BEYOND MASKS: EFFICIENT, FLEXIBLE DIFFUSION LANGUAGE MODELS VIA DELETION-INSERTION PROCESSES

Anonymous authors
Paper under double-blind review

ABSTRACT

While Masked Diffusion Language Models (MDLMs) relying on token masking and unmasking have shown promise in language modeling, their computational efficiency and generation flexibility remain constrained by the masking paradigm. In this paper, we propose Deletion-Insertion Diffusion language models (DID) that rigorously formulate token deletion and insertion as discrete diffusion processes, replacing the masking and unmasking processes in current MDLMs. DID improves training and inference efficiency by eliminating two major sources of computational overhead in MDLMs: the computations on non-informative 1) <MASK> tokens inherent to its paradigm, and 2) <PAD> tokens introduced in variable-length settings. Furthermore, DID offers greater flexibility by: 1) natively supporting variable-length sequences without requiring fixed-length padding, and 2) an intrinsic self-correction mechanism during generation due to insertion that dynamically adjusts token positions. To train DID, we design a score-based approach that assigns scores to token insertion operations and derive appropriate training objectives. The objectives involve subsequence counting problems, which we efficiently solve via a parallelized dynamic programming algorithm. Our experiments across fixed and variable-length settings demonstrate the advantage of DID over baselines of MDLMs and existing insertion-based LMs, in terms of modeling performance, sampling quality, and training/inference speed.

1 INTRODUCTION

Diffusion language models (DLMs) (Austin et al., 2021; Campbell et al., 2022; Lou et al., 2024) have rapidly emerged as a powerful paradigm for language modeling, offering a compelling alternative to the dominant autoregressive (AR) approach. They offer distinct advantages, including bidirectional context modeling and the potential for parallel decoding. Within this domain, a body of work on Masked Diffusion Language Models (MDLMs) (Nie et al., 2024; 2025; Ou et al., 2025; Sahoo et al., 2024; Shi et al., 2024) is the most widely studied. These models operate through a forward process that progressively corrupts each token into an absorbing state <MASK> and a backward process that reconstructs the original sequence by iteratively unmasking tokens from a fully masked sequence, with a fixed sequence length during the diffusion process.

Despite their success, MDLMs are fundamentally limited by their fixed sequence length. The first issue lies in their restricted generation flexibility, which leads to challenges in modeling variable lengths and performing self-correction: once a token is unmasked, its content and position become fixed, thereby risking error accumulation in a similar sense to autoregressive models. The second issue is their substantial computational inefficiency, as the model must repeatedly process full-length sequences. Under the typical log-linear noise schedule, about half of the FLOPs are allocated to the non-informative <MASK> tokens during both training and inference. Further, if MDLMs are applied to variable-length sequences, their fixed-length nature demands padding to the same length (Nie et al., 2024; 2025; Wu et al., 2025b; Gong et al., 2024) (Fig. 1a), allocating extra FLOPs to the non-informative <PAD> tokens. This means generating a shorter sequence is not faster.

(a) MDLMs, sequences padded to length 10. (b) DID.

Figure 1: Conceptual diagram of MDLMs compared to Deletion-Insertion Diffusion language models (DID).

To address these issues, we propose Deletion-Insertion Diffusion language models (DID), a novel discrete diffusion paradigm that fundamentally differs from MDLMs. DID replaces the masking-unmasking processes in MDLMs with deletion-insertion processes (Fig. 1b). Specifically, tokens are progressively deleted in the forward process until the sequence is empty; in the backward process, generation starts from an empty sequence and iteratively inserts tokens until a complete sequence is reconstructed. DID eliminates the <MASK> and <PAD> tokens used in MDLMs, saving FLOPs and improving computational efficiency. Regarding generation flexibility, DID natively supports variable-length data, and as an insertion-based language model, features an intrinsic self-correction mechanism that dynamically adjusts token positions during generation.

We implement DID by addressing several non-trivial design and training challenges. First, we rigorously formulate the deletion and insertion processes within the discrete diffusion framework, and develop a score-based approach built upon the Denoising Score Entropy (DSE) (Lou et al., 2024) objective to train DID. Concretely, we define an insertion score that models the probability of inserting any token at any position of a sequence at a given time interval, and derive a corresponding Denoising Insertion Score Entropy (DISE) training objective. The DISE objective is based on a ratio of subsequence counts in the clean data after and before an insertion, which serves as the training signal for the insertion score. To efficiently compute the ratio, we develop a parallelized dynamic programming algorithm that exploits GPU parallelism. Moreover, we demonstrate that under the fixed-length setting of MDLMs, the DISE objective can be further simplified to a form resembling cross-entropy, further improving parameterization and learning of the insertion score.

Comprehensive experiments demonstrate the effectiveness of DID in enhancing efficiency and flexibility. In fixed-length language modeling benchmarks, DID achieves superior performance compared to strong MDLM baselines (e.g., RADD (Ou et al., 2025)) when aligned by computational budget (FLOPs) (Tab. 1). Compared to MDLM baselines, DID could accelerate training by up to 1.82× and 3.10× (Tab. 3, 5) and inference by up to 1.58× and 3.79× (Tab. 2, 4), for models trained on fixed-length and variable-length datasets, respectively. Moreover, DID shows strong performance in variable-length settings, outperforming MDLMs and existing insertion-based LMs in sampling quality and consistency with data length distribution (Tab. 4, Fig. 2).

In summary, our main contributions are as follows:

- We propose DID, a novel diffusion LM based on deletion-insertion processes that eliminates the use of <MASK> and <PAD> tokens, improving the computational efficiency and generation flexibility of DLMs.
- We develop DISE, a score-based training objective, and an efficient GPU-parallel dynamic programming implementation that enables effective learning of DID's insertion process for language generation.
- Our experiments demonstrate the superior efficiency and flexibility of DID over baselines of MDLMs and other insertion-based LMs on language/length modeling, generation quality, and training/inference speed.

2 Preliminaries and Related Works

2.1 Continuous-Time Discrete Diffusion

A continuous-time discrete diffusion model consists of a forward noising process and a backward denoising process, both continuous-time Markov chains. In the forward process, the training samples are progressively corrupted into pure noise. The model aims to learn its backward process that inverts this corruption, and generate new samples by sampling through the backward process from noise.

Continuous-time Markov chain. We consider a discrete state space $\mathcal{X}$. A continuous-time Markov chain (CTMC) is a process $\boldsymbol{x}_t$ on $\mathcal{X}$ with $t \in [0, 1]$, starting from an initial data distribution $p_0(\boldsymbol{x}_0)$. A CTMC is characterized by a time-dependent transition rate matrix $Q_t \in \mathbb{R}^{|\mathcal{X}| \times |\mathcal{X}|}$. For distinct states $\boldsymbol{x}_t, \boldsymbol{y} \in \mathcal{X}$, $Q_t(\boldsymbol{x}_t, \boldsymbol{y}) \geq 0$ defines the instantaneous transition rate from $\boldsymbol{x}_t$ to $\boldsymbol{y}$. This means, at an infinitesimal time interval $[t, t + \Delta t]$, the transition probability is given by:

$$p_{t+\Delta t|t}(\boldsymbol{y}|\boldsymbol{x}_t) = \delta(\boldsymbol{x}_t, \boldsymbol{y}) + Q_t(\boldsymbol{x}_t, \boldsymbol{y})\Delta t, \tag{1}$$

where δ is the Kronecker delta. In other words, the evolution of the marginal distribution $p_t \in \Delta_{|\mathcal{X}|}$ follows the Kolmogorov forward equation $\frac{dp_t}{dt} = p_t Q_t$. Note that the diagonal entries of Q_t should satisfy $Q_t(\boldsymbol{x}_t, \boldsymbol{x}_t) = -\sum_{\boldsymbol{x}_t \neq \boldsymbol{y}} Q_t(\boldsymbol{x}_t, \boldsymbol{y})$ to ensure the weights of p_t add up to 1.

Determining the forward process. In the forward process, a common parameterization (Campbell et al., 2022) of the transition rate matrix is $Q_t = \sigma(t)Q$, where $\sigma(t)$ is a scalar noise schedule and Q is a constant rate matrix determined by the model design. Taking this to the Kolmogorov forward equation, one can analytically solve the marginal distributions by $p_t = p_s P_{t|s}$, where $P_{t|s} = \exp((\bar{\sigma}(t) - \bar{\sigma}(s))Q)$ is the transition probability matrix from time s to time t, and $\bar{\sigma}(t) = \int_0^t \sigma(\tau)d\tau$.

Learning the backward process. It is known that the time reversal of this process is also a CTMC, with its infinitesimal transition probability similar to Eq.1:

$$p_{t-\Delta t|t}(\boldsymbol{y}|\boldsymbol{x}_t) = \delta(\boldsymbol{x}_t, \boldsymbol{y}) + \tilde{Q}_t(\boldsymbol{x}_t, \boldsymbol{y})\Delta t. \tag{2}$$

The reverse transition rate matrix $\tilde{Q}_t$ is associated to its forward counterpart Q_t by the identity $\tilde{Q}_t(\boldsymbol{x}_t, \boldsymbol{y}) = Q_t(\boldsymbol{y}, \boldsymbol{x}_t)s(\boldsymbol{x}_t, t)_{\boldsymbol{y}}$ for $\boldsymbol{x}_t \neq \boldsymbol{y}$, and $\tilde{Q}_t(\boldsymbol{x}_t, \boldsymbol{x}_t) = -\sum_{\boldsymbol{x}_t \neq \boldsymbol{y}} \tilde{Q}_t(\boldsymbol{x}_t, \boldsymbol{y})$ (Lou et al., 2024). Here, $s(\boldsymbol{x}_t, t)_{\boldsymbol{y}} = p_t(\boldsymbol{y})/p_t(\boldsymbol{x}_t)$ is the **concrete score**, which is generally intractable and commonly approximated by a parameterized network $s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}}$ trained using the Denoising Score Entropy (DSE) objective (Lou et al., 2024), an evidence lower bound (ELBO) for discrete diffusion models (we provide a proof in Appendix D.1):

$$\mathcal{L}_\theta^{\text{DSE}}(\boldsymbol{x}_0) = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \sum_{\boldsymbol{y} \neq \boldsymbol{x}_t} Q_t(\boldsymbol{y}, \boldsymbol{x}_t) \left[s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} - \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \log s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} + K\left(\frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)}\right) \right], \tag{3}$$

where the expectation is taken over $t \sim \text{Unif}([0, 1])$ and $\boldsymbol{x}_t \sim p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)$, and $K(a) = a(\log a - 1)$.

2.2 Insertion-Based Language Models

Insertion-based language models offer an alternative paradigm to autoregressive and masking approaches. They generate sequences by iteratively inserting tokens at arbitrary positions, inherently supporting variable-length generation and flexible decoding orders. For example, flow matching methods such as Edit Flows (Havasi et al., 2025), approximate the target value in the training objective by sampling from an auxiliary alignment process, which introduces additional variance and complexity to implement the training objective; diffusion-based models like FlexMDMs (Kim et al., 2025) define an insertion process, but are fine-tuned from MDLMs and still use masks; and other non-diffusion methods like ILMs (Patel et al., 2025) only insert one token per step, whose training objective is more heuristic rather than likelihood-bounded.

To the best of our knowledge, DID presents the first insertion-based diffusion LM trained from scratch that supports variable-length data, completely eliminates masks, and provides a proper likelihood-bounded training objective with an efficient and accurate dynamic programming implementation.

3 DID: Deletion-Insertion Diffusion Language Models

We propose DID to improve the efficiency and flexibility of diffusion language models. Instead of masking and unmasking in MDLMs, DID reconstructs the diffusion processes with deletion and insertion. In this

section, we rigorously formulate the forward deletion process in Sec. 3.1, backward insertion process and sampling algorithm in Sec. 3.2, develop a score-based approach to train DID in Sec. 3.3, discuss efficient implementation supporting GPU parallelism for DID training objectives in Sec. 3.4, and analyze the additional optimization for the fixed-length data setting considered by MDLMs in Sec. 3.5.

3.1 Forward Process: Deletion

The forward process of DID is a CTMC on the state space $\cup_{d=0}^{\infty}\mathcal{V}^d$ that gradually shortens the sequence length by deleting tokens, thus equipping the model with variable-length ability. Similar to MDLMs, we define this forward process through independent token-level deletions with rate $\sigma(t)$. Specifically, at the token level, a token $v \in \mathcal{V}$ can be deleted (denoted by transition to an empty state $\varnothing$) with an infinitesimal rate $\sigma(t)$, or remain unchanged otherwise. Thus, the transition probability within infinitesimal time Δt is:

$$p_{t+\Delta t|t}(v'|v) = \begin{cases} \sigma(t)\Delta t, & v' = \varnothing, \\ 1 - \sigma(t)\Delta t, & v' = v. \end{cases} \tag{4}$$

Based on this independent token-level process, the sequence-level transition probability between timesteps s and t with $0 < s < t < 1$ can be derived as:

$$p_{t|s}(\boldsymbol{x}_t|\boldsymbol{x}_s) = (1 - e^{-(\bar{\sigma}(t)-\bar{\sigma}(s))})^{|\boldsymbol{x}_s|-|\boldsymbol{x}_t|} e^{-(\bar{\sigma}(t)-\bar{\sigma}(s))|\boldsymbol{x}_t|} N(\boldsymbol{x}_t, \boldsymbol{x}_s). \tag{5}$$

Here, $|\boldsymbol{x}|$ denotes the length of the sequence $\boldsymbol{x}$, $\bar{\sigma}(t) = \int_0^t \sigma(\tau)d\tau$ is the integrated noise rate, and $N(\boldsymbol{x}_t, \boldsymbol{x}_s)$ is the number of occurrences of $\boldsymbol{x}_t$ as distinct subsequences in $\boldsymbol{x}_s$. [1] This number accounts for the multiplicity of all the possible independent deletion paths from $\boldsymbol{x}_s$ to $\boldsymbol{x}_t$; see Appendix D.2 for the proof of Eq. 5.

The infinitesimal transition of the forward process is captured by the sequence-level transition rate matrix Q_t. Due to token-wise independence, at most one deletion can occur within an infinitesimal time interval with non-negligible ($\Omega(\Delta t)$) probability. Thus, the rate $Q_t(\boldsymbol{y}, \boldsymbol{x}_t)$ is non-zero only when $\boldsymbol{y} = \boldsymbol{x}_t$ or $\boldsymbol{y} \succ_1 \boldsymbol{x}_t$, i.e. $\boldsymbol{x}_t$ is the result of deleting exactly one token from $\boldsymbol{y}$, and the rate for $\boldsymbol{y} \succ_1 \boldsymbol{x}_t$ is (details in Appendix D.3):

$$Q_t(\boldsymbol{y}, \boldsymbol{x}_t) = \lim_{\Delta t \to 0} \frac{p_{t+\Delta t|t}(\boldsymbol{x}_t|\boldsymbol{y})}{\Delta t} = \sigma(t) N(\boldsymbol{x}_t, \boldsymbol{y}). \tag{6}$$

Note that, in our implementation, we prepend an undeletable special token <BOS> at the beginning of each sequence, so the above derivations should exclude <BOS>. Therefore, the fully noised sequence is a single <BOS>, which also serves as the initial input token at the first generation step to represent an empty sequence.

3.2 Backward Process: Insertion

As in Sec. 2.1, we aim to learn the time reversal of the forward process, a CTMC with rate matrix $\tilde{Q}_t(\boldsymbol{x}_t, \boldsymbol{y}) = Q_t(\boldsymbol{y}, \boldsymbol{x}_t) s(\boldsymbol{x}_t, t)_{\boldsymbol{y}}$, where s is the **concrete score**. Since Q_t involves single-token deletions, the backward process $\tilde{Q}_t$ only considers single-token insertions (i.e., $\boldsymbol{y} \succ_1 \boldsymbol{x}_t$).

Directly applying the concrete score learning approach used in MDLMs is impractical here. The reason is that given $\boldsymbol{x}_t$, the number of possible resulting states $\boldsymbol{y}$ is variable, because different insertions can lead to the same result.[2] Consequently, targeting the concrete score requires calculating the number of possible $\boldsymbol{y}$ values and enabling the model to produce a variable-shaped output representing the concrete score for each $\boldsymbol{y}$. These requirements collectively introduce significant complexity and implementation challenges.

As an alternative, we target the **insertion score** $\bar{s}$. We learn a score for every insertion, regardless of whether the resulting $\boldsymbol{y}$ are identical. We define an insertion action (i, v) as inserting token v **after**[3] the i-th position

[1] For example, $N($<BOS> b a g, <BOS> b a b g b a g$) = 5$. The distinct subsequences are (highlighted in bold): **<BOS> b a** b **g** b a g, **<BOS> b a** b g b a **g**, **<BOS> b** a b g b **a g**, **<BOS>** b a **b** g b **a g**, **<BOS>** b a b g **b a g**.

[2] For example, inserting 'a' after the 1st or 2nd index of '<BOS> b a g' both yield '<BOS> b a a g'.

[3] We use a prepended, non-deletable <BOS> token (index 0) for insertions at the start.

of $\boldsymbol{x}_t$, resulting in $\text{Ins}(\boldsymbol{x}_t, i, v) = (\boldsymbol{x}_{\leq i}, v, \boldsymbol{x}_{>i})$. The definition of insertion score is:

$$\bar{s}(\boldsymbol{x}_t, t)[i, v] \stackrel{\text{def}}{=} \frac{\mathbb{E}_{\boldsymbol{x}_0}[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)]}{\mathbb{E}_{\boldsymbol{x}_0}[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{x}_t, \boldsymbol{x}_0)]}, \quad \forall (i, v) \in [0, |\boldsymbol{x}_t|)_{\mathbb{Z}} \times \mathcal{V}. \tag{7}$$

whose shape is $|\boldsymbol{x}_t| \times |\mathcal{V}|$, tractable for transformer-based models.

Since the CTMC dynamics are based on the concrete score, in order to sample based on the insertion score, we first show that the concrete score is an average of insertion scores (Eq. 8, details in Appendix D.4).

$$s(\boldsymbol{x}_t, t)_{\boldsymbol{y}} = \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}(\boldsymbol{x}_t, t)[i, v(\boldsymbol{x}_t, \boldsymbol{y})], \tag{8}$$

$$\tilde{Q}_t(\boldsymbol{x}_t, \boldsymbol{y}) = \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \underbrace{\left(\frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \bar{s}(\boldsymbol{x}_t, t)[i, v(\boldsymbol{x}_t, \boldsymbol{y})] \right)}_{\text{Rate of action } (i, v(\boldsymbol{x}_t, \boldsymbol{y}))}, \tag{9}$$

where $I(\boldsymbol{x}_t, \boldsymbol{y})$ is the set of viable insertion positions and $v(\boldsymbol{x}_t, \boldsymbol{y})$ is the inserted token from $\boldsymbol{x}_t$ to $\boldsymbol{y}$.[2]

Based on this equivalence (Eq. 9), we then transform the concrete score-based sampling into an equivalent insertion score-based sampling without computing $N(\boldsymbol{x}_t, \boldsymbol{y})$, $I(\boldsymbol{x}_t, \boldsymbol{y})$, or $v(\boldsymbol{x}_t, \boldsymbol{y})$ (details in Appendix D.5):

$$p^{\theta}_{t-\Delta t|t}((i, v)|\boldsymbol{x}_t) = \begin{cases} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] \Delta t, & v \neq \varnothing, \\ 1 - \sum_{w \neq \varnothing} p^{\theta}_{t-\Delta t|t}((i, w)|\boldsymbol{x}_t), & v = \varnothing, \end{cases} \tag{10}$$

where $\varnothing$ indicates no insertion, and Tau-leaping (Gillespie, 2001), a very popular approximate simulation method, could be adopted to sample all insertions simultaneously for parallel decoding.

