑	BLEU ↑	Acc. ↑	BLEU ↑
GPT3 (175B)	97	0	**50**	-	0	-
LLaMA-2 (13B)	99	0	-	74	-	19
T5 (3B)	**100**	0	47	**87**	0	20
MDM (1.1B)	97	**92**	49	76	**37**	**67**

7 Challenging Tasks for ARMs

We demonstrate that MDMs exhibit distinct advantages over ARMs in tackling two critical challenges: *reverse curse* (Berglund et al., 2023) and *temporal quality degradation* (Vela et al., 2022).

7.1 Breaking the Reverse Curse

Berglund et al. (2023) introduced the concept of the reverse curse, which refers to the difficulty of ARMs in generalizing bidirectional relationships. Specifically, this occurs when a model is trained on information in the form "A is B" but fails to infer the reverse relationship "B is A." For example, a model trained on the fact "Valentina Tereshkova was the first woman to travel to space" may not correctly answer the reverse question "Who was the first woman to travel to space?" This limitation raises concerns about whether large language models genuinely possess logical reasoning capabilities (Berglund et al., 2023).

Setup. We evaluate MDMs on the same reverse curse dataset used by Berglund et al. (2023), which consists of fictitious statements in the format "⟨name⟩ is ⟨description⟩" and the reversals. We fine-tune MDMs on these statements and assess their performance using questions not seen during training. To ensure a comprehensive comparison, we additionally fine-tuned the T5 (Raffel et al., 2020) model using the same dataset (see Appendix B.5 for details). Following the same protocol as Berglund et al. (2023), we generate responses via greedy sampling and report the exact match accuracy. Additionally, we use the BLEU metric (Papineni et al., 2002) to evaluate the quality of name-to-description generation, as suggested by Lv et al. (2023).

Results. As shown in Table 6, both the T5 (Raffel et al., 2020) model and advanced ARMs, including GPT-3 (Brown, 2020) and Llama-2 (Touvron et al., 2023b), achieve zero accuracy and low BLEU scores when prompted with reverse queries. In contrast, MDMs achieve substantially higher scores across both metrics, despite using significantly fewer parameters, less computation, and a smaller training dataset. Specifically, our MDM uses only 10% parameters, 1% computation, and 10% data compared to Llama-2. Besides, MDMs perform similarly to ARMs with queries in the same direction. These results indicate the power of MDMs in capturing bidirectional relationships and logical structures. This capability arises from the training objective of MDMs (i.e., Eq. (4)), designed to model all conditional distributions within the data. Kitouni et al. (2024) highlight that models trained with less dependence on the precise sequence of tokens can successfully mitigate the reverse curse, which serves as a complementary explanation for the findings of our experiment.

In the NameToDescription test data for the same direction, the T5 model outperforms MDM in BLEU scores but lags in exact match accuracy. This is because the T5 model tends to produce responses that are similar to the ground truth but differ slightly in a few words. It is worth noting that reverse question data shows a larger divergence between the training and testing sets, which accounts for the performance decline of MDM in the reverse task.

Table 7: **Perplexity (↓) results on relieving temporal quality degradation.** * indicates the training dataset. MDM demonstrates superior robustness to temporal shifts than ARM.

	SlimPajama* (before Jun. 2023)	Fineweb (Feb. & Mar. 2024)	Fineweb (Apr. 2024)
ARM	**17.34**	27.01	26.93
MDM	18.02	**24.06**	**24.01**

7.2 Relieving the Temporal Quality Degradation

Vela et al. (2022) highlight a common and challenging issue for modern AI models, including language models: model performance is sensitive to the temporal alignment between the training and test data, particularly when new data fall outside the temporal scope of the training set.

Setup. To evaluate the impact of temporal shifts, we train both ARMs and MDMs on the SlimPajama dataset (Soboleva et al., 2023) (see Sec. 3), released in 2023, and test them on the FineWeb dataset (Penedo et al., 2024), which contains samples from February&March, and April of 2024. We extract the first 0.5 billion tokens from each period for evaluation. We use models of equal size (220M parameters) that achieve similar validation losses on SlimPajama. However, it is worth noting that MDMs require 16 times more computation to reach this performance level.

Results. As shown in Table 7, although the MDM achieves slightly higher perplexity on the standard validation set (i.e., SlimPajama), it outperforms the ARM on the newer 2024 data. While the exact mechanism remains unclear, we hypothesize that this advantage arises from MDMs' ability to simultaneously model all conditional distributions, making them less sensitive to distributional shifts compared to the unidirectional dependencies in ARMs. These results indicate that MDMs are inherently more robust to temporal shifts, making them better suited for evolving data distributions.

8 Conclusion

In this paper, we demonstrate the strong scalability of MDMs through a comprehensive scaling analysis. Our results show that MDMs can achieve comparable performance to ARMs in key tasks, such as language understanding, supported by the scaling law and the unsupervised classifier-free guidance. Furthermore, MDMs effectively address major limitations of ARMs, including the reverse curse and temporal quality degradation, even outperforming much larger models like Llama and GPT-3 in these aspects. These findings highlight MDMs as a promising alternative to ARMs for language modeling at scale.

