Neural Continuous-Time Markov Chain: Discrete Diffusion via Decoupled Jump Timing and Direction

Jingyuan Li[2,3,4*] **Xiaoyi Jiang**[1,4*] **Fukang Wen**[1,4] **Wei Liu**[3]
Renqian Luo **Yi Zhu**[1,2,4] **Zuoqiang Shi**[1,2,4] **Pipi Hu**[2,4†]
[1]Tsinghua University
[2]Beijing Institute of Mathematical Sciences and Applications
[3]Wuhan University [4]MathonAI

Abstract

Discrete diffusion models based on continuous-time Markov chains (CTMCs) have shown strong performance on language and discrete data generation, yet existing approaches typically parameterize the reverse rate matrix monolithically—through proxies such as concrete scores (SEDD) or clean-data predictions (MDLM, GIDD)—rather than aligning the parameterization with the intrinsic CTMC decomposition into jump timing and jump direction. We propose **Neural CTMC**, which exploits the underlying Poisson structure of CTMC dynamics by separately parameterizing the reverse process through an *exit rate* (when to jump) and a *jump distribution* (where to jump) via two dedicated network heads. We show that the evidence lower bound (ELBO) reduces to a path-space KL divergence between the true and learned reverse processes that factorizes into a Poisson KL for timing and a categorical KL for direction, and admits a tractable, gradient-equivalent and consistent loss. Experimentally, scored by Gemma2-9B, our pure-uniform Neural CTMC achieves 16.36 generative perplexity on TinyStories (vs. GIDD 37.60 and MDLM 42.66). On OpenWebText, it attains the best perplexity at the same training-token budget across 16–256 sampling steps among the methods we compare (e.g., at 128 steps: Neural CTMC 189.2 vs. uniform SEDD 207.3 and uniform GIDD 232.1). Pretrained weights and inference code are available at `https://huggingface.co/Jiangxy1117/Neural-CTMC`.

1 Introduction

Diffusion models have achieved remarkable success in continuous domains such as images [26, 11, 28] and audio, establishing score-based generative modeling as a dominant paradigm. Extending this success to discrete data—language [16, 10], proteins [1, 4], and molecular graphs [30]—is an active and challenging research direction, since the continuous score function and Langevin dynamics that underpin continuous diffusion have no direct discrete counterpart.

Discrete diffusion models formulate the forward corruption and reverse generation as continuous-time Markov chains (CTMCs) on a finite state space $\mathcal{S} = \{1, \dots, S\}$. D3PM [3] and Multinomial Diffusion [12] initiated this line by introducing discrete transition matrices in discrete time. Campbell et al. [5] extended the framework to continuous time, deriving a variational bound from CTMC path measures. Subsequent work has explored different parameterizations of the reverse rate matrix $R_t^\theta(i, j)$: SEDD [17] learns a concrete score ratio $q_t(j)/q_t(i)$ analogous to continuous score matching [13, 18]; MDLM [24] simplifies training via clean-data prediction (x_0-parameterization) under a

*Equal contribution.
†Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

masked (absorbing) forward process; GIDD [31] extends the x_0-parameterization to unify absorbing, uniform, and interpolating forward processes under a general framework; and concurrent efforts have explored discrete flow matching [6, 8] and simplified masked diffusion [25, 21]. These methods have achieved increasingly strong results on language modeling benchmarks.

Despite this progress, a fundamental structural property of CTMCs has not been directly reflected in the reverse-process parameterization. Any rate matrix admits a unique factorization

$$R_t(i,j) \;=\; \lambda_t(i) \cdot r_t(j \mid i), \qquad \lambda_t(i) = \sum_{j \neq i} R_t(i,j), \quad r_t(j \mid i) = R_t(i,j)/\lambda_t(i),$$

where $\lambda_t(i)$ is the *exit rate* governing *when* to jump and $r_t(\cdot \mid i)$ is the *jump distribution* governing *where* to jump. This exit-rate/jump-distribution decomposition is precisely the basis of Gillespie's algorithm [9, 2] for exact CTMC simulation: first draw a holding time from the exit rate, then choose the next state from the jump distribution. Despite being classical, this structure has not been reflected in the reverse-process parameterization of discrete diffusion models—existing methods still learn the reverse process through proxy quantities such as score ratios, clean-data predictions, or denoising distributions, which implicitly determine R_t^θ as a monolithic object. Inspired by the algorithmic decomposition in Gillespie's algorithm, our starting point is that the training objective itself also decomposes along this timing/direction structure, suggesting a parameterization and training rule that make this separation explicit.