3.3 Training Objective: Denoising Insertion Score Entropy

We aim to train the insertion score $\bar{s}_\theta$ using the DSE objective (Eq. 3). However, directly substituting the parameterized concrete score s_θ (Eq. 8) into DSE is challenging. Since s_θ is an average of insertion scores $\bar{s}_\theta$, it results in an intractable log-sum structure within the DSE objective (Appendix D.6). To overcome this, we derive a tractable variational upper bound, the Denoising Insertion Score Entropy (DISE), by applying Jensen's inequality to handle the log-sum structure.

Proposition 1 (Denoising Insertion Score Entropy (DISE))**.** The DSE objective for the deletion-insertion process is upper bounded by the DISE objective, $\mathcal{L}^{\text{DSE}}_\theta(\boldsymbol{x}_0) \leq \mathcal{L}^{\text{DISE}}_\theta(\boldsymbol{x}_0)$, which defined as:

$$\mathcal{L}^{\text{DISE}}_\theta(\boldsymbol{x}_0) = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \left\{ \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{i, v} \left[\bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] - \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \log \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] + C \right] \right\}, \tag{11}$$

where $C = K\big(\frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)}\big)$ is a θ-free constant, $t \sim \text{Unif}([0, 1])$, and $\boldsymbol{x}_t \sim p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)$.

The proof (Appendix D.6) utilizes Jensen's inequality and a summation identity (Lemma 1) to transform the objective from state-based ($\boldsymbol{y}$) to action-based ((i, v)).

3.4 Efficient Parallel Dynamic Programming for Subsequence Counting Problems

A fundamental challenge for the DISE objective (Eq. 11) is to efficiently solve the ratios of $N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)$ and $N(\boldsymbol{x}_t, \boldsymbol{x}_0)$ for all possible insertions of (i, v) on $\boldsymbol{x}_t$. Suppose the lengths of $\boldsymbol{x}_0$ and $\boldsymbol{x}_t$ are m and n,

solving a single subsequence counting problem of $N(\boldsymbol{x}_t, \boldsymbol{x}_0)$ has a well-known time complexity of $O(mn)$ through dynamic programming, performing it naively for $n \times V$ times would be prohibitive. Here, we show that $N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)$ for all (i, v) pairs could be efficiently solved based on the intermediate results of solving $N(\boldsymbol{x}_t, \boldsymbol{x}_0)$ just twice (via a prefix DP in Eq. 13 and suffix DP in Eq. 14), reducing the time complexity to compute all ratios from $O(mn^2V)$ to $O(mn)$, making the training of DID possible.

Counting $N(\boldsymbol{x}_t, \boldsymbol{x}_0)$. Here we briefly introduce the classic prefix DP and suffix DP solutions to this problem. Using Python's slicing syntax, the base cases of empty sequences are:

$$N(\boldsymbol{x}_t[:0], \boldsymbol{x}_0[:j]) = N(\boldsymbol{x}_t[n:], \boldsymbol{x}_0[j:]) = 1, \quad \forall j \in \{0, ..., m\}. \tag{12}$$

The **prefix DP** iteratively computes $N(\boldsymbol{x}_t[:i], \boldsymbol{x}_0[:j])$ from the solved subproblems $N(\boldsymbol{x}_t[:i], \boldsymbol{x}_0[:j-1])$ and $N(\boldsymbol{x}_t[:i-1], \boldsymbol{x}_0[:j-1])$ to get the final result $N(\boldsymbol{x}_t[:n], \boldsymbol{x}_0[:m])$, and the state transition is:

$$N(\boldsymbol{x}_t[:i], \boldsymbol{x}_0[:j]) = N(\boldsymbol{x}_t[:i], \boldsymbol{x}_0[:j-1]) + \delta(\boldsymbol{x}_t[i-1], \boldsymbol{x}_0[j-1]) \cdot N(\boldsymbol{x}_t[:i-1], \boldsymbol{x}_0[:j-1]). \tag{13}$$

The **suffix DP** iteratively computes $N(\boldsymbol{x}_t[i:], \boldsymbol{x}_0[j:])$ from the solved subproblems $N(\boldsymbol{x}_t[i:], \boldsymbol{x}_0[j+1:])$ and $N(\boldsymbol{x}_t[i+1:], \boldsymbol{x}_0[j+1:])$ to get the final result $N(\boldsymbol{x}_t[0:], \boldsymbol{x}_0[0:])$, and the state transition is:

$$N(\boldsymbol{x}_t[i:], \boldsymbol{x}_0[j:]) = N(\boldsymbol{x}_t[i:], \boldsymbol{x}_0[j+1:]) + \delta(\boldsymbol{x}_t[i], \boldsymbol{x}_0[j]) \cdot N(\boldsymbol{x}_t[i+1:], \boldsymbol{x}_0[j+1:]). \tag{14}$$

The time complexities are $O(mn)$ for both the prefix and suffix DPs. Notably, they could be batched and parallelized along the i-dimension, thus supporting vectorization and parallel execution on GPU CUDA cores, and it only needs to sequentially loop over the j-dimension for m times.

Counting $N(\textbf{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)$. It could be efficiently solved based on the results of prefix and suffix DP:

$$N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0) = \underbrace{\sum_{j=1}^{m} \Big[\delta(\boldsymbol{x}_0[j], v)}_{\text{index addition}} \cdot \underbrace{N(\boldsymbol{x}_t[:i], \boldsymbol{x}_0[:j-1])}_{\text{prefix DP result}} \cdot \underbrace{N(\boldsymbol{x}_t[i:], \boldsymbol{x}_0[j:])}_{\text{suffix DP result}} \Big], \tag{15}$$

due to the form of Eq. 15, results for all (i, v) pairs can be solved in parallel with an elementwise multiplication of the prefix and suffix DP result matrices, followed by an index addition that could be efficiently implemented with a sparse tensor coalescence. A PyTorch implementation of the DP algorithms is in Appendix G.1.

3.5 Simplified Model for Fixed-Length Setting

To facilitate a fair comparison with MDLMs on the widely-adopted fixed-length language modeling benchmarks (Tab. 1), and clearly isolate the superior FLOPs efficiency of DID, we develop a set of optimizations to enhance DID in the fixed-length setting. We show that when $|\boldsymbol{x}_0|$ is a constant, 1) the insertion score becomes time-independent as the time-dependent terms of $(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|}$ in Eq. 7 could be canceled out, which leads to 2) a sequence-level normalization property (details in Appendix D.7):

$$\sum_{i,v} \bar{s}(\boldsymbol{x}_t, t)[i, v] = |\boldsymbol{x}_0| - |\boldsymbol{x}_t|. \tag{16}$$

This benefits the parameterization and training of DID from two aspects. First, the insertion score becomes time-independent, i.e., the network does not require time t as input in the fixed-length setting, thus the parameterization reduces to $\bar{s}_\theta(\boldsymbol{x}_t)$, saves the parameters for time embedding, and enables a cache mechanism similar to (Ou et al., 2025) if the sequence is not changed between steps. Second, the output of the insertion score network could be explicitly normalized with a summation of $|\boldsymbol{x}_0| - |\boldsymbol{x}_t|$ as in Eq. 16. Therefore, the summation term of outputs from the insertion score network in the DISE objective (Eq. 11) turns into a constant, giving rise to a simplified Denoising Insertion Cross Entropy (DICE) objective:

$$\mathcal{L}_\theta^{\text{DICE}}(\boldsymbol{x}_0) = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \left\{ \sum_{i,v} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \Big[-\log \bar{s}_\theta(\boldsymbol{x}_t)[i, v] + C \Big] \right\}, \tag{17}$$

where $C = \log \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)}$ is a θ-free constant. DICE can be interpreted as a weighted cross-entropy loss between the predicted insertion scores and the ground truth of subsequence count ratios, hence the name.

Table 1: Zero-shot language modeling perplexity. Results for diffusion models are perplexity upper bounds.

Size	Method	WikiText	Lambada	Pubmed	AG News	LM1B	Arxiv	PTB
Small	RADD	38.27	51.82	56.99	73.18	72.99	85.95	**108.79**
	DID-S	38.72	49.10	55.02	76.02	74.04	82.41	115.37
	DID-F	**36.91**	**48.00**	**52.89**	**71.48**	**72.04**	**78.38**	111.60
Medium	RADD	28.44	44.10	41.06	48.96	60.32	66.28	**81.05**
	DID-S	29.19	41.94	40.84	52.53	59.88	63.95	91.87
	DID-F	**28.35**	**41.00**	**38.71**	**48.84**	**58.05**	**61.77**	87.09

4 EXPERIMENTS

We evaluate DID fixed-length models in Sec. 4.1, and variable-length models in Sec. 4.2. For more details, results, generation examples, and intermediate generation process, please refer to Appendix E, F, H.

4.1 DID FOR FIXED-LENGTH LANGUAGE MODELING

Settings. Following RADD (Ou et al., 2025), which serves as our baseline method of MDLM, we train DID of both small and medium sizes on the OpenWebText (OWT) dataset (Gokaslan & Cohen, 2019) with the DICE objective (Eq. 17). We adopt the GPT2 tokenizer (Radford et al., 2019), concatenate all sequences, and split them into fixed-length chunks of 1024 tokens, and the training batch size is 512. RADD-small and -medium are reproduced with their open-sourced model checkpoints (trained for 400K steps on OWT).

Zero-shot language modeling perplexity. We evaluate the zero-shot modeling perplexity on seven datasets in Tab. 1. Since DID eliminates the computational FLOPs for <MASK> and hence reduces FLOPs by approximately half compared to RADD for the same training steps, we compare DID under two training configurations: **DID-S** (Steps-aligned), trained for the same steps as RADD (400K) and **DID-F** (FLOPs-aligned), trained for double the steps (800K) to match the total computational budget. We observe that when aligned by training steps (DID-S), our model achieves performance comparable to RADD, despite utilizing only about half the computational FLOPs. When aligned by the total computational budget (DID-F), DID consistently outperforms RADD across the majority of datasets for both small and medium sizes. This demonstrates that the computational savings achieved by eliminating <MASK> tokens are effectively turned into improved modeling performance. We also provide an ablation study in Appendix F.1 for DID trained with DISE objective (Eq. 11), i.e., without the additional optimizations in DICE for fixed-length data introduced in Sec. 3.5. This results in a reduced performance than DICE-trained DID in Tab. 1, yet comparable to RADD.

Generative performance. We report generative perplexity (PPL), unigram entropy (diversity), and inference speed of direct sampling across different numbers of total denoising steps in Tab. 2. Compared with RADD, DID achieves significantly better generation quality (lower PPL) with fewer denoising steps. When more total denoising steps are used, RADD slightly outperforms DID, which is reasonable since RADD is naturally designed for the fixed-length setting. Nonetheless, DID could consistently outperform RADD in the variable-length setting, which is deferred to Tab. 4. Furthermore, DID consistently provides $\sim 1.5\times$ inference speedup. This improvement stems from the fact that the average sequence length during the iterative insertion process is shorter than the fixed-length process of RADD. We also provide nucleus sampling results in Appendix F.3, where DID achieves lower PPL and entropy compared with RADD, demonstrating a stronger annealing effect.

Training speed. [4] We report the training speeds for different model sizes in Tab. 3 to verify the efficiency gains of DID. DID demonstrates substantial speedups, increasing from $1.63\times$ to $1.82\times$ as we scale up the model. Empirically, removing the computations for <MASK> tokens significantly boosts training efficiency.

[4] Large models are not fully trained; only their training speeds are measured.

Table 2: Generative perplexity (PPL, evaluated by GPT2 Large), unigram entropy, inference time (in seconds), speedup, and average generation length for fixed-length models under different total denoising steps.

Method	Steps	16	32	64	128	256	512	1024
RADD	PPL	284.78	155.01	111.56	95.10	87.56	**84.00**	**84.05**
	Entropy	8.35	8.26	8.20	8.15	8.11	8.10	8.09
	Time (s)	0.220	0.317	0.499	0.879	1.644	2.882	4.512
DID	PPL	**158.93**	**110.06**	**97.32**	**91.25**	**86.98**	86.04	85.35
	Entropy	8.15	8.13	8.13	8.12	8.09	8.08	8.09
	Time (s)	**0.169**	**0.246**	**0.353**	**0.573**	**1.047**	**1.826**	**3.006**
	Speedup	1.30×	1.29×	1.41×	1.53×	1.57×	1.58×	1.50×
	Length	1023.29	1024.01	1024.18	1024.07	1023.91	1024.03	1024.10

While the reduction in FLOPs suggests a theoretical 2× speedup, the actual gains in Tab. 2 are more modest. This discrepancy, as discussed in (Zheng et al., 2025), arises because inference is not a purely FLOPs-bound task. For training, the observed gains in Tab. 3 are also slightly lower due to additional overheads such as the DP algorithm for loss implementation (Sec. 3.4), which is a constant overhead independent of the model size, thus the speedup could be scaled up with model size. Besides, a less developed system-level support for variable-length data also constrains the speedups.

Table 3: Average training time (in seconds) per 50 steps (i.e. batches) on OpenWebText.

	Small	Medium	Large
RADD	26.46	53.17	92.90
DID	**16.26**	**31.16**	**51.14**
Speedup	1.63×	1.71×	1.82×

4.2 DID for Variable-Length Language Modeling

Settings. Following ILM (Patel et al., 2025), an insertion-based approach, we train DID-small on the Stories dataset (Eldan & Li, 2023; Mostafazadeh et al., 2016) for 60K steps with batch size 512, utilizing its variable-length sequences (average length 213.43 under the Bert-base-uncased tokenizer (Devlin et al., 2018)) truncated to a maximum length of 1024 and without padding. We also train RADD-small on Stories for the same steps ($> 2\times$ FLOPs of DID), details in Appendix E. Since RADD requires fixed-length inputs, we pad all sequences to a length of 1024, which is in line with the setting for MDLM in ILM experiments (Patel et al., 2025). As a result, RADD generates fixed-length outputs containing `<PAD>` tokens, which we subsequently remove to obtain the final variable-length outputs. ILM is reproduced with its open-sourced checkpoints.

Table 4: Generative PPL, unigram entropy, inference time (in seconds), and average generation length for variable-length models under different denoising steps. *: as outliers significantly affect PPL, only samples with PPL < 300 are counted, †: speedup over RADD.

Method	Steps	64	128	256	512
ILM	PPL*	161.80	137.64	42.29	31.14
	Entropy	5.20	5.65	5.97	6.01
	Time (s)	**0.016**	**0.034**	**0.087**	**0.271**
	Length	63.34	120.77	206.44	234.44
RADD	PPL*	81.92	50.89	34.47	26.78
	Entropy	5.22	5.58	5.79	5.85
	Time (s)	0.246	0.441	0.827	1.461
	Length	110.66	200.73	349.54	353.47
DID	PPL	**22.78**	**21.07**	**21.90**	**23.88**
	Entropy	5.90	5.94	5.94	5.94
	Time (s)	0.090	0.132	0.218	0.388
	Speedup†	2.73×	3.34×	3.79×	3.76×
	Length	182.31	193.77	202.97	204.96

Generative performance. We report the generation quality and speed of direct sampling for variable-length models in Tab. 4. Compared with both baselines, DID maintains a significantly lower generative PPL (evaluated by GPT2 Large) with relatively high diversity and is more stable across different numbers of steps. This is important because strong sensitivity and non-convergence to the manually predefined number of generation steps are undesirable.

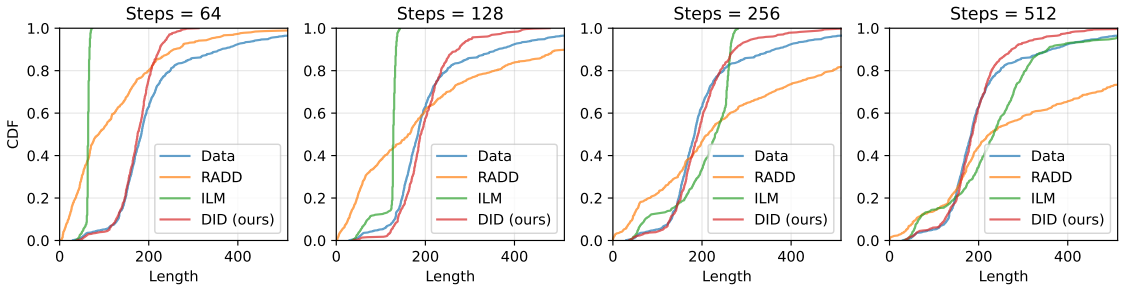

Figure 2: Cumulative distribution functions (CDFs) of generation length under different total denoising steps.

Regarding inference speed, DID achieves a speedup of up to 3.79× compared to RADD, due to the savings of <MASK> and <PAD> tokens for both model calling and distribution sampling. On the other hand, ILM achieves the fastest speed despite its lowest quality. We credit this to its over-simplified generation mechanism, which samples a token at exactly one *designated* position per step. In contrast, DID and RADD sample from all possible token positions to enable parallel decoding. ILM benefits considerably from this simplification, since categorical sampling is a major bottleneck in small models (Zheng et al., 2025). Moreover, ILM is limited to generating text shorter than the total steps (see Tab. 4), which also contributes to its faster sampling.

We also provide nucleus sampling results for variable-length models in Appendix F.3, and ablation studies of different padding lengths for RADD in Appendix F.4, addressing potential concerns that the 1024 padding length for RADD might be too long or unfair. When trained at a length of 512, RADD exhibits observable degradation and remains $\sim 1.59\times$ slower than DID, further confirming the original setting of ILM.

Length modeling. Besides, DID demonstrates superior length modeling capabilities, exhibiting consistency between the generation length distribution and the training data length distribution. This is demonstrated in Fig. 2, where the CDF of the length distribution of DID is closely aligned with the dataset compared to the baselines. DID's average generation length reported in Tab. 4 is also stable and approximating the ground truth distribution.

Table 5: Average training time (in seconds) per 50 steps on Stories.

	Small	Medium	Large
RADD	19.93	37.87	67.75
DID	**10.59**	**14.20**	**21.83**
Speedup	1.88×	2.67×	3.10×

Training speed. [5] We also compare the training speed on the Stories dataset (Tab. 5). The efficiency gains of DID are even more pronounced in the variable-length setting, reaching up to 3.10× speedup for large models. This is because RADD suffers significant overhead from processing <PAD> tokens (since the average length 213.43 is much shorter than the padded length 1024), while DID benefits from the shorter length.

Regarding the language modeling ability for variable-length models, please see Appendix F.2.

5 Conclusion

In this paper, we introduce DID to improve the computational efficiency and generation flexibility of diffusion language models by eliminating the use of <MASK> and <PAD> tokens in our paradigm. Theoretically, we formulate the diffusion processes for deletion and insertion, define an insertion score, and derive the corresponding training objectives and sampling algorithm for DID. We evaluated DID on modeling performance, generation quality, and training/inference speed, demonstrating the superiority of DID over the baselines of MDLM and existing insertion-based LMs in both fixed-length and variable-length settings. A discussion of the limitations and future works of this paper is provided in Appendix A.

[5] Medium and large models are not fully trained; only their training speeds are measured.

REFERENCES

Marianne Arriola, Aaron Gokaslan, Justin T Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. *arXiv preprint arXiv:2503.09573*, 2025.

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.

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.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One billion word benchmark for measuring progress in statistical language modeling, 2014. URL https://arxiv.org/abs/1312.3005.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. A discourse-aware attention model for abstractive summarization of long documents. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 2018. doi: 10.18653/v1/n18-2097. URL http://dx.doi.org/10.18653/v1/n18-2097.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*, 2022.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Ronen Eldan and Yuanzhi Li. Tinystories: How small can language models be and still speak coherent english?, 2023. URL https://arxiv.org/abs/2305.07759.

Daniel T Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *The Journal of chemical physics*, 115(4):1716–1733, 2001.

Aaron Gokaslan and Vanya Cohen. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus, 2019.

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.

Marton Havasi, Brian Karrer, Itai Gat, and Ricky TQ Chen. Edit flows: Flow matching with edit operations. *arXiv preprint arXiv:2506.09018*, 2025.