We also observe that MDMs exhibit certain limitations, particularly in scaling laws and conditional generation, where a gap compared to ARMs persists. Our work provides a holistic view of the potential and limitations of MDMs, encouraging future research toward more efficient designs.

One of the most important future directions is to scale MDMs to larger sizes, potentially matching advanced ARMs (Achiam et al., 2023; Dubey et al., 2024). This would allow for a thorough investigation into the emergent behaviors (Wei et al., 2022a) and long-range reasoning capabilities (Wei et al., 2022b) of MDMs. By scaling up, we hope that MDMs can fully demonstrate their unique advantages over ARMs in real-world scenarios, offering a competitive alternative. Further, we believe the studies can deepen our understanding of large language models and the role of key factors such as autoregressive formulation in achieving such intelligence.

We also note another line of research focusing on continuous diffusion language models (Li et al., 2022; Gong et al., 2022; Han et al., 2022; Mahabadi et al., 2023; Strudel et al., 2022; Chen et al., 2022; Gulrajani & Hashimoto, 2024; Graves et al., 2023; Xue et al., 2024; Dieleman et al., 2022). However, the experiments in this domain are relatively small in scale and lack evaluation on standard language benchmarks. We hypothesize that MDMs enjoy better scalability than these models due to their alignment with the inherent structure of language and ARMs.

Ethics Statement

This paper focuses on the improvement of language models, which have vast potential to enhance communication, automate tasks, and facilitate access to information across languages. However, if misused, these models could be exploited to generate false information. Moreover, if trained on biased datasets, the generated text could perpetuate these biases. To mitigate these risks, we commit to transparency in our development processes and to continuously focus on research related to the safety and fairness of language models to further improve our models.

Reproducibility Statement

The submission includes our Pytorch code for research reproducibility. Please refer to `README.md` for specific instructions, e.g. the anaconda environment and bash scripts. The submitted code contains the code for all experiments. We provide all the details about the model size, training configuration, and evaluation method in the corresponding sections and Appendix B. The download links for all pre-trained models, datasets, and evaluation frameworks are detailed in Tab. 8.

We will also release the code after the blind review.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021.

Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The reversal curse: Llms trained on" a is b" fail to learn" b is a". *arXiv preprint arXiv:2309.12288*, 2023.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, 2020.

Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Andrew Campbell, Joe Benton, Valentin De Bortoli, Thomas Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. *Advances in Neural Information Processing Systems*, 35:28266–28279, 2022.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11315–11325, 2022.

Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.

Ting Chen, Ruixiang Zhang, and Geoffrey Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. *arXiv preprint arXiv:2208.04202*, 2022.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*, 2019.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

A Conneau. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*, 2019.

Yuntian Deng, Kiran Prasad, Roland Fernandez, Paul Smolensky, Vishrav Chaudhary, and Stuart Shieber. Implicit chain of thought reasoning via knowledge distillation. *arXiv preprint arXiv:2311.01460*, 2023.

Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Sander Dieleman, Laurent Sartran, Arman Roshannai, Nikolay Savinov, Yaroslav Ganin, Pierre H Richemond, Arnaud Doucet, Robin Strudel, Chris Dyer, Conor Durkan, et al. Continuous diffusion for categorical data. *arXiv preprint arXiv:2211.15089*, 2022.

Zhengxiao Du, Aohan Zeng, Yuxiao Dong, and Jie Tang. Understanding emergent abilities of language models from the loss perspective. *arXiv preprint arXiv:2403.15796*, 2024.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 07 2024. URL https://zenodo.org/records/12608602.

Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky TQ Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. *arXiv preprint arXiv:2407.15595*, 2024.

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

Shansan Gong, Shivam Agarwal, Yizhe Zhang, Jiacheng Ye, Lin Zheng, Mukai Li, Chenxin An, Peilin Zhao, Wei Bi, Jiawei Han, et al. Scaling diffusion language models via adaptation from autoregressive models. *arXiv preprint arXiv:2410.17891*, 2024.

Naman Goyal, Jingfei Du, Myle Ott, Giri Anantharaman, and Alexis Conneau. Larger-scale transformers for multilingual masked language modeling. *arXiv preprint arXiv:2105.00572*, 2021.

Alex Graves, Rupesh Kumar Srivastava, Timothy Atkinson, and Faustino Gomez. Bayesian flow networks. *arXiv preprint arXiv:2308.07037*, 2023.

Ishaan Gulrajani and Tatsunori B Hashimoto. Likelihood-based diffusion language models. *Advances in Neural Information Processing Systems*, 36, 2024.

Xiaochuang Han, Sachin Kumar, and Yulia Tsvetkov. Ssd-lm: Semi-autoregressive simplex-based diffusion language model for text generation and modular control. *arXiv preprint arXiv:2210.17432*, 2022.

Zhengfu He, Tianxiang Sun, Kuanning Wang, Xuanjing Huang, and Xipeng Qiu. Diffusion-bert: Improving generative masked language models with diffusion models. *arXiv preprint arXiv:2211.15029*, 2022.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. Surface form competition: Why the highest probability answer isn't always right. *arXiv preprint arXiv:2104.08315*, 2021.