Moreover, the two mainstream forward processes in discrete diffusion are masked (absorbing) and uniform. The masked forward process is dominant in current state-of-the-art methods [24, 31], but its absorbing nature means a token, once unmasked during generation, is frozen for the rest of sampling, limiting diversity and precluding self-correction [31]. The uniform forward process avoids this limitation by allowing transitions among all S states, yet it has received comparatively little attention and generally yields weaker results [e.g., 17, 3, 31], among others. GIDD [31] partially alleviates these gaps by interpolating between masked and uniform forward processes, but its modification lies on the *forward* (noising) side; the reverse process is still parameterized monolithically through a clean-data predictor. In contrast, we keep the forward process simple and instead modify the *reverse* side, decomposing the reverse rate into its intrinsic timing and direction components.

Building on this perspective, we propose **Neural CTMC**, which addresses both gaps from the reverse-process side. First, we separately parameterize the exit rate λ_t^θ and the jump distribution r_t^θ for the reverse process via two network heads. We show that the θ-dependent part of the ELBO is equivalent to a reverse-process KL divergence up to a θ-independent constant, and that this KL naturally decomposes as

$$\mathrm{KL}^{\mathrm{Poi}}(\hat{\lambda} \| \lambda^\theta) \;+\; \hat{\lambda} \cdot \mathrm{KL}^{\mathrm{Cat}}(\hat{r} \| r^\theta),$$

a Poisson KL for jump timing and a categorical KL for jump direction. This decomposition is not introduced ad hoc at the architecture level; rather, it is induced by the CTMC path-space objective itself, which suggests learning timing and direction with separate heads. Notably, when specialized to the masked (absorbing) forward process, our objective recovers the MDLM loss exactly, and offers a new perspective on MDLM: its x_θ predictor is precisely our jump-distribution head r^θ, while the exit rate λ^θ collapses to a schedule-determined constant—showing that the decomposed formulation strictly generalizes existing approaches. **Second, equipped with this decomposed parameterization, Neural CTMC with a pure uniform forward process surpasses several state-of-the-art masked-based methods on the benchmarks we study.**

Contributions.

1. **A reverse-process parameterization aligned with CTMC structure.** We parameterize the reverse CTMC through separate exit-rate and jump-distribution heads, motivated by the intrinsic timing/direction factorization of CTMC dynamics (Sections 2 and 4.1).
2. **A decomposition of the reverse-process objective.** We derive a clean form of the path-space ELBO for CTMC-based discrete diffusion, showing that it reduces to a reverse-process KL that splits into a Poisson KL for exit rates and a categorical KL for jump directions. Similarly, its tractable conditional surrogate preserves gradients and minimizers under standard regularity assumptions (Section 4.1).
3. **Strong empirical performance with a pure uniform forward process.** On TinyStories, Neural CTMC achieves generative perplexity 16.36, compared with 37.60 for GIDD and 42.66 for

MDLM under our evaluation protocol. On OpenWebText, it achieves the lowest mean generative perplexity among the compared 131B-token checkpoints at all five sampling step counts from 16 to 256. We release open-source checkpoints for our pure uniform-noise discrete diffusion model.

2 Background

2.1 Continuous-Time Markov Chains

Let $\mathcal{S} = \{1, \dots, S\}$ be a finite state space. A CTMC on $\mathcal{S}$ over $[0, T]$ [20] is characterized by a rate matrix R_t with $R_t(i, j) \geq 0$ for $i \neq j$ and $\sum_j R_t(i, j) = 0$.

Definition 2.1. For some start time s and end time $t = s + \Delta$ $(t > s)$ as $\Delta \to 0$, we have

$$q_{t|s}(j \mid i) = \delta_{i,j} + R_t(i, j)\,\Delta + o(\Delta), \tag{1}$$

where R_t is called the forward transition rate.

The Kolmogorov forward equation $\frac{\mathrm{d}}{\mathrm{d}t} q_t = q_t R_t$ yields transition probabilities $P_{t|s}$ [5]. For the uniform-rate forward process with $R_t(i, j) = \beta_t$ for $i \neq j$, the cumulative transition probabilities take a closed form.