Jaeyeon Kim, Lee Cheuk-Kit, Carles Domingo-Enrich, Yilun Du, Sham Kakade, Timothy Ngotiaoco, Sitan Chen, and Michael Albergo. Any-order flexible length masked diffusion, 2025. URL https://arxiv.org/abs/2509.01025.

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

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete Diffusion Modeling by Estimating the Ratios of the Data Distribution. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 32819–32848. PMLR, 2024.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993. URL https://www.aclweb.org/anthology/J93-2004.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. A corpus and cloze evaluation for deeper understanding of commonsense stories. In Kevin Knight, Ani Nenkova, and Owen Rambow (eds.), *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 839–849, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1098. URL https://aclanthology.org/N16-1098/.

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

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025.

Manfred Opper and Guido Sanguinetti. Variational inference for markov jump processes. In J. Platt, D. Koller, Y. Singer, and S. Roweis (eds.), *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/file/735b90b4568125ed6c3f678819b6e058-Paper.pdf.

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. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=7yqjVgWWxx.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernandez. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1525–1534, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P16-1144.

Dhruvesh Patel, Aishwarya Sahoo, Avinash Amballa, Tahira Naseem, Tim G. J. Rudner, and Andrew McCallum. Insertion language models: Sequence generation with arbitrary-position insertions, 2025. URL https://arxiv.org/abs/2505.05755.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

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.

Subham Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024.

Subham Sekhar Sahoo, Zhihan Yang, Yash Akhauri, Johnna Liu, Deepansha Singh, Zhoujun Cheng, Zhengzhong Liu, Eric Xing, John Thickstun, and Arash Vahdat. Esoteric language models, 2025. URL https://arxiv.org/abs/2506.01928.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. *Advances in neural information processing systems*, 37:103131–103167, 2024.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023. URL https://arxiv.org/abs/2104.09864.

Guanghan Wang, Yair Schiff, Subham Sahoo, and Volodymyr Kuleshov. Remasking discrete diffusion models with inference-time scaling. *arXiv preprint arXiv:2503.00307*, 2025.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dllm: Training-free acceleration of diffusion llm by enabling kv cache and parallel decoding, 2025a. URL https://arxiv.org/abs/2505.22618.

Zirui Wu, Lin Zheng, Zhihui Xie, Jiacheng Ye, Jiahui Gao, Yansong Feng, Zhenguo Li, Victoria W., Guorui Zhou, and Lingpeng Kong. Dreamon: Diffusion language models for code infilling beyond fixed-size canvas, 2025b. URL https://hkunlp.github.io/blog/2025/dreamon.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 2015.

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. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=CTC7CmirNr.

CONTENTS

A Limitations and Future Works

First, this work presents the core framework of DID and, unlike the more established MDLMs, has not yet integrated many optimizations, such as advanced inference algorithms (Wu et al., 2025a; Wang et al., 2025), or hybrid models combining autoregressive approaches (Arriola et al., 2025; Sahoo et al., 2025). Since these optimizations are not inherently tied to a specific diffusion process, adapting them to DID represents a promising future direction. Second, although we have demonstrated the effectiveness, efficiency, and flexibility of DID, our models were trained at a relatively small scale due to resource constraints. As a result, their performance on larger and more complex tasks remains unexplored, and we leave scaling up DID to future work.

B The Use of Large Language Models (LLMs)

Following ICLR guidelines, we wish to clarify our use of Large Language Models (LLMs) during the preparation of this work.

The research ideas, methodology, experimental design, and analysis presented in this paper were developed entirely by the human authors. LLMs were not involved in the ideation process.

We utilized LLMs as tools for editing and polishing the text, helping to improve the clarity and phrasing in various sections of the main paper and the appendix. The authors have reviewed the manuscript thoroughly and take full responsibility for its content.

C Notation Summary

Table 6: Summary of Key Notations

Notation	Description
$\mathcal{V}$	Vocabulary.
$\mathcal{X} = \bigcup_{d=0}^{\infty} \mathcal{V}^d$	Sequence state space (variable length).
$\boldsymbol{x}_0, \boldsymbol{x}_t, \boldsymbol{y}$	Clean sequence; sequence at time t; another sequence state.
$\lvert\boldsymbol{x}\rvert$	Length of $\boldsymbol{x}$.
<BOS>, <MASK>, <PAD>	Begin-of-sequence (non-deletable), absorbing mask, padding.
$\varnothing$	Null token representing deletion or no insertion (not in $\mathcal{V}$).
Q_t, $\tilde{Q}_t$	Forward and reverse sequence-level CTMC rate matrices.
$p_{t\lvert s}(\cdot\lvert\cdot)$	Forward transition probability from s to t.
$p_t(\cdot)$	Marginal distribution at time t.
$\sigma(t)$, $\bar{\sigma}(t)$	Noise rate and its integral $\int_0^t \sigma(\tau)\, d\tau$.
$s(\boldsymbol{x}_t, t)_{\boldsymbol{y}}$	Concrete score $p_t(\boldsymbol{y})/p_t(\boldsymbol{x}_t)$.
$N(\boldsymbol{x}, \boldsymbol{y})$	The number of occurrences of $\boldsymbol{x}$ as a distinct subsequence of $\boldsymbol{y}$.
$\boldsymbol{y} \succ_1 \boldsymbol{x}$	$\boldsymbol{y}$ is obtained by inserting exactly one token into $\boldsymbol{x}$.
$v(\boldsymbol{x}, \boldsymbol{y})$	The unique inserted token when $\boldsymbol{y} \succ_1 \boldsymbol{x}$.
$\mathrm{Ins}(\boldsymbol{x}, i, v)$	The result of inserting token $v \in \mathcal{V}$ *after* position i of $\boldsymbol{x}$.
$I(\boldsymbol{x}, \boldsymbol{y})$	Valid insertion indices s.t. $\mathrm{Ins}(\boldsymbol{x}, i, v(\boldsymbol{x}, \boldsymbol{y})) = \boldsymbol{y}$.
$\bar{s}(\boldsymbol{x}_t, t)[i, v]$	Insertion score for insertion operation (i, v) at time t.
$\bar{s}(\boldsymbol{x}_t)[i, v]$	Time-independent insertion score (fixed-length setting).
$K(a)$	Convex function $a(\log a - 1)$.

D Detailed Proofs and Derivations

D.1 Derivation of Denoising Score Entropy Loss (Eq.3)

The training objective for discrete diffusion models is derived by minimizing a variational upper bound on the negative log-likelihood (NLL) of the data, $-\log p_0^\theta(\boldsymbol{x}_0)$.

Let $\mathbb{P}_{\boldsymbol{x}_0}$ denote the path measure (the probability distribution over entire trajectories) of the true posterior reverse process conditioned on the data $\boldsymbol{x}_0$. Let $\mathbb{P}^\theta$ denote the path measure of the learned reverse process parameterized by θ. By the data processing inequality, we have:

$$-\log p_0^\theta(\boldsymbol{x}_0) = D_{\mathrm{KL}}\left(\delta_{\boldsymbol{x}_0} \| p_0^\theta\right) \leq D_{\mathrm{KL}}(\mathbb{P}_{\boldsymbol{x}_0} \| \mathbb{P}^\theta). \tag{18}$$

We define the training objective as this variational upper bound: $\mathcal{L}(\theta) = D_{\mathrm{KL}}(\mathbb{P}_{\boldsymbol{x}_0} \| \mathbb{P}^\theta)$, assuming both processes share the same prior distribution at $t = 1$.

Both processes are Continuous-Time Markov Chains (CTMCs). Let $\widetilde{Q}_t^0(\boldsymbol{x}_t, \boldsymbol{y})$ denote the true conditional reverse transition rate (for $\mathbb{P}_{\boldsymbol{x}_0}$) and $\widetilde{Q}_t^\theta(\boldsymbol{x}_t, \boldsymbol{y})$ denote the parameterized reverse transition rate (for $\mathbb{P}^\theta$).

The KL divergence between path measures can be decomposed using the chain rule. If we consider a discrete-time approximation with infinitesimal step Δt, the total KL divergence is the sum of the expected KL divergences at each step. We analyze the KL divergence between the infinitesimal transition probabilities $p^0(\boldsymbol{y}|\boldsymbol{x}_t) = \delta(\boldsymbol{x}_t, \boldsymbol{y}) + \widetilde{Q}_t^0(\boldsymbol{x}_t, \boldsymbol{y})\Delta t + o(\Delta t)$ and $p^\theta(\boldsymbol{y}|\boldsymbol{x}_t) = \delta(\boldsymbol{x}_t, \boldsymbol{y}) + \widetilde{Q}_t^\theta(\boldsymbol{x}_t, \boldsymbol{y})\Delta t + o(\Delta t)$. The instantaneous KL divergence is (Opper & Sanguinetti, 2007):

$$D_{\mathrm{KL}}(p^0(\cdot|\boldsymbol{x}_t) \| p^\theta(\cdot|\boldsymbol{x}_t)) = \Delta t \sum_{\boldsymbol{y} \neq \boldsymbol{x}_t} \left(\widetilde{Q}_t^0(\boldsymbol{x}_t, \boldsymbol{y}) \log \frac{\widetilde{Q}_t^0(\boldsymbol{x}_t, \boldsymbol{y})}{\widetilde{Q}_t^\theta(\boldsymbol{x}_t, \boldsymbol{y})} + \widetilde{Q}_t^\theta(\boldsymbol{x}_t, \boldsymbol{y}) - \widetilde{Q}_t^0(\boldsymbol{x}_t, \boldsymbol{y}) \right) + o(\Delta t). \tag{19}$$

Summing these contributions and taking the continuous limit ($\Delta t \to 0$) yields the integral form for the path KL divergence:

$$\mathcal{L}(\theta) = \int_0^1 \mathop{\mathbb{E}}_{\boldsymbol{x}_t \sim p_{t|0}} \left[\sum_{\boldsymbol{y} \neq \boldsymbol{x}_t} \left(\widetilde{Q}_t^0(\boldsymbol{x}_t, \boldsymbol{y}) \log \frac{\widetilde{Q}_t^0(\boldsymbol{x}_t, \boldsymbol{y})}{\widetilde{Q}_t^\theta(\boldsymbol{x}_t, \boldsymbol{y})} + \widetilde{Q}_t^\theta(\boldsymbol{x}_t, \boldsymbol{y}) - \widetilde{Q}_t^0(\boldsymbol{x}_t, \boldsymbol{y}) \right) \right] dt. \tag{20}$$

We now substitute the specific definitions of these transition rates. The true conditional reverse rate $\widetilde{Q}_t^0$ is related to the forward rate $Q_t(\boldsymbol{y}, \boldsymbol{x}_t)$ by

$$\widetilde{Q}_t^0(\boldsymbol{x}_t, \boldsymbol{y}) = Q_t(\boldsymbol{y}, \boldsymbol{x}_t) \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)}. \tag{21}$$

The parameterized reverse rate $\widetilde{Q}_t^\theta$ is defined using the score network $s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}}$:

$$\widetilde{Q}_t^\theta(\boldsymbol{x}_t, \boldsymbol{y}) = Q_t(\boldsymbol{y}, \boldsymbol{x}_t) s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}}. \tag{22}$$

We substitute these rates (Eq.21 and Eq.22) into the expression inside the summation in Eq.20. The expression becomes:

$$\left(Q_t(\boldsymbol{y}, \boldsymbol{x}_t) \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \right) \log \frac{Q_t(\boldsymbol{y}, \boldsymbol{x}_t) \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)}}{Q_t(\boldsymbol{y}, \boldsymbol{x}_t) s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}}} + Q_t(\boldsymbol{y}, \boldsymbol{x}_t) s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} - Q_t(\boldsymbol{y}, \boldsymbol{x}_t) \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)}. \tag{23}$$

We simplify by canceling $Q_t(\boldsymbol{y}, \boldsymbol{x}_t)$ inside the logarithm and factoring it out from the entire expression:

$$= Q_t(\boldsymbol{y}, \boldsymbol{x}_t) \left[\frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \log \frac{\frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)}}{s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}}} + s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} - \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \right]. \tag{24}$$

We expand the logarithm ($\log(A/B) = \log A - \log B$) and rearrange the terms:

$$= Q_t(\boldsymbol{y}, \boldsymbol{x}_t) \left[\frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \left(\log \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} - \log s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} \right) + s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} - \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \right] \tag{25}$$

$$= Q_t(\boldsymbol{y}, \boldsymbol{x}_t) \left[s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} - \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \log s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} \ + \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \left(\log \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} - 1 \right) \right]. \tag{26}$$

Letting $K(a) = a(\log a - 1)$. Recognizing that the time integral $\int_0^1 dt$ combined with the expectation over $\boldsymbol{x}_t \sim p_{t|0}$ is equivalent to the expectation over uniform time $t \sim U(0, 1)$ and the corresponding conditional $\boldsymbol{x}_t$, the objective $\mathcal{L}(\theta)$ is exactly the DSE loss (Eq.3):

$$\mathcal{L}_\theta^{\text{DSE}}(\boldsymbol{x}_0) = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \sum_{\boldsymbol{y} \neq \boldsymbol{x}_t} Q_t(\boldsymbol{y}, \boldsymbol{x}_t) \left[s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} - \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \log s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} + K \left(\frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \right) \right]. \tag{27}$$

D.2 Derivation of Sequence-Level Transition Probability (Eq.5)

We derive the sequence-level transition probability $p_{t|s}(\boldsymbol{x}_t|\boldsymbol{x}_s)$ based on the definition of the DID forward process as an independent token-level deletion process. We begin by analyzing the dynamics of a single token.

The token-level process (Eq.4) is a Continuous-Time Markov Chain (CTMC) on the state space $\{v, \varnothing\}$, where $v \in \mathcal{V}$ denotes the presence of a token and $\varnothing$ denotes the deleted state. The transition rate matrix Q_t^{tok} at time t, indexed by $(v, \varnothing)$, is defined by the deletion rate $\sigma(t)$:

$$Q_t^{\text{tok}} = \begin{pmatrix} -\sigma(t) & \sigma(t) \\ 0 & 0 \end{pmatrix}. \tag{28}$$

Let $P_v(\tau)$ be the probability that the token is in state v at time $\tau \in [s, t]$, given it started in state v at time s ($P_v(s) = 1$). The evolution of this probability follows the Kolmogorov forward equation:

$$\frac{dP_v(\tau)}{d\tau} = P_v(\tau) Q_\tau^{\text{tok}}(v, v) + P_\varnothing(\tau) Q_\tau^{\text{tok}}(\varnothing, v) = -\sigma(\tau) P_v(\tau). \tag{29}$$

Solving this first-order ordinary differential equation by integrating from s to t:

$$\int_s^t \frac{dP_v(\tau)}{P_v(\tau)} = \int_s^t -\sigma(\tau) d\tau \implies \ln(P_v(t)) - \ln(P_v(s)) = -(\bar{\sigma}(t) - \bar{\sigma}(s)). \tag{30}$$

Thus, the probability that a single token survives during the interval $[s, t]$ is $P_v(t) = e^{-(\bar{\sigma}(t) - \bar{\sigma}(s))}$. Conversely, the probability of deletion is $1 - e^{-(\bar{\sigma}(t) - \bar{\sigma}(s))}$.

We now consider the Sequence-level transition from $\boldsymbol{x}_s$ to $\boldsymbol{x}_t$. Let $\Delta\bar{\sigma} = \bar{\sigma}(t) - \bar{\sigma}(s)$. A transition occurs if the tokens forming $\boldsymbol{x}_t$ survive and the remaining $|\boldsymbol{x}_s| - |\boldsymbol{x}_t|$ tokens are deleted. Since token deletions are independent, the probability of a specific path (a specific occurrence of $\boldsymbol{x}_t$ in $\boldsymbol{x}_s$) is $(e^{-\Delta\bar{\sigma}})^{|\boldsymbol{x}_t|} \times (1 - e^{-\Delta\bar{\sigma}})^{|\boldsymbol{x}_s| - |\boldsymbol{x}_t|}$.

The total transition probability $p_{t|s}(\boldsymbol{x}_t|\boldsymbol{x}_s)$ is the sum over all distinct paths, counted by the subsequence count $N(\boldsymbol{x}_t, \boldsymbol{x}_s)$. Therefore, we obtain:

$$p_{t|s}(\boldsymbol{x}_t|\boldsymbol{x}_s) = N(\boldsymbol{x}_t, \boldsymbol{x}_s)(1 - e^{-(\bar{\sigma}(t) - \bar{\sigma}(s))})^{|\boldsymbol{x}_s| - |\boldsymbol{x}_t|} e^{-(\bar{\sigma}(t) - \bar{\sigma}(s))|\boldsymbol{x}_t|}. \tag{31}$$

D.3 Derivation of the Transition Rate (Eq.6)

We derive the transition rate $Q_t(\boldsymbol{y}, \boldsymbol{x}_t)$ from a sequence $\boldsymbol{y}$ to a sequence $\boldsymbol{x}_t$, where $\boldsymbol{x}_t$ is obtained from $\boldsymbol{y}$ by deleting a single token (denoted as $\boldsymbol{y} \succ_1 \boldsymbol{x}_t$). This implies $|\boldsymbol{y}| = |\boldsymbol{x}_t| + 1$.

$$
\begin{aligned}
Q_t(\boldsymbol{y}, \boldsymbol{x}_t) &\triangleq \lim_{\Delta t \to 0} \frac{p_{t+\Delta t|t}(\boldsymbol{x}_t|\boldsymbol{y})}{\Delta t} && (32)\\
&= \lim_{\Delta t \to 0} \frac{(1 - e^{-(\bar{\sigma}(t+\Delta t) - \bar{\sigma}(t))})^{|\boldsymbol{y}|-|\boldsymbol{x}_t|} e^{-(\bar{\sigma}(t+\Delta t) - \bar{\sigma}(t))|\boldsymbol{x}_t|} N(\boldsymbol{x}_t, \boldsymbol{y})}{\Delta t} && (33)\\
&= \lim_{\Delta t \to 0} \frac{(1 - e^{-\sigma(t)\Delta t + o(\Delta t)})^1 \cdot e^{-(\sigma(t)\Delta t + o(\Delta t))|\boldsymbol{x}_t|} \cdot N(\boldsymbol{x}_t, \boldsymbol{y})}{\Delta t} && (34)\\
&= \lim_{\Delta t \to 0} \frac{(\sigma(t)\Delta t + o(\Delta t)) \cdot (1 - |\boldsymbol{x}_t|\sigma(t)\Delta t + o(\Delta t)) \cdot N(\boldsymbol{x}_t, \boldsymbol{y})}{\Delta t} && (35)\\
&= \lim_{\Delta t \to 0} \frac{\sigma(t)\Delta t + o(\Delta t)}{\Delta t} \cdot N(\boldsymbol{x}_t, \boldsymbol{y}) && (36)\\
&= \sigma(t) N(\boldsymbol{x}_t, \boldsymbol{y}). && (37)
\end{aligned}
$$

D.4 Derivation of the Relationship between Concrete Score and Insertion Score (Eq. 8)

We aim to prove the identity stated in Eq. 8. This identity concerns the backward process where transitions occur such that $\boldsymbol{y} \succ_1 \boldsymbol{x}_t$. Note that this condition implies $|\boldsymbol{y}| = |\boldsymbol{x}_t| + 1$.