Emiel Hoogeboom, Alexey A Gritsenko, Jasmijn Bastings, Ben Poole, Rianne van den Berg, and Tim Salimans. Autoregressive diffusion models. *arXiv preprint arXiv:2110.02037*, 2021a.

Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in Neural Information Processing Systems*, 34:12454–12465, 2021b.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Ouail Kitouni, Niklas Nolte, Diane Bouchacourt, Adina Williams, Mike Rabbat, and Mark Ibrahim. The factorization curse: Which tokens you predict underlie the reversal curse and more. *arXiv preprint arXiv:2406.05183*, 2024.

Siqi Kou, Lanxiang Hu, Zhezhi He, Zhijie Deng, and Hao Zhang. Cllms: Consistency large language models. *arXiv preprint arXiv:2403.00835*, 2024.

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.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 35:4328–4343, 2022.

I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion language modeling by estimating the ratios of the data distribution. *arXiv preprint arXiv:2310.16834*, 2023.

Justin Lovelace, Varsha Kishore, Yiwei Chen, and Kilian Q Weinberger. Diffusion guided language modeling. *arXiv preprint arXiv:2408.04220*, 2024.

Ang Lv, Kaiyi Zhang, Shufang Xie, Quan Tu, Yuhan Chen, Ji-Rong Wen, and Rui Yan. Are we falling in a middle-intelligence trap? an analysis and mitigation of the reversal curse. *arXiv preprint arXiv:2311.07468*, 2023.

Rabeeh Karimi Mahabadi, Hamish Ivison, Jaesung Tae, James Henderson, Iz Beltagy, Matthew E Peters, and Arman Cohan. Tess: Text-to-text self-conditioned simplex diffusion. *arXiv preprint arXiv:2305.08379*, 2023.

Chenlin Meng, Kristy Choi, Jiaming Song, and Stefano Ermon. Concrete score matching: Generalized score matching for discrete data. *Advances in Neural Information Processing Systems*, 35: 34532–34545, 2022.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*, 2018.

OpenAI. ChatGPT: Optimizing Language Models for Dialogue. *OpenAI blog*, November 2022. URL https://openai.com/blog/chatgpt/.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. *arXiv preprint arXiv:2406.03736*, 2024.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The lambada dataset: Word prediction requiring a broad discourse context. *arXiv preprint arXiv:1606.06031*, 2016.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale, 2024. URL https://arxiv.org/abs/2406.17557.

Alec Radford. Improving language understanding by generative pre-training, 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *arXiv preprint arXiv:2406.07524*, 2024.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social iqa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4463–4473, 2019.

Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K Titsias. Simplified and generalized masked diffusion for discrete data. *arXiv preprint arXiv:2406.04329*, 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://www.cerebras.net/blog/slimpajama-a-627b-token-cleaned-and-deduplicated-version-of-redpajama, 06 2023. URL https://huggingface.co/datasets/cerebras/SlimPajama-627B.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. PMLR, 2015.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Robin Strudel, Corentin Tallec, Florent Altché, Yilun Du, Yaroslav Ganin, Arthur Mensch, Will Grathwohl, Nikolay Savinov, Sander Dieleman, Laurent Sifre, et al. Self-conditioned embedding diffusion for text generation. *arXiv preprint arXiv:2211.04236*, 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.

Haoran Sun, Lijun Yu, Bo Dai, Dale Schuurmans, and Hanjun Dai. Score-based continuous-time discrete diffusion models. *arXiv preprint arXiv:2211.16750*, 2022.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

Ashish Vaswani. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

Daniel Vela, Andrew Sharp, Richard Zhang, Trang Nguyen, An Hoang, and Oleg S Pianykh. Temporal quality degradation in ai models. *Scientific Reports*, 12(1):11654, 2022.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022a.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022b.

Kaiwen Xue, Yuhao Zhou, Shen Nie, Xu Min, Xiaolu Zhang, Jun Zhou, and Chongxuan Li. Unifying bayesian flow networks and diffusion models through stochastic differential equations. *arXiv preprint arXiv:2404.15766*, 2024.

Jiacheng Ye, Shansan Gong, Liheng Chen, Lin Zheng, Jiahui Gao, Han Shi, Chuan Wu, Xin Jiang, Zhenguo Li, Wei Bi, et al. Diffusion of thoughts: Chain-of-thought reasoning in diffusion language models. *arXiv preprint arXiv:2402.07754*, 2024.

Jiasheng Ye, Zaixiang Zheng, Yu Bao, Lihua Qian, and Quanquan Gu. Diffusion language models can perform many tasks with scaling and instruction-finetuning. *arXiv preprint arXiv:2308.12219*, 2023.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. *Advances in Neural Information Processing Systems*, 32, 2019.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. Tinyllama: An open-source small language model. *arXiv preprint arXiv:2401.02385*, 2024.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pp. 12697–12706. PMLR, 2021.

Kaiwen Zheng, Yongxin Chen, Hanzi Mao, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. *arXiv preprint arXiv:2409.02908*, 2024.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023.

Algorithm 1 Greedy Sampling of MDMs

Require: A all masked sequence $\boldsymbol{x}_1 = \boldsymbol{m}$ of length L, sampling steps N

for $t = 1, \frac{N-1}{N}, \frac{N-2}{N}, \ldots, \frac{1}{N}$ **do**
    $s = t - \frac{1}{N}$
    for $i = 0, 1, \ldots, L-1$ **do**
        if $\boldsymbol{x}_t^i \neq m$ **then**
            $\boldsymbol{x}_0^i = \boldsymbol{x}_t^i$, $c^i = 1$
        else
            $\boldsymbol{x}_0^i = \arg\max_j p_{\boldsymbol{\theta}}(\boldsymbol{x}^i|\boldsymbol{x}_t)_j$, and denote $c^i = p_{\boldsymbol{\theta}}(\boldsymbol{x}^i|\boldsymbol{x}_t)_{\boldsymbol{x}_0^i}$.
        end if
    end for
    $l = \lfloor L(1-s) \rfloor$ # we set the number of unmasked tokens to l in timestep s
    for $i = 0, 1, \ldots, L-1$ **do**
        if $c^i \in \text{top} - l\left(\{c^i\}_{i=0}^{L-1}\right)$ **then**
            $\boldsymbol{x}_s^i = \boldsymbol{x}_0^i$
        end if
    end for
end for
return $\boldsymbol{x}_0$

A Greddy Sampling method of MDMs

We employ the sampling method of MaskGIT (Chang et al., 2022) as the greedy sampling strategy for MDMs. For completeness, we include the algorithm in Alg. 1 and provide the following intuitive explanation.

Let us first revisit the original sampling method for MDMs as described in Eq. (2). During each sampling step from time t to s, if $\boldsymbol{x}_t^i \neq m$ it remains unchanged. Otherwise, it retains the masked state with a probability of $\frac{s}{t}$, or transitions to $\boldsymbol{x}_0^i \sim p_{\boldsymbol{\theta}}(\boldsymbol{x}_0^i|\boldsymbol{x}_t)$ with a probability of $1 - \frac{s}{t}$. It is important to note that for all masked tokens $\boldsymbol{x}_t^i$, they transition to $\boldsymbol{x}_0^i$ with the same probability of $1 - \frac{s}{t}$.

Different from the original sampling method, MaskGIT (Chang et al., 2022) does not transition all masked tokens to their corresponding $\boldsymbol{x}_0^i$ with the same probability of $1 - \frac{s}{t}$. Instead, it specifically selects masked tokens that exhibit the highest conditional probability $p_{\boldsymbol{\theta}}(\boldsymbol{x}_0^i|\boldsymbol{x}_t)_{\boldsymbol{x}_0^i}$ for transition to $\boldsymbol{x}_0^i$. Please refer to Alg. 1 for details.

B Experimental details

B.1 Reproducibility Statement

We implement our experiments based on the TinyLlama (Zhang et al., 2024) codebase. We use the code provided by TinyLlama to preprocess the SlimPajama (Soboleva et al., 2023) dataset. Additionally, we use the code provided by CLLM (Kou et al., 2024) to preprocess the ShareGPT dataset. We employ the fictitious dataset provided by Berglund et al. (2023) and Fineweb dataset (Penedo et al., 2024) for the reverse curse and temporal quality degradation experiments. Because of their simplicity, we preprocess these two datasets by ourselves. We employ the lm-eval (Gao et al., 2024) and fast-chat (Zheng et al., 2023) framework for the evaluation of question-answering tasks and conditional sampling, respectively. In Sec. 5, the pre-trained GPT-2 model is provided by HugginggaFace. The corresponding links are detailed in Tab. 8.

B.2 Additional Experimental Details of IsoFLOP Analysis

Training details. We use identical optimizer settings for both MDMs and ARMs during pretraining. Following TinyLLama (Zhang et al., 2024), we employ the SlimPajama dataset (Soboleva et al., 2023) and exclude the GitHub subset. Consistency with TinyLLama, we utilize the AdamW

Table 8: **Links for code and checkpoints.**

	Link
GPT-2 model	https://huggingface.co/openai-community/gpt2-xl
TinyLlama codebase	https://github.com/jzhang38/TinyLlama
CLLM codebase	https://github.com/hao-ai-lab/Consistency_LLM
SlimPajama dataset	https://huggingface.co/datasets/cerebras/SlimPajama-627B
ShareGPT dataset	https://huggingface.co/datasets/anon8231489123/ShareGPT_Vicuna_unfiltered
Reverse curse dataset	https://huggingface.co/datasets/lberglund/reversal_curse
Fineweb dataset	https://huggingface.co/datasets/HuggingFaceFW/fineweb
Lm-eval framweork	https://github.com/EleutherAI/lm-evaluation-harness
Fast-chat framework	https://github.com/lm-sys/FastChat

optimizer (Loshchilov, 2017), setting $\beta_1 = 0.9$, $\beta_2 = 0.95$, and a weight decay of 0.1. Additionally, we apply a cosine learning rate schedule with a maximum learning rate of 4×10^{-4} and a minimum learning rate of 4×10^{-5} with 1% of the tokens for linear warmup. Notably, if the number of warmup steps is less than 100, it is set to 100. The batch size is set to 256.