First, we derive the explicit form of the concrete score $s(\boldsymbol{x}_t, t)_{\boldsymbol{y}} = p_t(\boldsymbol{y})/p_t(\boldsymbol{x}_t)$ by expanding the marginal distributions using the forward transition probability (Eq. 5):

$$
\begin{aligned}
s(\boldsymbol{x}_t, t)_{\boldsymbol{y}} = \frac{p_t(\boldsymbol{y})}{p_t(\boldsymbol{x}_t)} &= \frac{\mathbb{E}_{\boldsymbol{x}_0}[p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)]}{\mathbb{E}_{\boldsymbol{x}_0}[p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)]}\\
&= \frac{\mathbb{E}_{\boldsymbol{x}_0}\left[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|-|\boldsymbol{y}|} e^{-\bar{\sigma}(t)|\boldsymbol{y}|} N(\boldsymbol{y}, \boldsymbol{x}_0)\right]}{\mathbb{E}_{\boldsymbol{x}_0}\left[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|-|\boldsymbol{x}_t|} e^{-\bar{\sigma}(t)|\boldsymbol{x}_t|} N(\boldsymbol{x}_t, \boldsymbol{x}_0)\right]}\\
&= \frac{e^{-\bar{\sigma}(t)|\boldsymbol{y}|}(1 - e^{-\bar{\sigma}(t)})^{-|\boldsymbol{y}|}}{e^{-\bar{\sigma}(t)|\boldsymbol{x}_t|}(1 - e^{-\bar{\sigma}(t)})^{-|\boldsymbol{x}_t|}} \frac{\mathbb{E}_{\boldsymbol{x}_0}\left[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{y}, \boldsymbol{x}_0)\right]}{\mathbb{E}_{\boldsymbol{x}_0}\left[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{x}_t, \boldsymbol{x}_0)\right]}\\
&\overset{|\boldsymbol{y}|=|\boldsymbol{x}_t|+1}{=} \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{\mathbb{E}_{\boldsymbol{x}_0}\left[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{y}, \boldsymbol{x}_0)\right]}{\mathbb{E}_{\boldsymbol{x}_0}\left[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{x}_t, \boldsymbol{x}_0)\right]}. && (38)
\end{aligned}
$$

Next, we examine the right-hand side (RHS) of Eq. 8 and substitute the definition of the insertion score $\bar{s}$ (Eq. 7):

$$
\begin{aligned}
\text{RHS} &= \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}(\boldsymbol{x}_t, t)[i, v(\boldsymbol{x}_t, \boldsymbol{y})] && (39)\\
&= \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \left(\frac{\mathbb{E}_{\boldsymbol{x}_0}[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\text{Ins}(\boldsymbol{x}_t, i, v(\boldsymbol{x}_t, \boldsymbol{y})), \boldsymbol{x}_0)]}{\mathbb{E}_{\boldsymbol{x}_0}[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{x}_t, \boldsymbol{x}_0)]} \right). && (40)
\end{aligned}
$$

By definition of the index set $I(\boldsymbol{x}_t, \boldsymbol{y})$, for any $i \in I(\boldsymbol{x}_t, \boldsymbol{y})$, we have $\text{Ins}(\boldsymbol{x}_t, i, v(\boldsymbol{x}_t, \boldsymbol{y})) = \boldsymbol{y}$. Therefore:

$$
\text{RHS} = \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \left(\frac{\mathbb{E}_{\boldsymbol{x}_0}[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{y}, \boldsymbol{x}_0)]}{\mathbb{E}_{\boldsymbol{x}_0}[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{x}_t, \boldsymbol{x}_0)]} \right). \quad (41)
$$

The term inside the summation is independent of the index i. We factor it out and utilize the property that $\sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} 1 = |I(\boldsymbol{x}_t, \boldsymbol{y})| = N(\boldsymbol{x}_t, \boldsymbol{y})$:

$$\text{RHS} = \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \left(\frac{\mathbb{E}_{\boldsymbol{x}_0}[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{y}, \boldsymbol{x}_0)]}{\mathbb{E}_{\boldsymbol{x}_0}[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{x}_t, \boldsymbol{x}_0)]} \right) N(\boldsymbol{x}_t, \boldsymbol{y}) \tag{42}$$

$$= \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{\mathbb{E}_{\boldsymbol{x}_0}\left[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{y}, \boldsymbol{x}_0)\right]}{\mathbb{E}_{\boldsymbol{x}_0}\left[(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|} N(\boldsymbol{x}_t, \boldsymbol{x}_0)\right]}. \tag{43}$$

This matches the derived concrete score in Eq. 38, completing the proof.

D.5 Derivation of the Sampling Probability (Eq.10)

We aim to derive the probability of executing a specific insertion operation—inserting token v after position i—within an infinitesimal time interval $[t - \Delta t, t]$ during the backward process.

The backward process is a CTMC characterized by the parameterized reverse transition rate matrix $\tilde{Q}_t^\theta$. The rate of transition from state $\boldsymbol{x}_t$ to a different state $\boldsymbol{y}$ is defined as:

$$\tilde{Q}_t^\theta(\boldsymbol{x}_t, \boldsymbol{y}) = Q_t(\boldsymbol{y}, \boldsymbol{x}_t) s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}}. \tag{44}$$

In the DID framework, backward transitions occur only when $\boldsymbol{y} \succ_1 \boldsymbol{x}_t$. We substitute the forward transition rate (Eq.6), $Q_t(\boldsymbol{y}, \boldsymbol{x}_t) = \sigma(t) N(\boldsymbol{x}_t, \boldsymbol{y})$. We also substitute the parameterized concrete score s_θ, which is derived by parameterizing the relationship between the concrete score and the insertion score (Eq.8):

$$s_\theta(\boldsymbol{x}_t, t)_{\boldsymbol{y}} = \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{j \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_\theta(\boldsymbol{x}_t, t)[j, v(\boldsymbol{x}_t, \boldsymbol{y})]. \tag{45}$$

Substituting these expressions into the definition of $\tilde{Q}_t^\theta(\boldsymbol{x}_t, \boldsymbol{y})$:

$$\tilde{Q}_t^\theta(\boldsymbol{x}_t, \boldsymbol{y}) = (\sigma(t) N(\boldsymbol{x}_t, \boldsymbol{y})) \cdot \left(\frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{j \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_\theta(\boldsymbol{x}_t, t)[j, v(\boldsymbol{x}_t, \boldsymbol{y})] \right) \tag{46}$$

$$= \sigma(t) \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{j \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_\theta(\boldsymbol{x}_t, t)[j, v(\boldsymbol{x}_t, \boldsymbol{y})]. \tag{47}$$

This result demonstrates that the total transition rate from $\boldsymbol{x}_t$ to $\boldsymbol{y}$ is decomposed into a summation of individual components. Each term in the summation, $\sigma(t) \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \bar{s}_\theta(\boldsymbol{x}_t, t)[j, v(\boldsymbol{x}_t, \boldsymbol{y})]$, corresponds to the instantaneous rate of the specific insertion operation $(j, v(\boldsymbol{x}_t, \boldsymbol{y}))$ that transforms $\boldsymbol{x}_t$ into $\boldsymbol{y}$.

By the definition of a CTMC, the probability of a specific operation (i, v) occurring within the infinitesimal interval Δt is given by its corresponding rate multiplied by Δt. For $v \neq \varnothing$, we identify this probability directly from the decomposition above:

$$p_{t-\Delta t|t}^\theta((i, v)|\boldsymbol{x}_t) = \left(\sigma(t) \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] \right) \Delta t + o(\Delta t) \tag{48}$$

$$= \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] \Delta t + o(\Delta t). \tag{49}$$

This confirms the first case of Eq.10. The probability of no insertion occurring at position i (i.e., $v = \varnothing$) is determined by the normalization constraint, ensuring the sum of probabilities for all possible events at that position equals 1.

To confirm the equivalence between this action-based sampling and the required state-based transition, we verify that the action probabilities correctly recover the state transition probability $p^{\theta}_{t-\Delta t|t}(\boldsymbol{y}|\boldsymbol{x}_t)$. A transition to $\boldsymbol{y}$ occurs if any of the actions indexed by $I(\boldsymbol{x}_t, \boldsymbol{y})$ occurs. In the infinitesimal limit $\Delta t \to 0$, these actions are mutually exclusive events:

$$p^{\theta}_{t-\Delta t|t}(\boldsymbol{y}|\boldsymbol{x}_t) = \sum_{j \in I(\boldsymbol{x}_t, \boldsymbol{y})} p^{\theta}_{t-\Delta t|t}((j, v(\boldsymbol{x}_t, \boldsymbol{y}))|\boldsymbol{x}_t) + o(\Delta t) \tag{50}$$

$$= \left(\sigma(t) \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{j \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_{\theta}(\boldsymbol{x}_t, t)[j, v(\boldsymbol{x}_t, \boldsymbol{y})] \right) \Delta t + o(\Delta t). \tag{51}$$

By Eq. 47, the term in the parenthesis is exactly $\tilde{Q}^{\theta}_t(\boldsymbol{x}_t, \boldsymbol{y})$. Thus, $p^{\theta}_{t-\Delta t|t}(\boldsymbol{y}|\boldsymbol{x}_t) = \tilde{Q}^{\theta}_t(\boldsymbol{x}_t, \boldsymbol{y})\Delta t + o(\Delta t)$, confirming that the action-based sampling correctly implements the dynamics of the backward CTMC.

D.6 Derivation of the DISE Objective (Eq.11)

We derive the Denoising Insertion Score Entropy (DISE) objective (Eq.11) starting from the general DSE objective (Eq.3), demonstrating that DISE is a variational upper bound on DSE, $\mathcal{L}^{\text{DISE}}_{\theta}(\boldsymbol{x}_0) \geq \mathcal{L}^{\text{DSE}}_{\theta}(\boldsymbol{x}_0)$.

We begin with the DSE objective:

$$\mathcal{L}^{\text{DSE}}_{\theta}(\boldsymbol{x}_0) = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \sum_{\boldsymbol{y} \neq \boldsymbol{x}_t} Q_t(\boldsymbol{y}, \boldsymbol{x}_t) \left[s_{\theta}(\boldsymbol{x}_t, t)_{\boldsymbol{y}} - \frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \log s_{\theta}(\boldsymbol{x}_t, t)_{\boldsymbol{y}} + K\left(\frac{p_{t|0}(\boldsymbol{y}|\boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} \right) \right], \tag{52}$$

where $K(a) = a(\log a - 1)$. In the deletion–insertion process, the transition rate $Q_t(\boldsymbol{y}, \boldsymbol{x}_t)$ is non-zero only when $\boldsymbol{y} \succ_1 \boldsymbol{x}_t$.

We now substitute the specific definitions for the DID process. The transition rate is $Q_t(\boldsymbol{y}, \boldsymbol{x}_t) = \sigma(t) N(\boldsymbol{x}_t, \boldsymbol{y})$. The conditional probability ratio is:

$$\frac{p_{t|0}(\boldsymbol{y} \mid \boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t \mid \boldsymbol{x}_0)} = \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)}. \tag{53}$$

The parameterized concrete score (from parameterized Eq.8) is:

$$s_{\theta}(\boldsymbol{x}_t, t)_{\boldsymbol{y}} = \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_{\theta}(\boldsymbol{x}_t, t)[i, v(\boldsymbol{x}_t, \boldsymbol{y})]. \tag{54}$$

We examine the expression inside the bracket of the DSE objective by substituting these definitions. Let $v = v(\boldsymbol{x}_t, \boldsymbol{y})$ for brevity in the following block:

$$\begin{aligned}
& s_{\theta}(\boldsymbol{x}_t, t)_{\boldsymbol{y}} - \frac{p_{t|0}(\boldsymbol{y} \mid \boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t \mid \boldsymbol{x}_0)} \log s_{\theta}(\boldsymbol{x}_t, t)_{\boldsymbol{y}} + K\left(\frac{p_{t|0}(\boldsymbol{y} \mid \boldsymbol{x}_0)}{p_{t|0}(\boldsymbol{x}_t \mid \boldsymbol{x}_0)} \right) \\
&= \frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \left(\sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_{\theta}(\boldsymbol{x}_t, t)[i, v] \right) \\
&\quad - \left(\frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \right) \log \left(\frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \left(\sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_{\theta}(\boldsymbol{x}_t, t)[i, v] \right) \right) \\
&\quad + K\left(\frac{e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \right).
\end{aligned} \tag{55}$$

We expand the $K(\cdot)$ term using $K(a) = a(\log a - 1)$. By expanding the logarithms ($\log(AB) = \log A + \log B$), we observe that the terms involving the time factor $\log(\frac{e^{-\bar{\sigma}(t)}}{1-e^{-\bar{\sigma}(t)}})$ cancel out exactly.

The expression inside the bracket simplifies significantly by factoring out the time prefactor:

$$
\begin{aligned}
&= \frac{e^{-\bar{\sigma}(t)}}{1-e^{-\bar{\sigma}(t)}} \Bigg[\frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \left(\sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] \right) \\
&\quad - \frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \log \left(\frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \left(\sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] \right) \right) \\
&\quad + \frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \left(\log \frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} - 1 \right) \Bigg].
\end{aligned} \tag{56}
$$

Substituting this simplified bracket back into the main objective equation and multiplying by $Q_t(\boldsymbol{y}, \boldsymbol{x}_t) = \sigma(t) N(\boldsymbol{x}_t, \boldsymbol{y})$. The DSE objective can be decomposed into three terms (T1, T2, T3):

$$
\mathcal{L}_\theta^{\text{DSE}}(\boldsymbol{x}_0) = \text{T1} + \text{T2} + \text{T3}. \tag{57}
$$

We define these terms based on the components derived above:

$$
\text{T1} = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1-e^{-\bar{\sigma}(t)}} \sum_{\boldsymbol{y} \succ_1 \boldsymbol{x}_t} N(\boldsymbol{x}_t, \boldsymbol{y}) \left[\frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] \right]. \tag{58}
$$

$$
\text{T2} = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1-e^{-\bar{\sigma}(t)}} \sum_{\boldsymbol{y} \succ_1 \boldsymbol{x}_t} N(\boldsymbol{x}_t, \boldsymbol{y}) \left[-\frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \log \left(\frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] \right) \right]. \tag{59}
$$

$$
\text{T3} = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1-e^{-\bar{\sigma}(t)}} \sum_{\boldsymbol{y} \succ_1 \boldsymbol{x}_t} N(\boldsymbol{x}_t, \boldsymbol{y}) K \left(\frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \right). \tag{60}
$$

We now apply Jensen's inequality to T2. The term inside the logarithm is an average of insertion scores. Because the logarithm function is concave, $\log(\frac{1}{N} \sum a_i) \geq \frac{1}{N} \sum \log a_i$. Since the logarithm is negated, this leads to an upper bound on T2:

$$
-\log \left(\frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] \right) \leq -\frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \log \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v]. \tag{61}
$$

We define T2_{Bound} as the upper bound for T2:

$$
\text{T2} \leq \text{T2}_{\text{Bound}} = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1-e^{-\bar{\sigma}(t)}} \sum_{\boldsymbol{y} \succ_1 \boldsymbol{x}_t} N(\boldsymbol{x}_t, \boldsymbol{y}) \frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \left[-\frac{1}{N(\boldsymbol{x}_t, \boldsymbol{y})} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \log \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] \right]. \tag{62}
$$

We define the DISE objective as the upper bound obtained by replacing T2 with T2_{Bound}, ensuring $\mathcal{L}_\theta^{\text{DISE}}(\boldsymbol{x}_0) \geq \mathcal{L}_\theta^{\text{DSE}}(\boldsymbol{x}_0)$:

$$
\mathcal{L}_\theta^{\text{DISE}}(\boldsymbol{x}_0) = \text{T1} + \text{T2}_{\text{Bound}} + \text{T3}. \tag{63}
$$

To simplify the nested summations and transform the objective from state-level ($\boldsymbol{y}$) to operation-level ((i, v)), we rely on the following identity.

Lemma 1 (Summation change of variables)**.** Let $v(\boldsymbol{x}_t, \boldsymbol{y})$ denote the unique inserted token and $I(\boldsymbol{x}_t, \boldsymbol{y})$ the set of valid insertion positions that yield $\boldsymbol{y}$ from $\boldsymbol{x}_t$. For any function G,

$$\sum_{\boldsymbol{y} \succ_1 \boldsymbol{x}_t} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} G\big(i,\, v(\boldsymbol{x}_t, \boldsymbol{y}),\, \boldsymbol{y}\big) \;=\; \sum_{i,v} G\big(i,\, v,\, \mathrm{Ins}(\boldsymbol{x}_t, i, v)\big). \tag{64}$$

Proof. Define the index sets

$$\mathcal{A} = \{(\boldsymbol{y}, i) :\; \boldsymbol{y} \succ_1 \boldsymbol{x}_t,\; i \in I(\boldsymbol{x}_t, \boldsymbol{y})\}, \qquad \mathcal{B} = \{(i, v) :\; i \in \{0, \ldots, |\boldsymbol{x}_t|\}, v \in \mathcal{V}\}. \tag{65}$$

The map $g : \mathcal{B} \to \mathcal{A}$ defined by $g(i, v) = (\mathrm{Ins}(\boldsymbol{x}_t, i, v),\, i)$ is a bijection, with its inverse $f : \mathcal{A} \to \mathcal{B}$ defined by $f(\boldsymbol{y}, i) = (i,\, v(\boldsymbol{x}_t, \boldsymbol{y}))$. Changing variables over this bijection gives the claimed identity. □

We apply Lemma 1 to simplify T1, $\mathrm{T2}_{\mathrm{Bound}}$, and T3.

For T1, the $N(\boldsymbol{x}_t, \boldsymbol{y})$ terms cancel before the summation.

$$\mathrm{T1} = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{\boldsymbol{y} \succ_1 \boldsymbol{x}_t} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v(\boldsymbol{x}_t, \boldsymbol{y})] \tag{66}$$

$$= \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{i,v} \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v]. \quad \text{(Using Lemma 1)} \tag{67}$$

For $\mathrm{T2}_{\mathrm{Bound}}$, cancellation of $N(\boldsymbol{x}_t, \boldsymbol{y})$ yields:

$$\mathrm{T2}_{\mathrm{Bound}} = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{\boldsymbol{y} \succ_1 \boldsymbol{x}_t} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} -\frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \log \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v(\boldsymbol{x}_t, \boldsymbol{y})]. \tag{68}$$

Using Lemma 1, noting that $\boldsymbol{y} = \mathrm{Ins}(\boldsymbol{x}_t, i, v)$:

$$\mathrm{T2}_{\mathrm{Bound}} = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{i,v} -\frac{N(\mathrm{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \log \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v]. \tag{69}$$

For T3, we first utilize the fact that $N(\boldsymbol{x}_t, \boldsymbol{y}) = \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} 1$.

$$\mathrm{T3} = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{\boldsymbol{y} \succ_1 \boldsymbol{x}_t} \sum_{i \in I(\boldsymbol{x}_t, \boldsymbol{y})} K\left(\frac{N(\boldsymbol{y}, \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)}\right). \tag{70}$$

Applying Lemma 1 to T3:

$$\mathrm{T3} = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{i,v} K\left(\frac{N(\mathrm{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)}\right). \tag{71}$$

We combine T1, $\mathrm{T2}_{\mathrm{Bound}}$, and T3 to obtain the final DISE objective. Let C denote the constant term T3's summand.

$$\mathcal{L}_\theta^{\mathrm{DISE}}(\boldsymbol{x}_0) = \mathrm{T1} + \mathrm{T2}_{\mathrm{Bound}} + \mathrm{T3} \tag{72}$$

$$= \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \left\{ \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{i,v} \left[\bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] - \frac{N(\mathrm{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \log \bar{s}_\theta(\boldsymbol{x}_t, t)[i, v] + C \right] \right\}. \tag{73}$$

D.7 Derivation of the Sequence-Level Normalization Property (Eq. 16)

In the fixed-length setting (Sec. 3.5), we assume $|\boldsymbol{x}_0| = K$ (a constant) for all data samples. Under this assumption, the insertion score becomes time-independent. We aim to prove the normalization property (Eq. 16), restated here for the fixed-length context:

$$\sum_{i,v} \bar{s}(\boldsymbol{x}_t)[i, v] = K - |\boldsymbol{x}_t|. \tag{74}$$

The proof relies on a fundamental combinatorial identity regarding subsequence counts.