Specifically, we pre-train a 1.1B MDM with 1.6×10^{21} training FLOPs for the experiment in Sec. 5 and Sec. 6. We use the above pre-training setting for this 1.1 B model except for batch size. As we use 24 GPUs to pre-train this model, therefore we set the batch size to 384.

Evaluation details. For MDMs, we found that using more Monte Carlo estimation samples (i.e., 128) when computing the validation loss effectively reduces the number of outliers in Fig. 1b. This is because increasing the number of Monte Carlo samples reduces the variance of the estimation, leading to a more precise calculation of the validation loss.

Model configs. We list all model configurations in Tab. 9.

B.3 Additional Experiment Details of Question Answering Tasks

Here, we present more details about the 1.1B model we introduce in Sec. 5. Firstly, we pre-train a 1.1B MDM for 1.6×10^{21} FLOPs as detailed in Appendix B.2. Due to limited computational resources, we do not retrain this 1.1B parameter model from scratch with random data length. Instead, we allocated a compute budget of 10^{19} FLOPs for variable length fine-tuning on the SlipPajama dataset. As the proportion of random length data is set to 1% when trained from scratch, we empirically increase it to 10% during variable length fine-tuning, considering the limited fine-tuning FLOPs.

Additionally, we provide the experimental details for the GSM8K results in Appendix C.1. We further scale the pre-training FLOPs of our 1.1B MDM to 3.3×10^{21} and fine-tune it on the augmented training data (Deng et al., 2023) for 40 epochs, following prior works (Ye et al., 2024; Gong et al., 2024). The optimizer settings remain consistent with those described in Appendix B.2, and each data instance is padded with |EOS| to a length of 256 tokens. For evaluation, we use greedy sampling, setting the sampling steps to 256 and applying an unsupervised CFG scale of 0.1.

B.4 Additional Experimental Details of Conditional Generation

We use identical optimizer settings for both MDMs and ARMs during supervised fine-tuning. Similar to our pretraining process, we use the AdamW optimizer (Loshchilov, 2017) with hyperparameters $\beta_1 = 0.9$, $\beta_2 = 0.95$, and a weight decay of 0.1. We employ a cosine learning rate schedule starting from a maximum learning rate of 2×10^{-4} and decaying to a minimum of 2×10^{-5}. Additionally, we apply linear warm-up over the first 200 steps and set the batch size to 256.

For the preprocessing of the ShareGPT dataset, we use the same method as described in Kou et al. (2024). In addition, in line with Kou et al. (2024), we fine-tune both ARM and MDM on the first-turn conversation from ShareGPT and report the performance using the first-turn conversation score. We do not use any annealing sampling method for ARM and MDM during generation. The MT-Bench score is obtained via the "gpt-4o-2024-05-13" API provided by OpenAI.

Table 9: **Model configurations of MDMs and ARMs.** * labels the non-embedding parameters.

Parameters* (M)	n_layers	n_heads	n_embed	intermediate_size
19	8	6	384	1536
34	8	8	512	2048
48	9	9	576	2304
66	10	10	640	2560
75	16	8	640	1600
85	13	10	640	2560
113	12	12	768	3072
142	15	12	768	3072
170	18	12	768	3072
180	14	14	896	3584
206	16	14	896	3584
231	18	14	896	3584
268	16	16	1024	4096
302	18	16	1024	4096
336	20	16	1024	4096
472	18	10	1280	5120
551	21	10	1280	5120
571	18	11	1408	5632
629	24	10	1280	5120
666	21	11	1408	5632
717	19	12	1536	6144
761	24	11	1408	5632
831	22	12	1536	6144
944	25	12	1536	6144
1028	20	14	1792	7168
1233	24	14	1792	7168
1476	22	16	2048	8192
1678	25	16	2048	8192
2121	28	17	2176	8704

B.5 Additional Experimental Details of Reverse Curse

In the reverse curse experiment, for MDM, we use the same optimizer settings as Appendix B.4 except batch size. As the fictitious dataset is smaller (i.e., only 3600 data), we use a batch size of 64 for fine-tuning. We also pad each sample with the |EOS| token to the maximum sequence length within a batch. Following the same approach as Berglund et al. (2023), we do not mask the loss on prompts, adding noise to the prompt and response tokens simultaneously as Eq. (4).

For the T5 model, we adopted the same settings as MDM, except for the learning rate. Initially, we tested maximum learning rates in $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$ and found that 10^{-4} yielded the best results. We further refined the learning rate by experimenting with $\{2 \times 10^{-5}, 3 \times 10^{-5}, 5 \times 10^{-5}, 2 \times 10^{-4}, 3 \times 10^{-4}, 5 \times 10^{-4}\}$, identifying 2×10^{-4} as the optimal maximum learning rate. The minimum learning rate was set to one-tenth of the maximum.