Lemma 2 (Subsequence Count Identity)**.** For any sequences $\boldsymbol{x}_t$ and $\boldsymbol{x}_0$ such that $\boldsymbol{x}_t$ is a subsequence of $\boldsymbol{x}_0$, the following identity holds:

$$\sum_{i,v} N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0) = N(\boldsymbol{x}_t, \boldsymbol{x}_0)(|\boldsymbol{x}_0| - |\boldsymbol{x}_t|). \tag{75}$$

Proof of Lemma 2. We prove the statement $\sum_{i,v} N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0) = N(\boldsymbol{x}_t, \boldsymbol{x}_0)(|\boldsymbol{x}_0| - |\boldsymbol{x}_t|)$ via a bijective proof, by constructing two sets of equal cardinality. Let $S(\boldsymbol{x}, \boldsymbol{z})$ denote the set of index tuples corresponding to all occurrences of a subsequence $\boldsymbol{x}$ in $\boldsymbol{z}$, such that $N(\boldsymbol{x}, \boldsymbol{z}) = |S(\boldsymbol{x}, \boldsymbol{z})|$.

First, consider the set A, defined as the set of pairs (I, j), where I is the index tuple of an occurrence of $\boldsymbol{x}_t$ in $\boldsymbol{x}_0$, and j is an index in $\boldsymbol{x}_0$ that is not part of that occurrence:

$$A = \{(I, j) : I \in S(\boldsymbol{x}_t, \boldsymbol{x}_0), j \in \{1, \dots, |\boldsymbol{x}_0|\} \setminus I\}. \tag{76}$$

The cardinality of A is $|A| = N(\boldsymbol{x}_t, \boldsymbol{x}_0)(|\boldsymbol{x}_0| - |\boldsymbol{x}_t|)$.

Second, consider the set B, defined as the set of pairs $((i, v), J)$, where (i, v) is an insertion operation on $\boldsymbol{x}_t$, and J is the index tuple of an occurrence of the resulting sequence, $\text{Ins}(\boldsymbol{x}_t, i, v)$, in $\boldsymbol{x}_0$:

$$B = \{((i, v), J) : J \in S(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)\}. \tag{77}$$

The cardinality of B is $|B| = \sum_{i,v} N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)$.

We now establish a bijection between A and B. For any element $(I, j) \in A$, we define a mapping to an element in B as follows: let the inserted token be $v = \boldsymbol{x}_0[j]$, and let the insertion position relative to $\boldsymbol{x}_t$ be $i = |\{k \in I : k < j\}|$. The new index tuple is $J = I \cup \{j\}$, sorted. The subsequence $\boldsymbol{x}_0[J]$ is precisely $\text{Ins}(\boldsymbol{x}_t, i, v)$ by construction. This defines a unique mapping $f : A \to B$.

Conversely, for any element $((i, v), J) \in B$, we can define an inverse mapping. The index of the inserted token in $\boldsymbol{x}_0$ is the $(i + 1)$-th element of the sorted tuple J; let this be j. Removing this index yields the tuple $I = J \setminus \{j\}$, which corresponds to an occurrence of $\boldsymbol{x}_t$. This defines a unique mapping $g : B \to A$.

Since a one-to-one correspondence exists between the sets, their cardinalities must be equal. Therefore, $|A| = |B|$, which proves the lemma. □

Proof of Eq. 16. Under the fixed-length assumption ($|\boldsymbol{x}_0| = K$), the definition of the insertion score (Eq. 7) simplifies because the time-dependent terms $(1 - e^{-\bar{\sigma}(t)})^{|\boldsymbol{x}_0|}$ are constant $(1 - e^{-\bar{\sigma}(t)})^K$ and cancel out, leading to the time-independent score:

$$\bar{s}(\boldsymbol{x}_t)[i, v] = \frac{\mathbb{E}_{\boldsymbol{x}_0}[N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)]}{\mathbb{E}_{\boldsymbol{x}_0}[N(\boldsymbol{x}_t, \boldsymbol{x}_0)]}. \tag{78}$$

We sum this score over all possible insertion operations (i, v):

$$\sum_{i,v} \bar{s}(\boldsymbol{x}_t)[i, v] = \sum_{i,v} \frac{\mathbb{E}_{\boldsymbol{x}_0}[N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)]}{\mathbb{E}_{\boldsymbol{x}_0}[N(\boldsymbol{x}_t, \boldsymbol{x}_0)]} \tag{79}$$

$$= \frac{1}{\mathbb{E}_{\boldsymbol{x}_0}[N(\boldsymbol{x}_t, \boldsymbol{x}_0)]} \mathbb{E}_{\boldsymbol{x}_0} \left[\sum_{i,v} N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0) \right]. \tag{80}$$

We apply the combinatorial identity (Lemma 2) to the summation inside the expectation, substituting $|\boldsymbol{x}_0| = K$:

$$\sum_{i,v} N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0) = N(\boldsymbol{x}_t, \boldsymbol{x}_0)(K - |\boldsymbol{x}_t|). \tag{81}$$

Substituting this back:

$$\sum_{i,v} \bar{s}(\boldsymbol{x}_t)[i, v] = \frac{1}{\mathbb{E}_{\boldsymbol{x}_0}[N(\boldsymbol{x}_t, \boldsymbol{x}_0)]} \mathbb{E}_{\boldsymbol{x}_0} \left[N(\boldsymbol{x}_t, \boldsymbol{x}_0)(K - |\boldsymbol{x}_t|) \right]. \tag{82}$$

Since $(K - |\boldsymbol{x}_t|)$ is constant with respect to the expectation over $\boldsymbol{x}_0$:

$$\sum_{i,v} \bar{s}(\boldsymbol{x}_t)[i, v] = (K - |\boldsymbol{x}_t|) \frac{\mathbb{E}_{\boldsymbol{x}_0}[N(\boldsymbol{x}_t, \boldsymbol{x}_0)]}{\mathbb{E}_{\boldsymbol{x}_0}[N(\boldsymbol{x}_t, \boldsymbol{x}_0)]} = K - |\boldsymbol{x}_t|. \tag{83}$$

This confirms the normalization property stated in Eq. 16. □

D.8 Derivation of the DICE Objective for Fixed-Length Data (Eq. 17)

In the fixed-length setting (Section 3.5), we assume $|\boldsymbol{x}_0| = K$ (constant). This assumption leads to time-independent insertion scores $\bar{s}_\theta(\boldsymbol{x}_t)[i, v]$ (as shown in Appendix D.7) and allows for the exact simplification of the DISE objective into the Denoising Insertion Cross-Entropy (DICE) objective.

We start from the DISE objective (Eq.11):

$$\mathcal{L}_\theta^{\text{DISE}}(\boldsymbol{x}_0) = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{i,v} \Bigg[\bar{s}_\theta(\boldsymbol{x}_t)[i, v] - \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \log \bar{s}_\theta(\boldsymbol{x}_t)[i, v] + K \left(\frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \right) \Bigg]. \tag{84}$$

We rearrange the expression inside the square brackets using the definition $K(a) = a(\log a - 1)$.

$$\begin{aligned} &\bar{s}_\theta[i, v] - \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \log \bar{s}_\theta[i, v] + \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \left(\log \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} - 1 \right) \\ &= \left(\bar{s}_\theta[i, v] - \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \right) \\ &\quad + \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \left(\log \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} - \log \bar{s}_\theta[i, v] \right). \end{aligned} \tag{85}$$

We substitute this rearrangement back into the DISE objective:

$$\begin{aligned} \mathcal{L}_\theta^{\text{DISE}}(\boldsymbol{x}_0) = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{i,v} \Bigg[&\left(\bar{s}_\theta[i, v] - \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \right) \\ &+ \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \left(\log \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} - \log \bar{s}_\theta[i, v] \right) \Bigg]. \end{aligned} \tag{86}$$

We now utilize the crucial normalization properties derived from the fixed-length assumption ($|\boldsymbol{x}_0| = K$). From Lemma 2 (Appendix D.7), the true subsequence count ratios satisfy:

$$\sum_{i,v} \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} = K - |\boldsymbol{x}_t|. \tag{87}$$

As discussed in Section 3.5, we design the network architecture such that the parameterized scores $\bar{s}_\theta$ exactly satisfy the same normalization constraint:

$$\sum_{i,v} \bar{s}_\theta(\boldsymbol{x}_t)[i, v] = K - |\boldsymbol{x}_t|. \tag{88}$$

Because both the true ratios and the parameterized scores sum to the same value, the summation of the first group of terms in Eq. 85 vanishes:

$$\sum_{i,v} \left(\bar{s}_\theta(\boldsymbol{x}_t)[i, v] - \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \right) = (K - |\boldsymbol{x}_t|) - (K - |\boldsymbol{x}_t|) = 0. \tag{89}$$

Therefore, the DISE objective simplifies exactly. We define this simplified form as the DICE objective, which is exactly equal to the DISE loss under the fixed-length setting ($\mathcal{L}_\theta^{\text{DICE}}(\boldsymbol{x}_0) = \mathcal{L}_\theta^{\text{DISE}}(\boldsymbol{x}_0)$):

$$\mathcal{L}_\theta^{\text{DICE}}(\boldsymbol{x}_0) = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \sum_{i,v} \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \left(\log \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} - \log \bar{s}_\theta(\boldsymbol{x}_t)[i, v] \right). \tag{90}$$

Rearranging the terms inside the summation yields the final DICE objective:

$$\mathcal{L}_\theta^{\text{DICE}}(\boldsymbol{x}_0) = \mathop{\mathbb{E}}_{t, \boldsymbol{x}_t} \left\{ \sum_{i,v} \frac{\sigma(t) e^{-\bar{\sigma}(t)}}{1 - e^{-\bar{\sigma}(t)}} \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)} \left[- \log \bar{s}_\theta(\boldsymbol{x}_t)[i, v] + C \right] \right\}, \tag{91}$$

where $C = \log \frac{N(\text{Ins}(\boldsymbol{x}_t, i, v), \boldsymbol{x}_0)}{N(\boldsymbol{x}_t, \boldsymbol{x}_0)}$ is a θ-free constant.

E Experimental Details

E.1 Fixed-Length Training Details

Model architecture. Following RADD, our models use an encoder-only transformer architecture with a dropout rate 0.02, rotary position embedding (Su et al., 2023), and untied word embeddings between input and output. Other model architecture hyperparameters are summarized in Tab. 7. The FFN dimension is 4× hidden dimension.

Size	Layers	Hidden Dimension	Attention Heads
Small	12	768	12
Medium	24	1024	16
Large	36	1280	20

Table 7: Model architecture hyperparameters for difference model sizes of RADD and DID.

We use FlashAttention (Dao et al., 2022) to support attention computation for packed variable-length sequences, while RADD uses PyTorch's standard scaled dot product attention for its fixed-length batched inputs, which is also based on FlashAttention.

Dataset and pre-processing. For a fair comparison, we train DID on OpenWebText dataset (Gokaslan & Cohen, 2019) with the same steps (400K steps) or compute budget (800K steps) as RADD (Ou et al., 2025), under the log-linear noise schedule $\bar{\sigma}(t) = -\log(1-t)$, we train DID with a batch size 512, and a context length 1024. We tokenize OpenWebText dataset with the GPT2 (Radford et al., 2019) tokenizer, the vocabulary size is 50257. We follow the data pre-processing adopted in RADD, concatenating all sequences in the training dataset, then splitting it into chunks of fixed length 1024. We add a <BOS> special token to the first position of each chunk to model the insertion behavior before the first normal token by modeling the insertion after the <BOS> special token. We use the same token as <EOS> for <BOS>. We evaluate the zero-shot language modeling perplexity on WikiText, Lambada, Scientific Papers (Arxiv and Pubmed, abstract parts), AG News, LM1B, and PTB datasets (Merity et al., 2017; Paperno et al., 2016; Cohan et al., 2018; Zhang et al., 2015; Chelba et al., 2014; Marcus et al., 1993) in Tab. 1.

Optimization. For a fair comparison, we did not do a hyperparameter search, our optimization configuration strictly follows RADD, we use AdamW (Loshchilov & Hutter, 2019) optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon =$ 1e-8, a weight decay rate of 0.03, a constant learning rate 3e-4 that linearly warmed up from 0 over the first 2500 steps, a gradient norm clipping value 1, an exponential moving average (EMA) with a decay rate 0.9999, and float16 is enabled for mixed-precision training. To avoid CUDA OOM errors during model training, we set the gradient accumulation steps to 1, 2, 2 for DID small, medium, and large; and 2, 4, 4 for RADD small, medium, and large.

Hardware. All models are trained on 8 NVIDIA H100 80GB GPUs.

E.2 Variable-Length Training Details

Model architecture. Following ILM (Patel et al., 2025), we train a small model with 85M non-embedding parameters. Different from the fixed-length model, the variable-length model is not time-independent, i.e. we need to input the time information into the network. Therefore, we employ an adaptive layernorm for each transformer block as in the practice of DiT (Peebles & Xie, 2023) (as well as the time-dependent masked diffusion models like SEDD (Lou et al., 2024) and MDLM (Sahoo et al., 2024)), the condition embedding dimension is 128, which brings about extra parameters. To keep the total parameter amount of a transformer block unchanged, we reduce the FFN dimension to 3.5$\times$ hidden dimension. Following ILM, we use a dropout rate of 0.1, rotary positional embedding, and untied word embeddings between input and output. FlashAttention is also employed for the efficient computation of variable-length data.

Dataset and pre-processing. Following ILM, we train variable-length models on the Stories dataset. We tokenize it with the Bert-Base-Uncased tokenizer, whose vocabulary size is 30522. Each sentence in the datasets is an individual training datapoint; hence, the training sequences are of variable lengths. The Stories dataset is truncated with a maximum context length of 1024. The batch size is 512. Data is also noised with a log-linear noise schedule in the diffusion forward process.

Optimization. Following the experiment settings of ILM, we use an AdamW optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon =$ 1e-8, a weight decay of 0.01 on all parameters (including biases and normalization layers), a constant learning rate 1e-4 that linearly warmed up from 0 over the first 1000 steps, a gradient norm clipping value 1, an exponential moving average (EMA) with a decay rate 0.9999, and bfloat16 is enabled for mixed-precision training. Stories dataset is trained for 60k steps. To avoid CUDA OOM errors during model training, we set the gradient accumulation steps to 1, 1, 1 for DID small, medium, and large; and 2, 4, 4 for RADD small, medium, and large.

Hardware. The models are trained on 8 NVIDIA H100 GPUs.

E.3 Sampling Details

Timestep discretization. We use a standard uniform timestep discretizaition grid, i.e. $t(i) = i/N$, where N is the total number of denoising steps in generation.

The number of samples. We evaluate the averaged generative perplexity (evaluated by GPT2 Large), unigram entropy, generation length, and inference time over 1024 samples with a batch size of 32.

Data precision. According to the precision issue of Gumbel-argmax sampling with float32 discussed in (Ou et al., 2025; Zheng et al., 2025), we use float64 precision for all generation tasks to ensure accurate categorical samplings.

Hardware. The sampling experiments are conducted on a single NVIDIA H100 80GB GPU.

E.4 Evaluation Metrics

Zero-shot language modeling perplexity is used to evaluate how well a model is trained, which uses the model to be evaluated to calculate the exponential of the average negative log-likelihood per token on datasets unseen by the model (i.e. zero-shot):

$$\text{PPL} = \exp\Big(\mathbb{E}_{x\sim p_{\text{data}}(x)}\Big[-\frac{\log p_\theta(x)}{L(x)}\Big]\Big), \tag{92}$$

where the likelihood can be exact (e.g. in auto-regressive models), or bounded (e.g. in diffusion models), and $L(x)$ is the length of x.

Generative perplexity is a widely used metric to evaluate the quality of model-generated text, whose definition is also as in Eq.92, the differences are p_{data} is generated by the model to be evaluated, and θ is another off-the-shelf model to calculate the likelihood.

Unigram entropy is a metric to evaluate the diversity of model-generated text, which is based on the token occurrence frequency in a sentence. For a sentence x with length L, the entropy is:

$$H = -\sum_{i=1}^{L}\frac{N(x_i,x)}{L(x)}\log_2\frac{N(x_i,x)}{L(x)}, \tag{93}$$

where $N(x_i, x)$ is the occurrence time of token x_i in sentence x, and $L(x)$ is the length of x.

F More Experimental Results

F.1 Ablation Study of Sequence-Level Normalization for Fixed-Length Models

We provide the ablation study of fixed-length models trained without the sequence-level normalization introduced in Sec. 3.5, i.e. trained with the DISE objective in Eq. 11 rather than the DICE objective in Eq. 17. We train FLOPs-aligned models of the small size, i.e. trained for 800K steps for DID models, and 400K steps for RADD. As shown in Tab. 8, the ablation version (DID-F w/o SeqNorm) exhibits an inferior performance compared to DID-F, demonstrating that utilizing the DICE objective for training could achieve an enhanced performance in the fixed-length setting. On the other hand, the ablation version (DID-F w/o SeqNorm) remains comparable to RADD, demonstrating the reasonability of the DISE objective.

F.2 Language Modeling Performance for Variable-Length Models

We analyze the language modeling performance for variable-length models, as ILM does not have a likelihood-bounded training objective, only the language modeling performances of RADD and DID could be evaluated.

Table 8: Ablation study of sequence-level normalization for fixed-length DID, the zero-shot language modeling perplexity on seven datasets are reported. Results for these diffusion models are perplexity upper bounds.

Size	Method	WikiText	Lambada	Pubmed	AG News	LM1B	Arxiv	PTB
Small	RADD	38.27	51.82	56.99	73.18	72.99	85.95	**108.79**
	DID-F w/o SeqNorm	38.55	50.17	53.76	73.25	72.69	81.95	118.63
	DID-F	**36.91**	**48.00**	**52.89**	**71.48**	**72.04**	**78.38**	111.60

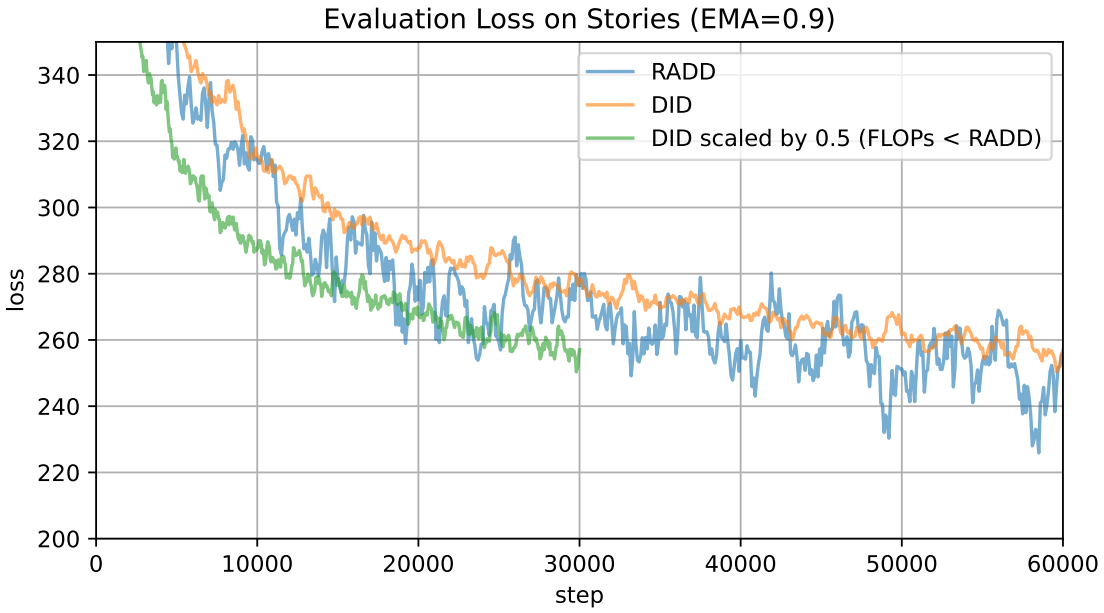

Figure 3: Evaluation loss curve on Stories dataset for RADD and DID in the variable-length setting.