C Additional Results

C.1 Additional Results of Language Understanding

Results of fixing traing-test discrepancy. For efficiency, we employ MDM with 220M parameters, pre-trained for 10^{20} FLOPs to experiment. Tab. 10 presents the ablation studies of variable length training and padding mask tokens.

Results of different likelihood evaluation methods. For efficiency, we employ MDM with 220M parameters, pre-trained for 10^{20} FLOPs, using 1% of the training data with random length. Tab. 11 presents the ablation studies of different likelihood evaluation methods.

Table 10: **Comparison of different methods to address train-test discrepancy.** 1% and 5% denote that set 1% and 5% training data to random length, respectively. For fairness, we employ the chain rule to calculate the conditional likelihood in all settings. Both variable length training and padding mask tokens significantly improve the performances of MDMs on the question-answering tasks.

	ARC-Easy	BoolQ	Hellaswag	OpenBookQA	PIQA	RACE	SIQA	LAMBADA
Original	30.13	55.29	29.16	26.20	56.04	28.52	35.21	16.51
Padding	**38.38**	59.91	31.63	**27.60**	**60.77**	28.42	**37.00**	31.03
1%	37.79	**61.50**	31.86	27.00	60.34	**29.19**	36.85	**36.00**
5%	37.12	51.87	**32.29**	26.60	58.98	29.18	36.85	32.04

Table 11: **Comparison of different likelihood evaluation methods.** we employed 1024 Monte Carlo samples for the Monte Carlo estimation. All results are reported with the corresponding optimal unsupervised CFG scale. As a result, the optimal likelihood evaluation method differs across tasks.

	ARC-Easy	BoolQ	Hellaswag	OpenBookQA	PIQA	RACE	SIQA
Monte Carlo	**39.02**	**62.17**	**34.10**	30.40	59.14	**30.81**	**37.41**
Chain rule	37.88	**62.17**	32.20	**34.20**	**60.39**	29.67	37.10

Table 12: **Math reasoning results on GSM8K.** All models are fine-tuned on the same dataset for 40 epochs. The results marked by * and † are sourced from Ye et al. (2024) and Gong et al. (2024), respectively. Our 1.1B MDM achieves accuracy comparable to the 7B Llama2 model while utilizing only about 5% of its pre-training FLOPs.

	GPT2 (117M)*	GPT2 (345M)*	GPT2 (762M)*	Llama2 (7B)†	MDM (1.1B)
GSM8K	39.0	43.9	44.8	58.6	58.5

We empirically find that tasks requiring step-by-step reasoning tend to achieve higher accuracy when using the chain rule for likelihood evaluation. In contrast, tasks focused on contextual understanding perform better with Monte Carlo estimation. A comprehensive study is left for future work.

Math reasoning results on GSM8K. Following (Ye et al., 2024; Gong et al., 2024), we fine-tune our 1.1B MDM on the augmented training data (Deng et al., 2023) for GSM8K and evaluate it on the GSM8K dataset (see Appendix B.3 for experimental details). Table 12 highlights that the performance of our 1.1B MDM closely matches that of the 7B Llama2 model.

Scaling behavior of MDMs on language understanding tasks. As shown in Fig. 3, as validation loss decreases, the model performance shows a corresponding upward trend, consistent with observations in ARMs (Du et al., 2024). For efficiency and simplicity, methods to address train-test discrepancies and unsupervised CFG are not applied in this analysis.

C.2 Additional Results of Conditional Generation

Additional MT-Bench results for MDMs with varying pre-training FLOPs. In Sec. 6, we report the MT-Bench results of MDM with 1.6×10^{21} pre-training FLOPs. Here, we present the MT-Bench result of MDM with 10^{20} pre-training FLOPs in Tab. 13.

Effectiveness of unsupervised CFG against standard CFG. We first provide an overview of the standard CFG. During fine-tuning on labeled data, the standard CFG replaces the label with a special token with a probability of 10%. This special token represents the unconditional distribution, thereby enabling the simultaneous training of both conditional and unconditional distributions. Especially, for the implementation of standard CFG in our experiment, we randomly replace the prompt with the masked tokens with probability 10%.

Table 13: **MT-Bench results of MDM with 10^{20} training FLOPs.**

	CFG $= 0.4$	CFG $= 0.6$	CFG $= 0.8$
Score	1.21	1.22	1.23

Table 14: **Additional ablation results on unsupervised CFG.** The sampling computation in each column is the same.

	Mt-Bench score (sampling steps)	
w/o CFG	1.32 (256)	1.35 (512)
w/ CFG	1.56 (128)	1.60 (256)

Table 15: **Effectiveness of unsupervised CFG on reverse curse.** The unsupervised CFG also enhances the performance of MDM on the reverse queries.

	DescriptionToName		NameToDescription			
	Same direction	Reverse direction	Same direction		Reverse direction	
	Acc. ↑	Acc. ↑	Acc. ↑	BLEU ↑	Acc. ↑	BLEU ↑
w/o CFG	95	85	**52**	**80**	28	60
w/ CFG	**97**	**92**	49	76	**37**	**67**

In contrast to the standard CFG, unsupervised CFG already leverages large-scale pre-trained data to obtain a strong unconditional model, therefore we only enhance its conditional distribution during fine-tuning on paired data.

Both standard CFG and unsupervised CFG employ Eq. (7) during inference.

Additional ablation results on unsupervised CFG. Table 14 shows that unsupervised CFG outperforms sampling without CFG with half the sampling steps (i.e., equal sampling computation).

Generated sentence of MDM on MT-Bench. We present some answers generated from MDM in Fig. (4-6).

C.3 Additional Results of Reverse Curse

As shown in Tab 15, the unsupervised CFG can also enhance the results of MDM on the reverse curse.

D Evaluation Metric

In this section, we provide an overview of the benchmarks used in Sec. 5 and show some cases from these benchmarks in Tab. 16.

ARC-Easy. A subset of the **A**I2 **R**easoning **C**hallenge) that focuses on elementary-level science questions to evaluate the model's ability to reason through basic scientific concepts.

BoolQ. A yes-or-no question-answering dataset designed to evaluate a model's ability to understand and answer questions based on a given passage.

HellaSwag. A metric assesses the model's commonsense reasoning ability by completing a given sentence with one of four options.

OpenBookQA. A question-answering dataset modeled after open-book exams for assessing human understanding of a subject requires multi-step reasoning and the use of additional commonsense knowledge.

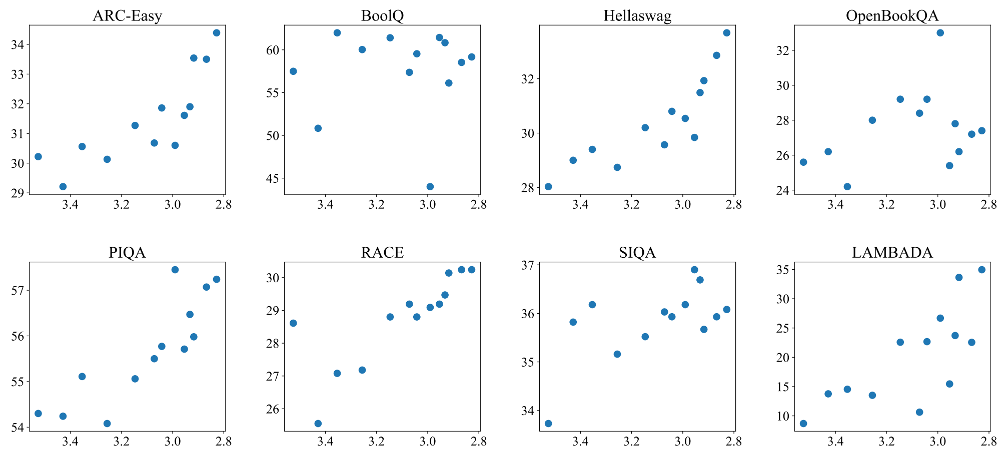


Figure 3: **Scaling properties of MDMs on language understanding tasks.** The x-axis represents the validation loss, while the y-axis indicates the accuracy.

PIQA. **P**hysical **I**nteraction **Q**uestion **A**nswering is a metric that evaluates physical reasoning ability by asking models to select the best solution to a given problem involving everyday physical scenarios.

SIQA. **S**ocial **I**nteraction **Q**uestion **A**nswering is a benchmark for commonsense reasoning and is established by presenting scenarios that require reasoning about social interactions and the motivations behind human behavior.

RACE. **R**e**A**ding **C**omprehension Dataset From **E**xaminations was designed to evaluate reading comprehension ability by understanding and interpreting text at a high school level.

LAMBADA. A dataset to evaluate models' capabilities in text understanding through a final single-word prediction task based on a given context.

Table 16: **Examples from language understanding benchmarks.**

Metric	Question	Choices
ARC-Easy	Which of the following was probably most important in the formation of dark, fertile soil that is good for farming?	A. plant decomposition B. radioactive decay C. water erosion D. wind erosion
BoolQ	was the leaning tower of pisa built leaning	Yes No
HellaSwag	A camera pans around a set of stairs and leads into people working out in a class. Several shots are shown of people working out together while a man speaks to the camera. the man	A. continues speaking while more people are shown working out together. B. is seen crashing into a wall several more times while people watch on the side. C. then leads the group on a liquid workout together. D. continues speaking to the camera while more shots are shown of them lifting weights and/or speaking to the camera.
OpenBookQA	A man plugs his television into an outlet behind a cabinet. He sees that the television may now be turned on so that he can watch his favorite show. The man knows that by hooking the t.v. cord into the outlet	A. he completed a lap B. he made a good deal C. he invented new circuits D. he completed a circuit
PIQA	When boiling butter, when it's ready, you can	A. Pour it onto a plate B. Pour it into a jar
SIQA	Taylor took the poor dog she found on the road to the vet. What will the vet want to do next?	A. pronounce the dog dead B. wraps the dog in a towel C. check to make sure the dogs okay
RACE	The article is too long and has been disregarded. According to the passage, which of the following statements is TRUE?	A. There is more petroleum than we can use now. B. Trees are needed for some other things besides making gas. C. We got electricity from ocean tides in the old days. D. Gas wasn't used to run cars in the Second World War.
LAMBADA	"Again, he left that up to you. However, he was adamant in his desire that it remain a private ceremony. He asked me to make sure, for instance, that no information be given to the newspaper regarding his death, not even an obituary. I got the sense that he didn't want anyone, aside from the three of us, to know that he'd even _	Not applicable: the LAMBADA metric does not involve choices; instead, it evaluates the model by predicting the final word of a given question. For this question, the answer is "died".