As Stories is a specialized dataset, i.e. not so general like OpenWebText, we only show the evaluation loss curve on its own validation dataset in Fig. 3, where the curves for DID exhibit more stability than RADD's. Besides, we also report the curve scaled by 0.5, whose FLOPs are less than RADD (at the same x-coordinate) as the <PAD> and <MASK> used for RADD occupy more than 1/2 of the computational FLOPs in variable-length setting, this curve remains lower than the RADD curve, demonstrating the superiority of language modeling of DID over RADD in the variable-length setting.

F.3 Nucleus Sampling Results

We provide the evaluation of generation quality with nucleus sampling, a.k.a. top-p sampling, with $p = 0.9$ in Tab. 9 for fixed-length models in Tab. 10 discussed in Sec. 4.1 and variable-length models discussed in Sec. 4.2.

Table 9: Nucleus sampling results for fixed-length models of RADD and DID trained on OpenWebText. Generative perplexity (PPL, evaluated by GPT2 Large), unigram entropy, and average generation length (for DID) under different denoising steps are reported.

Method	Steps	16	32	64	128	256	512	1024
RADD	PPL	95.55	55.21	40.38	34.22	31.79	30.31	30.51
	Entropy	7.99	7.89	7.82	7.77	7.70	7.70	7.69
DID	PPL	**54.40**	**36.80**	**31.73**	**29.61**	**28.23**	**27.29**	**27.05**
	Entropy	7.71	7.69	7.69	7.64	7.60	7.60	7.58
	Length	1021.88	1023.66	1024.02	1024.12	1023.95	1024.06	1024.12

As shown in Tab. 9, with nucleus sampling, DID generations exhibit both lower generative perplexity and lower diversity (measured by unigram entropy), demonstrating the annealing effect for DID is stronger than RADD.

Table 10: Nucleus sampling results for variable-length models of ILM, RADD and DID trained on Stories. Generative perplexity (PPL, evaluated by GPT2 Large), unigram entropy, and average generation length (for DID) under different denoising steps are reported. *: as outliers significantly affect PPL, only samples with PPL < 300 are counted.

Method	Steps	64	128	256	512
ILM	PPL*	174.29	27.04	14.50	15.31
	Entropy	5.02	5.38	5.44	5.45
	Length	62.68	110.05	120.75	122.99
RADD	PPL*	50.63	28.62	22.21	16.11
	Entropy	4.81	5.20	5.39	5.52
	Length	96.14	188.64	228.51	265.53
DID	PPL	**12.33**	**12.74**	**12.46**	**12.83**
	Entropy	5.69	5.72	5.73	5.69
	Length	161.64	171.91	179.62	174.39

As shown in Tab. 10, unlike the fixed-length models in Tab. 9, nucleus sampling results of DID consistently offer lower generative perplexity and higher diversity compared to ILM and RADD, further demonstrating the superiority of DID in the variable-length setting. Besides, the annealing effects could also be observed at the sentence-level, the generation results by nucleus sampling is shorter than those by direct sampling reported in Tab. 4.

F.4 Comparisons with RADD of Different Padding Lengths

As described in Sec. 4.2, the padding length of RADD in the variable-length setting is a hyperparameter to be pre-defined, which is a complexity for MDM in the variable-length setting, as well as a source of its inefficiency of `<PAD>` computation. Here we provide another setting of padding length, a shorter one of 512, to train RADD for the same steps (60K) on Stories dataset, and evaluate its generation quality and speed, the cumulative distribution functions (CDFs) of generation length for different models under different total denoising steps are also shown in Fig. 4, alike the experiments in Sec. 4.2.

Table 11: Ablation study of padding length for RADD in the variable-length setting. Models are trained on Stories for 60K steps. Generative perplexity (PPL, evaluated by GPT2 Large), unigram entropy, inference time (in seconds), and average generation length under different denoising steps are reported. *: as outliers significantly affect PPL, only samples with PPL < 300 are counted.

Method	Steps	64	128	256	512
RADD-512	PPL*	80.19	57.32	40.27	35.45
	Entropy	4.89	5.21	5.56	5.63
	Time (s)	0.123	0.221	0.383	0.608
	Length	91.83	138.70	191.43	207.93
RADD-1024	PPL*	81.92	50.89	34.47	26.78
	Entropy	5.22	5.58	5.79	5.85
	Time (s)	0.246	0.441	0.827	1.461
	Length	110.66	200.73	349.54	353.47
DID	PPL	**22.78**	**21.07**	**21.90**	**23.88**
	Entropy	5.90	5.94	5.94	5.94
	Time (s)	**0.090**	**0.132**	**0.218**	**0.388**
	Length	182.31	193.77	202.97	204.96

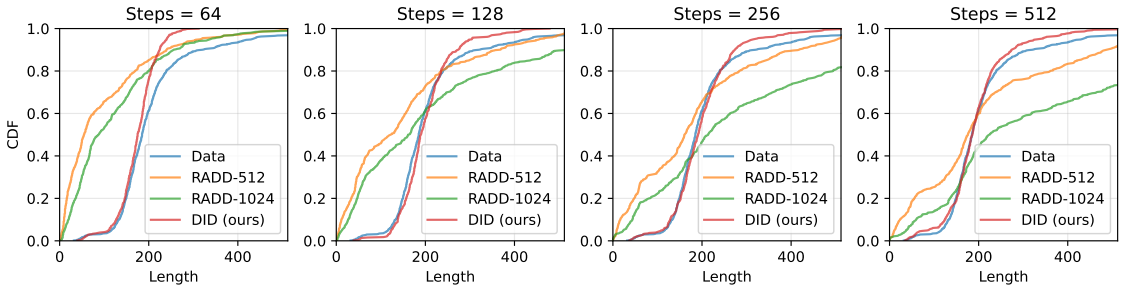

Figure 4: Cumulative distribution functions (CDFs) of generation length under different total denoising steps.

As shown in Tab. 11, the generation quality of RADD with padding length 512 exhibits observable degradation, i.e. higher generative perplexity and lower diversity (measured by unigram entropy), compared to RADD with padding length 1024, demonstrating the reasonability of setting the padding length as 1024, which is also ILM's original training configuration, even though it is highly inefficient as the average length of Stories training dataset is only 213.43, much shorter than 1024, leading to more computational cost for the <PAD> tokens. On the other hand, RADD with padding length 512 achieves a speedup $\approx 2\times$ RADD with padding length 1024, yet still $\approx 1.59\times$ slower than DID on average.

Regarding length modeling, as shown in Fig. 4, although the average generation length of RADD-512 in Tab. 11 is closer to the ground truth (213.43) than RADD-1024, the CDFs in Fig. 4 still show a strong deviation to the ground truth while DID achieves much closer CDFs, which indicates the superiority of DID over MDMs for the variable-length generation task.

G Implementation Details

G.1 PyTorch Implementations of the Dynamic Programming Algorithms

```python
def get_N_ratios(batch, remain_indices, seqlens, token_dim):
    # batched data alignment
    prefix_padded_xt = torch.zeros_like(batch, device=batch.device) - 1
    prefix_data_mask = seqlens[..., None] > torch.arange(batch.shape[1], device=batch.device)[None, ...]
    prefix_padded_xt[prefix_data_mask] = batch[remain_indices]
    prefix_si_eq_tj = batch.unsqueeze(-1) == prefix_padded_xt.unsqueeze(-2)

    suffix_padded_xt = torch.zeros_like(batch, device=batch.device) - 1
    suffix_data_mask = seqlens[..., None] > torch.arange(batch.shape[1] - 1, -1, -1, device=batch.device)[None, ...]
    suffix_padded_xt[suffix_data_mask] = batch[remain_indices]
    suffix_si_eq_tj = batch.unsqueeze(-1) == suffix_padded_xt.unsqueeze(-2)

    B, S = batch.shape

    # N(x_t, x_0) prefix dp
    prefix_dp = torch.zeros(B, S+1, S+1, dtype=torch.double, device=batch.device)
    prefix_dp[:, :, 0] = 1
    for i in range(1, S+1):
        prefix_dp[:, i, 1:] = torch.addcmul(prefix_dp[:, i-1, 1:], prefix_si_eq_tj[:, i-1], prefix_dp[:, i-1, :-1])

    # N(x_t, x_0) suffix dp
    suffix_dp = torch.zeros(B, S+1, S+1, dtype=torch.double, device=batch.device)
    suffix_dp[:, :, -1] = 1
    for i in range(S-1, -1, -1):
        suffix_dp[:, i, :-1] = torch.addcmul(suffix_dp[:, i+1, :-1], suffix_si_eq_tj[:, i], suffix_dp[:, i+1, 1:])

    # N(y, x_0) / N(x_t, x_0) prefix-suffix dp
    V = token_dim
    N_ratios = []
    for b in range(B):
        N = prefix_dp[b, -1, seqlens[b]]
        pr = prefix_dp[b, :-1, 1:seqlens[b] + 1]
        su = suffix_dp[b, 1:, S - seqlens[b] + 1:]
        pr_su = (pr / N) * su

        S, T = pr_su.shape
        rows = batch[b].unsqueeze(1).expand(S, T).reshape(-1)
        cols = torch.arange(T).to(batch.device).unsqueeze(0).expand(S, T).reshape(-1)
        values = pr_su.reshape(-1)

        mask = values.abs() >= 1e-6  # (S*T,) bool mask
        rows = rows[mask]
        cols = cols[mask]
        values = values[mask]
```

```
        indices = torch.stack([rows, cols], dim=0)
        N_ratio = torch.sparse_coo_tensor(indices, values, size=(V, T))
        N_ratios.append(N_ratio)

    packed_N_ratios = torch.cat(N_ratios, 1)
    return packed_N_ratios.t().coalesce() # (\sum_b |x_t|_b, V)
```

Listing 1: PyTorch source code for N-ratios computation via prefix DP, suffix DP, which can be parallelized; and prefix-suffix dynamic programming, which can be efficiently implemented with an elementwise matrix multiplication and a sparse tensor coalescence, as described in Sec. 3.4.

H GENERATION EXAMPLES

Here we provide generation examples of the fixed-length model trained on OpenWebText in Appendix H.1 and variable-length model trained on Stories in Appendix H.2. Besides, we also demonstrate the intermediate generation process in Appendix H.3. All samples are generated by the direct sampling algorithm under the float point precision of fp64 for accurate categorical sampling.

H.1 SAMPLES GENERATED BY THE FIXED-LENGTH MODEL TRAINED ON OPENWEBTEXT

========== Sample 1 ==========

Moon Environment, NEONSEA, CANANA INTERACTIVES,SEVERTY Union ENTRY,PORTS IN DEMO LEAC,,SQUANTIFOCUS, CAZARY, and WACKUP. COvenATION BY BAR WAS PURPOSE TO TEAM DEFENSE, DEFENSE IS THE CHART OF OUTER AND NEGATIVE AND IRONSIDE.

*Selected for Hexagonal Player counting)(Best players from THREE EXPLicates.) of Medals PT50 showed signs of PT in US Switch Ball, SH DROPIONS DOWN HUMAN RIGHTS, CT PORTS were only available for Queensland Olympics USA PAT NACHINO FUTURE Videos reserved. The PACI did mean that, in every four teams from these groups UNMERCHANT WEIGHT CARRYING THE LED ARTS OF BROADCASTING SO NECESSARY DOES, ORBURNVE OFF FOVPORT.ProtarpYSCHIETICS COMICS.COM WILL END CONSENTSLY.

CLICK HERE TO SEE RIGDINS OF AUSTRALIAN BUTTE SON BASING<|endoftext|>by Jay R. Tyardots in California Scientist, January 3, http://www.chronicle.com/news/science-technology/124428.2938drozier55.htm

Engineers working on creating a quantum computing field say they've made balls of quantum information that is able to vibrate like graphene by adjusting its signals like a natural quantum bus. The result is a powerful 250-watt device that could be used for computing purposes within a decade after showing the circuitry could be tuned over time.

The research is in IEEE's Physical Review Letters, Designing the World's First Atom of Silicon Nanotubes.

Better than the current solid-state transistor of opposable states, silicon nanotubes that slip into a hurdle metal sandwich (DA50N2) create FCU.

The non-energetic bits could be on their way into next-generation "intense" processors and even be used in quantum computing.

The researchers said

"The circuit design delivered a new transistor of exponentially lower power," said Enrique Cameron, a nitrogen-dot tech at Rice University in Houston, Texas.

The transistors are integrated with what the researchers call quantum control an internal signal source designed to tell the circuit of the transistor it wants the mechanism to act on. If a magnetic or electrostatic oscillation is expected or unwanted, controlling the quantum control system. The internal signal also controls how much current can be drawn.

Researchers have showed quantum computing in a 256-nd transistor with a count of DSNO, but more research needed increase the size and lithography of the chip even higher.

For the first study for such a transistor done in the same environment, researchers used a unique voltage coupler device. The voltage coupler holds the value of a sine wave and information about the direction of voltage and current.

To output a voltage the computer acts on the sum of the voltage and current which is what people might get if divided a computer transfer rate of 16 Gbit/s (3.24 GB per 100 processors) into half.

According to the group's preliminary result their signals-based feeding strategy, which has also found a way to allow a smartphone to create its own elemental energy, could be widely used at lower power levels. If meant then anything involving new electronics is possible as well.

"The speed in which this sort of mechanism lets off electrons through the direction could have really interesting legs," said Dan Schaberer, a professor in the physics department at Princeton who led the design but who didn't publish a copy of the paper.

The use of a dielectric drive system oscillating the same spirals the actuators provides more mechanical control.

"This is analogous to acceleration using laser in phased scanning microscopy," Cameron said.

Naturally it may be difficult to get like that, but the MIT team said it was using some rough theoretical calculations in the paper and details of the initial version have been tweaked within the theory it is possible.

Guy Lee, a different team at the Technology University Berlin, who's the scientist who developed the device, said "for what most people would call a rapid development challenge team's concept is utterly unique."

He added the team is interested in using future flash-memory chips, for example, as a sufficient unit of mass for quantum logic converters.

"Once quantum bits are developed as nanocomposites and lithography problems solved, then I will not be surprised to see all the fantastic uses come up," Cameron said.<|endoftext|>Here are some Bitcoin events and meetups we were hoping you found important information about and why.

Lindsay Reslove is a research analyst with Coin Private Wealth Research. She is a student in the MSU's departments of business administration and computer science. She and John King, a business in Flint, Mich

========== Sample 2 ==========

barbecues, unstudded haircuts with a passing connection to classical musicians and singers, African stars Bakawisi Sim and Kutimbla Grime. In Case of The Grace of Contessa, the speaker describes Chichamaster Rupert Brecht's interpretation of the Chop Talk, up close to Hoop Orchestra and the Brooklyn Hall Chop bust. The Volk Attendant is dramatologist Philip Hill's take on exploring Bartoz's interpretation of The Young God (Directed by Geraeus Castoré); Penelope Vernon's superb books on Mijunset's Frativity, the Howeñal movement and other forms of Esperanto music are only one of many created in the United States. Her favorites include the e. Additions Games (1811), The Anomal Valley Merry Stories (1815), and the Nieos family Magazine (1856).

The Continental Brand, the dramatist Herbert's Own Dame Vis, introduces English sleuthmuths Ian Woodward and Charles Feching, and doesnt yourself to the first meeting of witty, energetic and well-traveled Spioclasts from San Diego. San Diego the Altun, inspired by a series of articles by Santiago Contamas and José de Septé, the American Innerts, reflecting the peak of the Esperanzaism American Renaissance, tells from the air of Sclepre's Esperanto sophisticate Avarula Grothomo, a desperate person suffering from fibresesticular use (as in the processing of rhubarb). This city may have gone to war or appeared beneath man's feet. Domaniic unic is not the only thing involving Esperanto; the Atl-mas have described a plague. But apparently that never existed.<|endoftext|>Sangefeng: The Yoruba businessman whose family and children lived in a small sarandi (boat) before the Typhoon Haiyan destroyed the last homes there is thought to have been safe in the Philippines for at least next few weeks. Philippines Home Minister, Peter Bola, told on Monday that authorities had discovered a connection between Mr Shinse and the latest Islamic anti-American manifesto during a vitriolic sermon he gave last Sunday at a funeral.

Judging from the "manifesto", Mr Shinse fits the double-standard of the dodgy wealthy Chinese lawyer, where his son had an affair.

His son said he is seen as a "gynoid, disabled in body, deracinated". He said he is mentally unstable. "But that does not mean I'm happy."

He was approached this week by a Haiyan victim. "I broke it getting in the plane, but I said I thought it might be good to find it," he said. This man said that the skull bones in his basement had been removed and awaiting identification.

Throughout last week and a half with displaced people his nephew has approached outside his office to identify themselves. He recently helped organise seminars of the Islamic branch in the Philippines.

"I keep everything on tapes," he said. "Every morning I have to stuff it out the window in front of the TV screen, I can carry it anymore."<|endoftext|>Three hundred revolutionary followers, were took part in Guatemala's Bernardo (Barefoot) alongside 50 Venezuelans, 20 Ecuadoran asylum seekers, to celebrate the revolution, giving a talk-outout to the US Presidential candidate Sanders for expressing "cynicism" about a Socialist state.

"This shows that democracy is not so bad in the world," Patrick Dubo, one of the Trotskyists from the Unido Brigda Party said during the rallies.Some of the crowd continued to protest for Che Guevara and Fidel Castro, as they claimed that they supported "separatism".

"In the internet came a Facebook group which is affirmed to be an alternative Socialist state which fully affirms the concept of "Apartheid" one unnamed person told the press.

Dubo said that they were representatives of anti-socialist Christians which promote Marxism and fight immorality through government policies. He added that socialism has never really disappeared in South America because its ideas are with more with "fleat-blading".On the other hand self-styled "centrist businessman" Ric Williams fully supported the courage despite this by arguing that expressed the spirit of patriotism.

Reporting Daniel Jenks, Elias Osauga, Thompson Salazar and Guy Aveculs

...<|endoftext|>This week, Dow Chemical Co. CEO George Lopulos finally gave the company's 3,000 employees fresh concessions that break codes and specified overhaul of operations. By Jan. 25, they were signed a tentative contract that would represent more than a third of some 5,000 workers.But while the original tentative agreement was set for the end of this month,

========== Sample 3 ==========

"It came from someone who felt the urgency of it and refreshed us."

The idea of an airport being a catalyst to switch venues appeared at the time. The airport has been a home for a decade, with facilities real and perceived to be as nice as they come. Boston has a few Green Card 250, excellent, venues. It is not the site of 60 such ad golf-related contracts according to a report from the AP. Taunton was affected by that, as was the clutch conversion on his Jan. 10, 2014 road trip that made national TV headlines.

He slid perfectly from outside the Patriots' six-yard line and 25 yards out before flipping it over Willy Chopade Jr.'s end zone. This kept him on the field with Patriotsmate Moss.

"We were hoping we were still going to be playing because of Moss," Haley said. "We just came to grips."

Boston has shown a growing appreciation of Shifkar's vision, and now he has the bang of a stadium it will live for. After departure of Kow, its most active sponsors, the Patriots purchased HSBC. The successful discussion of other offers and candidates included Shad Khan, a partner who does global retail on the West End of Oxford.

"We were hoping to go to Fenway and put an identity more behind the location," Shikbaram said. "We are excited about what we can do under the roof but and that's part of the truly team experience. Even before we got the bowl, this is a sport that runs. Fenway Park is a great atmosphere for a guy on NFL foot."

Haley said he can harness that momentum for 2017 based on the gambling on his offensive outlook.

"The other guys are going to push us," Haley said he and his Patriots will return championship opportunities. "I said to the fans it'll remind us of being among the greats and being the champs and champion. They are going to be the lynchpin in this Big Game. And it's not because they live here, but because they're going to be the stick. Meet them, run them out of sight. To have people, that's not something that other teams have done in NFL history."

We covered how the Patriots approached January's New York vacation trip at The Syllabus.com and their spring season traveled since June. Experience our very unique 2015 Mobile Guide at http://quotepan.com .