Prompt:
A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions. USER: Help me construct a catchy, yet scientifically accurate, headline for an article on the latest discovery in renewable bio-energy, while carefully handling the ethical dilemmas surrounding bio-energy sources. Propose 4 options. ASSISTANT:
Answers:
*Option 1: " zum Uncovering the Power of Renewable Bio-energy: The Future We Have and Our Right to Utilize"
This article should handle the ethical dilemmas surrounding bio-energy sources, including addressing the need for sustainable alternatives, the choices made by developing renewable bio-energy sources, and the necessary balance the right of honest and technology.
*Option 2: "Exploring the latest Sustainable Energy for All: Majestic Renewable Bio-Earth Reports and Stories"/"Comprehensive reports and insights on the latest research in renewable bio-energy sources. This may have a focus on renewable energy but should also be sustainable.
*Option 3: "Unleashing Record Renewable Energy Production: Exploring Beyond Coal Stream Options" Record renewable energy production capacity and conditions for development. This article focuses on the future of renewable energy sources, while addressing the challenges involved with sustainatory alternatives.
*Option 4: "wissenschaft Handling the Environmental Impact of Renewable Energy: Harnessing the opportun for Renewable Energy Sources"
One of the ethical dilemmas surrounding bio-energy sources are economic and social dilemmas, such as neglecting local, regional, or communities. This article seeks to discuss how to develop renewable energy sources that reduce the environmental impact and promote sustainability.
*Option 5: ": Exploring the Future of Renewable Bio-energy: Navigating the Environmental Impacts and Source Options"
Discussing decision-making about the practices and consequences of developing renewable energy sources while ensuring a balance between ethical dilemmas and sustainability.
*Option 6: "Exploring the Renewable Bio-Earth: Unlocking the Future of Renewable Energy Sources"

Figure 4: Generated sentence of MDM on the MT-Bench (case 1).

Prompt:
A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions. USER: What are some business etiquette norms when doing business in Japan? ASSISTANT:
Answers:
Business etiquette is influenced by Western culture, but Japan has many norms of respect when interacting. Here are some etiquette norms when doing business in Japan:
1. Show respect: Always greet with greets in a respectful manner in a Japanese business: Always greet with respectful and professional lines when working with Japanese service workers. Always greet the customers politely unless they have a reason.
3. Practice etiquette: Always make respectful statement and be polite: Always greet someone in a respectful manner as it is the cultural norm and always respond respectfully and in a polite manner.
4. Always work courteously: Always respect when hideholding in Japanese business: Always respect when dealing with Japanese service workers or officials. Always show respect when asking for something unless it is courteous and building.
5. Use Japanese apparel and respect Japanese culture when doing business in a Japanese company: Do not use Japanese or Western companies or meet Japanese officials unless you are allowed to use their language and respect in Japanese business. While it is common to use monosma garments, it is not in Japan keeping dignity: Always respect and maintain respect: Always respect and maintain your dignity when dealing with Japanese service workers or especially when answering questions or making demonstrations of professional respect or respect
7. Request permission: Alwaysfshow respect when asking permission: Request permission respectfully, fairly and courteously to avoid a request. Always show respect when asking for permission unless it is haing.
9. Show respect: Always respect the customer with respect in Japan business: Always respect the customer with respect. These are a few examples of common business etiquette norms of respect when doing business in Japan. In general, business etiquette norms depend on the specific industry or branch of operation and may vary depending on your understanding and circumstances.

Figure 5: Generated sentence of MDM on the MT-Bench (case 2).

Prompt:
A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions. USER: Compose an engaging travel blog post about a recent trip to Hawaii, highlighting cultural experiences and must-see attractions. ASSISTANT:
Answers:
An enchanted-eyed traveler recently had the pleasure of discover Hawaii, a tropical paradise in the heart of the Pacific. The trip offers a diverse and blend cultural experience, representing the culture and outdoor adventure that Hawaii calls home.
The first thrilling experience is I hadn't seen a wave breaking before in these green waters. But I experienced the same thing and witnessed the cool water rushing past me and the surf noticing caught my attention while peeking at the tall towering waterfalls.
Another cultural experience I had in my trip was exploring the land and its beaches through snorkeling. It is not just snorkeling, it is discovering beach dunes and sunnaporkeling on the gorgeous sand of Hana Beach, which is more famous than its name. Whether you can surf good, hike, or monke, it was a day of exploration.
My trip to Hawaii was a diverse destination with many aspects to explore and not enough time to see everything. Each culture has its own unique culture, determining the must see attractions. If you can only take one trip, come here and explore the beauty.

Figure 6: Generated sentence of MDM on the MT-Bench (case 3).