Related<|endoftext|>A more detailed revelation of Viking's dramatic rise is now available. But The Mammals tended to rely on land expeditions to glean information. Now that more information about the world without computers or maps could accessed.

Wendy Robinson, The Arctic Lion

New Viking Denials Confirmed by Areologists

A underwater trench was once paved off off the UK coast in Viking engineers' location claimed by locals. But drilling inside the channel, sealed for a year by the distortion of the erupted Swarbrough volcano, one in November discovered 8 otheral signed of submarine submarines working against the foraminiferous seawater, accessing new information about its tunnel depths and carbon sources.

Throughout the last century leading officials swore the Viking civilisation was dead, but now archaeologists finally know why.

Adrian Ashleyman, minister in charge of the Western Isles, covers every mile of the Viking civilization for thousands of years up until the 10th century, in a presentation at the Ocean Foundation. "It's amazing how much they travel and the work has been done to get where they are, but it's pretty awful to pull nails in a far schedule. The lines need to be so easy to get across"

Click see the BBC report.

Related myths: Viking rune, USSR, Viking generals<|endoftext|>Because former whistle-blower, Edward Snowden has sent journalists and world bloggers cocktail clippings of a New York-based government security

training, Mounties Sustainability Training taught by British Commando Troopers, it's an excellent venue for journalists to get their information.

We hear that the U.S. may have bit of a rock on the memory of its golden-west. Snowden has succeeded in making the FSA comfortable shedding its hard currency, boasting to the Guardian recently that the store in which he formed his fortune in Iceland was valued at a staggering $256.5million$.

It's no different to Volkswagen World Group's overpriced program and supermarket boss, Valentino Rossi's private company which has been working on technology that can be based on Italian 500's Series 7's.

This thinking causes problems. Companies in Germany and Austria license special patents and sell them only to companies that have a record of

========== Sample 4 ==========

its proposed rules were filed. Seeking to amend the proposed rule, FERC viewed the strong language put forth by an association opposed to the exemption when it met with the statement in a letter from Dan Stern with the Sierra Club. Moreover, the National Farm Bureau and the Federation of Sportsmen and Fishermen, which owns lands for conservation, both backed the new "no-Mine" rules. But FERC didn't allow for this. "We had to try to craft an artificial, complicated substantive rule where power was responsible for conservation that the industry doesn't want at all," Butler says.

The new RRE rules leave farmers with only a title to stake each acre of unused land to mine copper. Buildings must be fewer than 10 feet tall, and need to fitted to a VVAC pole to allow the same air flow. No more than 25 acres are permitted and a 30-acre farm can be mined using hydraulic fracturing. They have to prevent the potential mining operation, built a fence, took part in field tests, and gathered evidence from the site of having found copper that has trampled on production lines.

FERC did not dispute rule specifics—no specific copper facilities the no-mine would be laid—but did release an assessment saying that public processes would determine "the cutting and the use of pipes" and just what amount of public land to be included in maps should be limited. "It's all on a market herhorse," Slaughter says. "It's a de facto emissions test."

Nutter warned that the regulations might actually result in "a more anti-competitive system" than one that does allow for polluting, subdivision and enforce strong rules. "The industry does not see it right and rejects the idea that it's worse because it increases pollution," he says. "The Public Works Committee would find it even worse."

Public approval for a draft rule will reportedly take 12-16 months as well as legal challenges.<|endoftext|>A coalition of ex-elites gathered at Laurelview High School this past week. They're part of the Coalition (recent graduates and teachers from urban county were touted for tenure) then showed them how to fight neighborhood segregation. They are so-called Chiefs in the program, and they recruit more black kids through. They simple this practice that is commonplace in large cities and observed pro-Confederate celebrations.

I joined Michael Smart, coordinator of that program, at his rock concert, Rebel Wars, and watched a video of Confederate flag shootouts outside the program.

The group of eight black students lectured and asked teacher Vincent Henderson about characteristics of black communities.

Potish, "Most of us were in tears," following decades of history.

"Then we were surprised," Henderson said. "We were almost sure we had it with us."

"We were thinking about it," Smart said. "We just weren't expecting it."

"The CMO brought us back to races of neighborhoods that are African American. You're talking about lead epidemic and youth unemployment numbers," Smart continued. "You've reported a dramatic employment loss on top of low personal drug using. And you're gonna see a lot of student prejudice and crime in the process."

Though they only a few seconds to tap into the students' attitude with statements like "Motivation, Director of Excellence," officials did acknowledge the past positive attitudes and report they were raising them.

I heard much more Cal sentiments from the rest of the presentation, albeit somewhat-coarse ones.

"They did not apologize for classroom clashes, not even the token degree," Smart told me. "We're not converts."

They made light of the school's efforts in Combat Simulator, a battlefront that drew resentment after the Burbank response of black students.

"They want to know about our diversity program," Principal Frank Fund interrupted. "They make assumptions about who wants them to live and police where they want them to live."

"We actually want kids like these," Fund said. "Eventually, that'll be the biggest part of it."

Historically, many in Chicago's gated neighborhoods have been little but crippled. Since 1992, a handful of private universities where black leads have developed a detrimental regime where they send low-income students from the disadvantaged through Northern schools to improve opportunities for minorities.

For a rural district full of Latino immigrants, these black-only programs have Tucson's black students underperforming at disproportionate rates to kids from whites, Hispanics, and other groups. The effects are little-noticed, and rarely openly presented.

Jew's album, Jazz is now played widely, performing nationwide and last year winning an award in collaboration with the Sierra Club, and the Parents Association of Arizona. They call on the district to address low rates of distance and point to the research that comes out against Unified.

"We're putting black children at a disadvantage," says Jew, who has added Jazz to Spotify himself. "

========== Sample 5 ==========

Dr, S Scott Road, LPM, NY 57025 Hours: 7P.M. – 9 PM (10 a.m service; 50 foot Zion TeaPot; chef Beleo, LPMO, chefbeleo.com); Dinner at 7 p.m Tickets online H22D through tickets [404] 775-624-3404 ; public (202) 769-6678.

Website, ChefBalleo.com

Line 4665/4667<|endoftext|>Accelerated around A number of doors with bays to divert between two types of facility. One venue houses the "Em Lion" training and the second one for Saturday night gatherings. Event tickets will be given out hand handed with or without ID to pay. The vast majority of event will be inside the Center for people to catch up with our manager and owner free ops. For partygoers there will be entertainment at every level from the alley and sliding ramp to hooters skateboarding, and pools.

POLRY days temporary weekends and dynamic peoples schedules

Saturday band on all day never has before. Neither school nor college

Must use good for jams between the residence drive to police

Only vintage, topping, tattooing (except for Halloween) and older patrons update our article in two weeks

Place Events

Chocolate: 6500-10000 people coming to town Night manager is on 24 hours and has control of the event (transportation times, gives permits to Huppi-Pickers, funding license to city agency to claim, and team with the owner and operator of the business] 300 people Total event is double 300 Children Lengths must be > 30 vs old chance

Music will be live hand tapping (as will Ad and Staging throughout the building)

Special ticket bidding "Cool City" ends Jan 19 which will last the day after.

Registration 5 slots as long as per picture the touch machines using 5 keys Trans-card from Ticketmaster will fund rates Food Trucks/ door.

The event is after the princess's reception is in each auditorium featuring live music, jazz, and theater as the walk out to athlete camp commencement rehearsals before night. There will be separate event for kids to get to these and other things for the kids to meet the party.

Tickets

Pick up on the private chapel events etc. these are not major facilities. Last come first serve

Tickets: $10orless + fees$

Believe that the jQuery art space is a space that is always open

Saturday operating license called business day or city license

No license $95treatandfoodservice$

Entertain to the Saturday breakfast event

Featuring a dinner atmosphere at most events.

"Lovers in Trust" Auction designed to get kids and people from the neighborhood get to know some other

Cent hours chance to win charity items

hold a laptop and tablet inside during the Emperor's Lion Training session.

All photos have been stolen from the art event

Security department approved for benefits only.

Notes

Lowers: $50Token$

(From @shermark)

4.Mairesomata,

Here is the first image of some of our staff that have gotten hurt in this space | Pic (from @Dunader)

How to comment well from the months in this space based on being able experienced in so much politicotation

5. Letters

Here is a season 15 video of Elsa on Lierson handled this space and what is happening, how to congratulate staffers (see here for first time public announcement October 2005), how to give free tickets to the prom.<|endoftext|>This spring, meet at a Chipotle University of Iowa, Ames location, representatives of the Chipotle's signature chicken giant said.

An event has been put together by the Native American Union for FEED Friday from 9-11 p.m. Details on exact locations for the event weren't available.

Si que héres Inigo, SE ANNOCALE CHICK 10 – Inigo for the 27th date on May 7th!! — Equalization (@Cornish Alliance Equalization) March 27, 2016

Click here for onscreen printable image of the location – 700 Mr. Petty Avenue.

Privacy

In the release, which had more details on Chipotle, the University's College of Agriculture and Economics said, "Georgetown and its partner specializes in protein, pirouin, cellophane, and tinder-minder products."

Meanwhile, more past Chipotle showings:

LUS / GRODYS Club

LatINO STARS Launchday starts at 2 a.m. Friday when Las Vegas's Graduation Day Lounge will be on hand (bar not applicable)

9 p.m. on Friday, April 7.

Shillam

H.2 Samples Generated by the Variable-Length Model Trained on Stories

========== Sample 1 ==========

once there was a boy who liked to explore - even though he was small. every day he ' d struggle to see what he could find. he would pick small foods in his garden and visit his neighbours. one day, he noticed a delicious sauce on the food. he was so tempted to try it out! he struggled to eat it, not as usual, so he decided to buy a popular pasta with sauce. he shared it with all his friends who enjoyed their sauce. but, he refused when he urged them to keep buying. in consideration, someone called out to him : " go ahead, try the sauce. what if the sauce means you won ' t have to really savor the taste? " the boy was happy that he couldn ' t resist the temptation to try the sauce. from then on, everyone in the neighbourhood remembered the popular sauce and dreamed of becoming a more daring sauce expert.

========== Sample 2 ==========

john was a very popular three year old boy in his town. he liked to play in his backyard in the grass. every day, he would imagine all the different things the world started to offer. one day, john looked up in the sky and came up with an idea. he got off his bike to learn the wisdom. he rode back to the meadow and told the wise owl about the wisdom. the owl said she said the world was full of special and sweet wisdom if he was patient. after a long day, the wisdom yielded a lot. john thanked the wise owl. she had worked hard and was able to learn something magical. john continued on his bike, soon out of the meadow. he saw a beautiful butterfly and smiled as attractive as she had seen him.

========== Sample 3 ==========

tim was playing in the park one day. he skipped around, shouting. it was a game but there were no toys around. he was feeling disappointed and soon saw a boy. the boy was a lot bigger than no other boy but no one ever seemed to understand. he recognized it from a playground. the boy asked if he wanted to join him. tim hesitated at first for a few moments. but he felt a bit scared, so he decided to follow the boy. he started to climb up the ladder at first but as he went higher and higher he felt helpless. the boy saw tim and said he was looking for help. tim was gentle and asked to try to lift him up the ladder but he was very worried. after a

while the boy said goodbye to tim and ran off. tim felt sad that he had to go home without to help a friend. he regrets climbing the ladder.

========== Sample 4 ==========

timmy and lily are friends. they like to play together. one day, they see a lady on a bench. she has a big bag and a red stick. the candy is sweet and bitter. lily wants to eat candy too. she reaches into her bag and takes the candy. " no, lily! " timmy says. " that is my candy. you can ' t have it. i want this candy. it is a candy for you. " lily did not listen. she does not want to share her candy. she tries to part with her candy with the stick. but the stick is stuck. it stays loose and hurts her teeth. " ow! " lily cries. " you broke my stick! " timmy laughs. he drops the stick and pushes lily away. " no, thank you! " lily says. " you can ' t have my candy. go away now. " tom does not find another stick. he did not know how to have it. he bites the stick and pulls it out. " look, lily, i am eating candy! " tom says. " am i eating kite? " lily looks at tom. she saw what tom was doing. she feels silly and scared. she takes off her hat and her shoes and pants. " oops! sorry, tom, " timmy says. " i did not mean to scare you. i just wanted to bite the stick. maybe we can play something else. " " i know, tom, " lily says. " but i still like the stick. it is better. you can have another candy. " tom feels bad. he also likes lily. he thinks lily is generous. he gave lily a big, red apple. " here, lily, " he says. " this is a cake. it was not magic. it was a secret. " " thank you, lily, " tom says. " but next time, do not eat the stick. you are a good friend. can i help you? " lily smiles. she is glad that tom could help. she was generous. " okay, lily, " tom says. " i will share the stick. we can play together. " they play with a ball, a kite, a book and other toys. they have fun with the ball. they are happy.

========== Sample 5 ==========

once upon a time there was a boy named jack. he was worried because he had no worries with him, every day him would carry jack away for a day at work the first time to eat a yummy lunch. one day jack saw some of his mum outside. they were so kind and helpful to all of them. they did not put a heavy box at school or they brought any more delicious treats. jack told his mum about this because he went to work looking for something else to do. jack soon had an idea : he bought a lunchbox and place all of his mum ' s lunch. he carried each piece, and then drive home. his mum was so relieved and smiled. jack was so relieved and happy. from that day on, jack always carried his lunch with him everyday, and people never forgot to go home without a worried look.

========== Sample 6 ==========

once there was a dog named spot. spot had a big toy that was very hairy. he loved to play with it and hug him. every day spot would sign when he would get a slapped race. one time, spot was ready to show his signa€ ” black pawch. he wanted to show his friends how fast he could sign? when he signed with his hand, his friends saw him close by. they liked the show. the show was very funny. when spot managed to sign his name, his friends would even make clapping. in the end, spot and his friends all got slap spots! they are now the best! all day long, spot is signing, and even his hand feels good. everyone around him is happy every week. because of the day, spot brought a prize for his slapch with him.

========== Sample 7 ==========

once upon a time there was a gorilla. he was big and very strong. he stored food in a crack. the gorilla enjoyed the food and enjoyed it. one day, a little boy was watching the gorilla by himself. he got stuck in a crack. he felt scared and demanded the gorilla some help from it. he knew what to do. he asked the penguin for help. the penguin grabbed a big rock and swam over to the crack. the gorilla worked very hard with his help from the penguin. the little boy was so happy. he ate his food and the penguin thanked him for the sweet treat.

========== Sample 8 ==========

once upon a time, there was a little girl named lily who loved to play with her ball. every day, she would go to the park and send her ball to him inside a loud song. one day, lily saw a mean boy named max playing

his guitar. he was playing his song and was not friendly. lily wanted to help max, so she played with him and made the music beautiful. max became friendly and welcomed lily with her ball. they had a great time together and at the end of the musical song, max ended nicely and said goodbye. lily felt sad that he was not sharing nicely. lily went home and used a plan. she brought out her guitar and spoke to max. she told him that he wasn ' t nice and allowed her to hold the guitar. max was happy and grateful. from that day on, lily and max didn ' t matter what friends tried from each other.

========== Sample 9 ==========

once upon a time, there was a messy little boy named timmy. timmy liked to play with his toys and do chores. one day, timmy ' s stomach felt really hungry. his tummy was growling, and his food was all over the floor. his mom saw that timmy had eaten all his food, and she knew the bad smell came from lunch. timmy ate it, but she found it was disgusting and looked at the napkins on the floor. she said, " timmy, napkins are not good to eat. " she scoldolded timmy and asked him again before eating his meals. she said, " you don ' t know you would have gotten sick, though. " timmy said, " i ' m sorry, i won ' t eat them next time. " he put his hands on the napkins and threw them away so his clothes were safe from harm. from then on, timmy made sure to listen to his mom and eat his napkins before she went to eating. he never wanted to eat his napkins again. the end.

========== Sample 10 ==========

anna liked to play with the sack. she liked to pull things out and see what she could find. she asked her dad sometimes. he explained that she was anything she could find. one day, anna found the sack in the field. she looked inside and saw her friend, lily, who was playing with a dog. she opened the sack and saw many shiny things. anna was happy. she wanted to explore more. but then, she saw a big black cat. the cat was sitting on the edge of a bush. it had dirty eyes. anna did not know what it was. she thought, " maybe it is the cat. i can make it happy. " she went to the shiny thing and stroked it gently. the cat did not go away. it hissed. it said, " no, anna, what are you doing? go out! let ' s stay behind the tree. " anna did not listen. she stepped closer to the cat. she reached her hand out. she pulled its head. the shiny thing moved. it made a ring. it came alive with a band. it moved and snapped. anna heard the snap. she screamed. she ran back. the cat bit her. it hurt her finger. she could not hold it. she cried, " ow, ow, ow, ow, it hurts! it hurts, it hurts! " anna ran to the field. she saw the fence and the other people. she saw the cat too. she heard the people shouting and running. she saw the cat in the tree. she ran to get help. she looked for lily. she saw her mom walking in the car. she saw lily ' s finger. she felt sorry for her. she gave her ring to her. she went to lily and said, " i ' m sorry, mom. i was wrong. lily was naughty. she tried to grab lily ' s ring. " her mom smiled and hugged lily. she said, " it ' s okay, anna, it ' s okay. it doesn ' t hurt. but it ' s not your fault. you should have left the sack alone. " anna was still sad. she said, " i ' m sorry, mom. i love you, lily. " she hugged lily and learned her lesson. she never came back. she hugged lily and went home. she left the cat alone. where lily was sad. she did not harm it.

========== Sample 11 ==========

once upon a time, there was a little girl named lily. she loved to play with her toys and pet dog, max. one day, she went to the park with her mom and saw a leopard flying from its back to its home. she excitedly asked her mom, " what is that, mommy? " her mom said to lily, " this is polly. this leopard is very cheap, so if it misses its home, i can take it to you. " lily listened carefully and said, " thank you, mommy. polly is so kind. can i recommend it to a party for you at the park? " later that week, lily and max watched the leopard and when it arrived, the leopard landed on lily ' s fur. she was so excited! she learned that the leopard had a modest quality on its home. later that day, lily forgot about the leopard and went to bed for a nap. she fell asleep with max, just like the modest parrot. when she woke up, she thanked max for helping her.

========== Sample 12 ==========

once upon a time there was an ancient, grey kangaroo. he lived in a sunny spot near a pond. one night he heard a magical sound. it was a voice coming from the sky. startled, the kangaroo called out. he didn ' t know what to do. but then he noticed a small, red balloon glimmering in the air. he curiously approached the alien

and it turned into a truck blocking his direction. it was carrying an old gas truck and it was perfectly heavy. " what is this? " the voice replied. the gasoline truck quickly zaneded the balloon and released it into the sky. the kangaroo almost entered an ancient castle and it was even more beautiful than than he had ever been there before. suddenly the gas truck spoke with a hop and the voice said, " this is a gas castle! you can ' t come back! " the kangaroo was very scared, but he had to figure out the alien ' s names. he ran higher and higher, and the dragon slowly started to slowly spring until it was gone. the kangaroo thought it was gone. the next morning, the ancient kangaroo returned to where the alien had returned. he looked back, not clear with fear - he was even more relieved to see that the magical creature had given him back. he had emerged in the right circle!

========== Sample 13 ==========

once upon a time, there was a little rabbit named benny. benny loved to hop around the forest and play with his friends. one day, benny met a friendly rabbit named rosie. benny introduced rosie to his new friend. " what ' s that? " he asked. " that ' s a toy staff, " said rosie, " it ' s a loud sound that makes me reverse. " benny didn ' t understand what that meant, but rosie continued. benny asked rosie if he wanted to teach rosie her name staff. rosie said, " i don ' t know it, a toy staff means my number 10.... " benny smiled and said, " okay, let ' s count to ten. " benny and rosie learned how to reverse from ten over half cards. they had so much fun counting and learning, they played with the toy staff all day long. when benny and rosie grew up playing, they had so much fun together and were able to play with their new name toy staff.

========== Sample 14 ==========

once upon a time there was a little boy named john. john had a grain wheat mill, who worked there. one day john went to the mill to make flour. he picked out some flour and put the wheat in the corn mill. as he ran over to his house, the flour slowly churned and added it to a slice of the mill. john smiled at what he had done. when he was finished when he rolled out his door, he met another boy called bob. bob said, " would you want to play with me? ". john smiled and said, " yes! " so they had fun playing and running around the mill. when john was done, bob knew exactly how bob had done it. he held up the grain and carried them both hands. he said, " we have to take care of our wheat and be balanced for what we need ". john was balanced with excitement. he welcomed him, gave him a pat on the shoulder of my mill and was extremely pleased. bob laughed, and told john that day he and the wheat were friends forever.

========== Sample 15 ==========

ben and mia like to play on the navy boat on the sea. they pretend they are divers and fight sharks in space. one day, they go to a big island with their house. it is very pretty and green. it has a long sail, a flag, and a blue door. they open the door and see a lot of water in the water. they lower the boat and look over the boat. " hello, many ships! " mia says. " maybe they are pirate? " ben wonders. " maybe they are treasure, " mia says. " or other ships. we are swimming and looking for something to look for. " they see a big ship in the water. it is red and yellow and has a sword. " maybe it ' s a land from here, " ben says. " what ' s a port? " mia says. " maybe it ' s a pirate! " " who is the treasure? " ben says. he dived close and lands on the ship at mia. he lands on mia next and puts on a scarf. she smiles and laughs. " did you hear the sound of the treasure? " ben asks. " one, two, 3, blast off! " mia says. she smiles and jumps high. she sees the sky, the sun, a shark, a butterfly, a shark, and a duck. they jump off the ship and run to the shore. they call their mom and dad. " mom, look what we found! " ben says. " we are in space! " mia says. she hugs their mom and dad. they tell their parents about their adventure. " we dive in the air and dive with a ship, " ben says. " we are great and brave explorers! "

========== Sample 16 ==========

once upon a time there was a young boy named sam. sam was very happy because he had a nice bench. every day he used it to sit in the park. one day, he was sitting in the park on the bench while a boy came and slapped his hand. " ouch " the boy said the boy. sam looked up in surprise and started to cry. the kind boy said, " i have an idea to help the young boy. you can use your cane to make hi! it ' s all okay ". sam wiped up his tears

and looked at the young boy in the bench. it was a kind surprise that the boy had seen him before. he said, " you ' re not as good as nice as the bench you like and why did who slap my hand? ". the young boy smiled and told sam that hea€™s quarrel with a friend. he said, " let ' s write down my bench and draw it. it looks like a special book ". the young boy smiled and said, " yes, let ' s draw on the bench ". sam and his friend spent lots of time playing and talking. they wrote and drew the bench and gave it a hug. sam and the boy enjoyed the bench together for many years went by and their canes were over, so they were happy.

========== Sample 17 ==========

once there were two friends, daniel and norman. they were mighty friends, who loved to play baseball. they would practice their best in every golf game. even on this competitive day, norman still wanted to score a goal without not being four goals as rough as norman. one day, daniel sneaked into norman ' s lead up getting his best grades. johnnie said, " let ' s invite our team to try it out. in the starting game! " the team was excited, they called the veterinarian. norman dashed to the chess room and daniel asked for a tough stick. johnnie took out a tough stick and finally swung it against his friend in the second seat. they both cheered as norman was proud that they could continue on their physical soccer game. at last, they were able to score successfully.

========== Sample 18 ==========

once upon a time, there was a little boy named timmy. timmy had a toy weapon, a brown sword. one day, timmy ' s mom came in and asked him to stop playing with his sword. but timmy didn ' t let go of his weapon and wanted to show her where he found it. " timmy, your weapon is on your bed, " his mom said mad, but timmy didn ' t listen. later, timmy ' s mom asked him to clean up my room and put it away. " mom said i will shoot away from your car, " she replied. timmy did a great job cleaning up the room with his dolls and his weapon was very messy. but when he came back, his sister came in and saw that all of his toys were gone. timmy couldn ' t find his toy anywhere. he looked in the closet, in the closet, and even in the closet. his sister searched everywhere, but she couldn ' t find it. timmy felt sad because he didn ' t want to make her worry. finally, he told his sister that he did not think he was stupid of any extra toys. his sister decided to find his toy and they searched the treasure together. the sword was found at its spot and timmy was glad no one could spot it.

========== Sample 19 ==========

once upon a time there was a modern woman. she had a very important job and she was very careful with it. each day she looked at her job, as it was a great job. then one day, the woman heard a loud noise. she looked around, looking for help. she started to look out there. suddenly, she saw a little girl running towards a 3 year old policeman. she was running and rushing to avoid her. the policeman stopped and said, " are you alright? i ' m so scared and lost. " the elderly woman went to find the cop and saw how brave the little girl was. he asked her if she wanted to help. the little girl told her yes and the cop started hunting. but the little girl bit her and threw everything away. in the end, the woman ' s heart was left increased and into distress. the moral of the story is that we must always be careful when we are hunting, especially when someone is about three years old and the little girl needed help. if we argued with someone, or else remains in our life, we can start causing consequences. this story requires a serious ending, figuring out those who are looking out for help and can cause trouble.

========== Sample 20 ==========

once upon a time, lily and max were best friends. one day, they found a delicate flower in the grass. it was very delicate with lots of spiky petals. max loved flowers and wanted to hold it. but the flower was too big for him and max couldn ' t hold it very much. but then, they found a snapwig and the flower fell and landed in their hands. it burst in two! lily was so happy that max found a beautiful flower they could rectangle. they decided to build a castle in the castle. lily looked for pretty stones as they found, but even though it was hard for them to find that stones would be making the castle stronger. so, they promised to keep the castle strong

and delicate from one day on. they played together every day, and from that day on, the delicate flower made its castle strong and attractive.

H.3 Demonstration of the Intermediate Generation Process

========== Step 0 ==========

[CLS]

========== Step 1 ==========

[CLS] lucy

========== Step 2 ==========

[CLS], lucy her

========== Step 3 ==========

[CLS], lucy very her.

========== Step 4 ==========

[CLS], lucy lucy very her.

========== Step 5 ==========

[CLS], lucy smile lucy, very her.

========== Step 6 ==========

[CLS], lucy. smile lucy she, very her.

========== Step 7 ==========

[CLS], lucy. smile lucy she, very her.

========== Step 8 ==========

[CLS], lucy. smile. lucy she and, very her.

========== Step 9 ==========

[CLS], lucy. it smile. lucy she and, very her.

========== Step 10 ==========

[CLS], lucy. it smile. lucy she and, very after her.

========== Step 11 ==========

[CLS], lucy. it smile. lucy she and, very after her. her

========== Step 12 ==========

[CLS], lucy.. it smile. lucy she and, very after her. park her

========== Step 13 ==========

[CLS], lucy.. it smile. lucy she and, very after her. park her.

========== Step 14 ==========

[CLS], lucy. a. it smile. lucy she and, very pretty after her. park her.

========== Step 15 ==========

[CLS], lucy. a. it smile. lucy she and farm, very pretty after her. park her.

========== Step 16 ==========

[CLS], lucy. a. it smile. lucy she the and farm farm, very pretty all after finished her. park her.

========== Step 17 ==========

[CLS], lucy. a. it smile. lucy to she the and farm farm,. very pretty playing all after finished her. park her.

========== Step 18 ==========

[CLS], there lucy. a. it smile. lucy to she the and farm farm,. very pretty playing all after finished her. park her.

========== Step 19 ==========

[CLS], there lucy. a. it smile. lucy to she the and farm farm,. onion very pretty playing all after finished her. to park her.

========== Step 20 ==========

[CLS], there lucy. a. it and smile. lucy to. she the and farm farm,. onion very pretty playing all after finished her. to park her mom.

========== Step 21 ==========

[CLS], there lucy. a. it and smile. lucy to. she the and farm farm,. onion very pretty playing all after finished her. to park her mom more.

========== Step 22 ==========

[CLS], there lucy. a. it and smile. lucy to. she saw the and farm farm,. onion very pretty playing all after finished her. to park her mom the more fun.

========== Step 23 ==========

[CLS] a, there lucy. a. it and smile. lucy the to. she saw the and farm farm,. onion very pretty playing all after finished her. to park her mom the more fun.

========== Step 24 ==========

[CLS] a, there lucy. a. it and smile. lucy the to. she saw the and farm farm,. onion very pretty playing all after finished her. to park her mom the more fun.

========== Step 25 ==========

[CLS] a, there lucy. a. it and smile. lucy the to. she saw the and farm farm,. onion very pretty playing all after finished her. to park her mom the more fun.

========== Step 26 ==========

[CLS] a, there lucy. a. it and smile. lucy the to. she saw the and farm farm,. onion very pretty playing all after finished her. to park her mom the more fun.

========== Step 27 ==========

[CLS] a, there lucy. a. it and smile. lucy the park to. she saw the and farm farm,. onion very pretty playing all after finished, her. to park her mom the they more fun.

========== Step 28 ==========

[CLS] a, there lucy. a. it and smile. lucy the park to. she saw the and farm farm,. onion very pretty playing all after finished, her. to park her mom the they more fun.

========== Step 29 ==========

[CLS] a, there was lucy. she a. it and smile. lucy the park to. she saw the and farm farm,. onion very pretty playing all after finished, her. to park with her mom the they more fun.

========== Step 30 ==========

[CLS] a, there was lucy. she a. it attractive and smile. lucy the park to. she saw the and farm farm,. onion very pretty playing all after finished, her. to park with her mom the they more fun.

========== Step 31 ==========

[CLS] a, there was lucy. she a. it attractive and smile. lucy the park to. she saw the and farm farm,. onion very pretty playing all after finished, her. very to park with her mom the they more fun the.

========== Step 32 ==========

[CLS] a, there was lucy. she a. it attractive and smile. lucy the park to. she saw the and farm farm,. onion very pretty playing all after finished, her. very to park with her mom the they more fun the.

========== Step 33 ==========

[CLS] a, there was lucy. she a. it attractive and smile. lucy the park to. she saw the and farm farm,. onion very pretty. playing all after finished, her. very to park with her mom the they more fun the.

========== Step 34 ==========

[CLS] a, there was lucy. she a. it attractive and smile. lucy the park to. she saw the and farm farm,. onion very pretty. playing all after finished, her. was very to the park with her mom the they more fun the.

========== Step 35 ==========

[CLS] a time, there was lucy. she a. it was attractive and smile. lucy the park to. she saw the trees and farm farm,. onion very pretty. playing all after finished, her. was very to have the park with her mom the they more fun the.

========== Step 36 ==========

[CLS] a time, there was lucy. she a. it was attractive and smile. lucy the park to. she saw the trees and farm farm,. onion very pretty. playing all after finished, her. was very to have the park with her mom the they more fun the.

========== Step 37 ==========

[CLS] a time, there was lucy. she a. it was attractive and smile. lucy the park to. she saw flowers the trees and the farm farm,. onion very pretty. playing all after finished, her. was very to have the park with her mom the they more fun the.

========== Step 38 ==========

[CLS] a time, there was lucy. she a. it was attractive and smile. day lucy the park to. she saw flowers the trees and the farm farm, onions. onion very pretty. playing all after finished, went her. was very to have the park with her mom the onion they more fun the.

========== Step 39 ==========

[CLS] a time, there was lucy. she a. it was attractive and made smile. day lucy the park to. she saw flowers the trees and the farm farm, onions. onion very pretty. playing all after finished, went her. was very to have the park with her mom the onion they more fun the.

========== Step 40 ==========

[CLS] a time, there was lucy. she a. it was very attractive and made smile. day lucy the park to. she saw flowers the trees and the farm farm, onions. onion very pretty. playing all after finished, went her. was very to have the park with her mom the onion they more fun the.

========== Step 41 ==========

[CLS] a time, there was named lucy. she a. it was very attractive and made smile. day lucy the park to. she saw flowers, the trees and the farm farm, onions. onion very pretty. playing all after finished, went her. was very to have the park with her mom the onion they more fun the.

========== Step 42 ==========

[CLS] a time, there was named lucy. she a. it was very attractive and made smile. day lucy the park to. she saw flowers, the trees, and the farm farm, onions. onion very pretty. playing, all after finished, went her. was very to have the park with her mom the onion they more fun at the.

========== Step 43 ==========

[CLS] a time, there was named lucy. she a. it was very attractive and made smile. day lucy the park to. she saw flowers, the trees, and the farm farm, onions. tomatoes onion very pretty. playing, all after finished, went her. was very to have the park with her mom the onion they planned more fun at the.

========== Step 44 ==========

[CLS] a time, there was named lucy. she a. it was very attractive and made smile. day lucy went the park to. she saw flowers, the trees, and the farm farm, onions. tomatoes onion were very pretty. playing, all after finished, went her mom. she was very to have the park with her mom the onion they planned more fun at the.

========== Step 45 ==========

[CLS] a time, there was named lucy. she a. it was very attractive and made smile. day lucy went the park to. she saw flowers, the trees, and the farm farm, onions. orange tomatoes onion were very pretty. playing, watered all after finished, went her mom. she was very to have the park with her mom the onion they planned more fun at the.

========== Step 46 ==========

[CLS] a time, there was named lucy. she a. it was very attractive and made smile. day lucy went the park to. she saw flowers, the trees, and the farm farm, onions. orange tomatoes onion were very pretty. playing, watered all after finished, went her mom. she was very to have the park with her mom the onion they planned more fun at the.

========== Step 47 ==========

[CLS] a time, there was named lucy. she a. it was very attractive and made smile. day lucy went the park to. she saw flowers, the trees, and the farm farm, onions. orange tomatoes and onion were very pretty. playing,

watered all after finished planting, went her mom. she was very to have seen the park with her mom and the onion they planned more fun at the.

========== Step 48 ==========

[CLS] a time, there was named lucy. she a. it was very attractive and made smile. day lucy went the park to. she saw flowers, the trees, and the farm farm, onions. orange tomatoes and onion were very pretty. playing, watered all after finished planting, went to her mom. she was very to have seen the park with her mom and the onion they planned more fun at the.

========== Step 49 ==========

[CLS] a time, there was named lucy. she a dress. it was very attractive and made smile. day lucy went the park to. she saw flowers, the trees, and the farm farm, lucy onions. they orange tomatoes, and onion were very pretty. playing, watered all after finished planting, went to her mom. she was very to have seen the park with her mom and the onion they planned more fun at the.

========== Step 50 ==========

[CLS] a time, there was named lucy. she a dress. it was very attractive and made smile. day lucy went the park to. she saw flowers, the trees, and the farm farm, lucy onions. they orange tomatoes, and onion were very pretty. playing, watered all after finished planting, went to her mom. she was very to have seen the park with her mom and the onion they planned more fun at the.

========== Step 51 ==========

[CLS] once a time, there was named lucy. she a dress. it was very attractive and made smile. day lucy went to the park to. she saw flowers, the trees, and the farm farm, lucy saw onions. they orange tomatoes, and onion. were very pretty. playing, watered all after finished planting, went to her mom. she was very to have seen the park with her mom and the onion they planned more fun at the day.

========== Step 52 ==========

[CLS] once a time, there was named lucy. she a dress. it was very attractive and made her smile. day lucy went to the park to. she saw flowers, the trees, and the farm farm, lucy saw onions. they orange tomatoes, and onion. thought were very pretty. playing, watered all after lucy finished planting, went to her mom. she was very to have seen the park with her mom and the onion they planned more fun at the day.

========== Step 53 ==========

[CLS] once a time, there was named lucy. she a dress. it was very attractive and made her smile. day lucy went to the park to. she saw flowers, the trees, and the farm farm, lucy saw onions. they orange tomatoes, and onion. thought were very pretty. playing, watered all vegetables after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion they planned more fun at the day.

========== Step 54 ==========

[CLS] once a time, there was little named lucy. she a dress. it was very attractive and made her smile. day lucy went to the park to. she saw flowers, the trees, and the farm farm, lucy saw onions. they orange tomatoes, and onion. thought were very pretty. after playing, watered all vegetables after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion they planned more fun at the every day.

========== Step 55 ==========

[CLS] once a time, there was little named lucy. she a dress. it was very attractive and made her smile. day, lucy went to the park to. she saw flowers, the trees, and the farm farm, lucy saw onions. they orange tomatoes,

and onion. thought were very pretty. after playing, lucy watered all her vegetables after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion they planned more fun at the park every day.

========== Step 56 ==========

[CLS] once a time, there was little named lucy. she a dress. it was very attractive and made her smile. one day, lucy went to the park to. she saw flowers, the trees, and the farm farm, lucy saw onions. they orange tomatoes, and onion. thought they were very pretty. after playing, lucy watered all her vegetables after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion they planned more fun at the park every day.

========== Step 57 ==========

[CLS] once a time, there was a little named lucy. she a nice dress. it was very attractive and made her smile. one day, lucy went to the park to. she saw flowers, the trees, and the farm the farm, lucy saw onions. they orange tomatoes, and onion. thought they were very pretty. after playing, lucy watered all her vegetables after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion they planned more fun at the park every day.

========== Step 58 ==========

[CLS] once a time, there was a little named lucy. she a nice dress. it was very attractive and made her smile. one day, lucy went to the park to. she saw the flowers, the trees, and the farm the farm, lucy saw onions. they orange tomatoes, and onion. thought they were very pretty. after playing, lucy watered all her vegetables after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion they planned more fun at the park every day.

========== Step 59 ==========

[CLS] once upon a time, there was a little girl named lucy. she had a nice dress. it was very attractive and made her smile. one day, lucy went to the park to play. she saw the flowers, the trees, and the farm. the farm, lucy saw many onions. they orange tomatoes, and onion. thought they were very pretty. after playing, lucy watered all her vegetables after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion they planned more fun days at the park every day.

========== Step 60 ==========

[CLS] once upon a time, there was a little girl named lucy. she had a nice dress. it was very attractive and made her smile. one day, lucy went to the park to play. she saw the flowers, the trees, and the farm. the farm, lucy saw many onions. they orange tomatoes, and onion. she thought they were very pretty. after playing, lucy watered all her vegetables after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion they planned more fun days at the park every day.

========== Step 61 ==========

[CLS] once upon a time, there was a little girl named lucy. she had a nice dress. it was very attractive and made her smile. one day, lucy went to the park to play. she saw the flowers, the trees, and the farm. at the farm, lucy saw many onions. they orange tomatoes, and orange onion. she thought they were very pretty. after playing, lucy watered all her vegetables after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion. they planned more fun days at the park every day.

========== Step 62 ==========

[CLS] once upon a time, there was a little girl named lucy. she had a nice dress. it was very attractive and made her smile. one day, lucy went to the park to play. she saw the flowers, the trees, and the farm. at the farm, lucy saw many onions. they were orange tomatoes, and orange onion. she thought they were very pretty.

after playing, lucy watered all her vegetables after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion. they planned more fun days at the park every day.

========== Step 63 ==========

[CLS] once upon a time, there was a little girl named lucy. she had a nice dress. it was very attractive and made her smile. one day, lucy went to the park to play. she saw the flowers, the trees, and the farm. at the farm, lucy saw many onions. they were orange tomatoes, and orange onion. she thought they were very pretty. after playing, lucy watered all her vegetables. after lucy finished planting, went to her mom. she was very happy to have seen the park with her mom and the onion. they planned more fun days at the park every day.

========== Step 64 ==========

[CLS] once upon a time, there was a little girl named lucy. she had a nice dress. it was very attractive and made her smile. one day, lucy went to the park to play. she saw the flowers, the trees, and the farm. at the farm, lucy saw many onions. they were orange tomatoes, and orange onion. she thought they were very pretty. after playing, lucy watered all her vegetables. after lucy finished planting, she went to her mom. she was very happy to have seen the park with her mom and the onion. they planned more fun days at the park every day.