Definition 2.2. The cumulative transition probabilities of the CTMC are given by

$$q_{t|0}(j \mid i) = \mathrm{Cat}(j;\, P_t e_i), \qquad P_t = \alpha_t I + \beta_t \pi \mathbf{1}^\top, \tag{2}$$

where $\alpha_t + \beta_t = 1$ with $\alpha_0 = 1, \alpha_T = 0$, π is a fixed distribution (e.g., the uniform distribution $\pi = \frac{1}{S}\mathbf{1}$, or a mask distribution $\pi = e_{\texttt{[M]}}$ where $e_{\texttt{[M]}}$ is the one-hot vector for the mask token; π may also depend on t), and $\mathbf{1}$ is the all-ones vector. As $t \to T$, $q_{t|0}(\cdot \mid i) \to \pi$ for all i, so the prior distribution is $p_{\text{prior}} = \pi$.

Path measure. Càdlàg (right-continuous with left limits) paths provide a general framework for stochastic processes with jumps, encompassing Lévy processes, semimartingales, and others. A CTMC corresponds to piecewise constant càdlàg paths and is characterized by a path measure $\mathbb{Q}_{x_0}$. A trajectory W drawn from the CTMC can be fully described through its jump times $T_1, \dots, T_n$ and its state values between jumps $W_0, W_1, \dots, W_n$, where at jump time T_k the CTMC jumps from state W_{k-1} to W_k.

Definition 2.3. A path space measure $\mathbb{Q}_{x_0}$ assigns probabilities to a drawn trajectory W from time 0 to T in the sense of

$$\mathbb{Q}_{x_0}(W \in \mathrm{d}\omega) := \mathbb{P}(W_0 \in \mathrm{d}\omega_0,\, (T_1, W_{T_1}) \in \mathrm{d}(t_1, \omega_1),\, \dots,\, (T_n, W_{T_n}) \in \mathrm{d}(t_n, \omega_n),\, T_{n+1} \geq T)\,,$$

where $\mathrm{d}\omega_k$ and $\mathrm{d}t_k$ denote infinitesimal neighborhoods around the points $\omega_k \in \mathcal{S}$ and $t_k \in [0, T]$.

Writing $x_0, x_1, \dots, x_n$ for the state values and $t_1, \dots, t_n$ for the jump times, with $X_t = x_k$ for $t \in [t_k, t_{k+1})$ (where $t_0 := 0$ and $t_{n+1} := T$), the path measure density takes the following form.

Lemma 2.4. *Let $\mathbb{Q}_{x_0}$ denote the path measure induced by a CTMC with rate matrix R_t starting from initial state x_0. For a path ω with jumps at times $0 \leq t_1 < \cdots < t_n < T$ and states $x_0 \to x_1 \to \cdots \to x_n$:*

$$\mathrm{d}\mathbb{Q}_{x_0}(\omega) = \underbrace{\prod_{k=1}^{n} R_{t_k}(x_{k-1}, x_k)}_{\textit{jump rates}} \cdot \underbrace{\exp\left(-\int_0^T \lambda_t(X_t)\,\mathrm{d}t\right)}_{\textit{survival probability}}, \tag{3}$$

where $\lambda_t(i) := \sum_{j \neq i} R_t(i, j)$ is the total transition rate out of state i.

The proof is given in Appendix C. Equation (3) will serve as the starting point of our approach: we treat λ_t and r_t as the two natural targets of approximation, and learn them with separate network heads λ_t^θ and r_t^θ via the path-measure objective derived in Section 4.1.

2.2 Discrete Diffusion and Existing Parameterizations

A CTMC-based discrete diffusion model consists of a *forward* process and a *reverse* (generative) process. The forward process corrupts data $X_0 \sim p_{\text{data}}$ via a CTMC with rate matrix R_t, yielding conditionals $q_{t|0}(x \mid x_0) := \mathbb{P}(X_t = x \mid X_0 = x_0)$ and marginals $q_t(x) := \mathbb{E}_{x_0 \sim p_{\text{data}}}[q_{t|0}(x \mid x_0)]$, with $q_t \to p_{\text{prior}}$ as $t \to T$. A key structural property of the rate matrix is the following decomposition.

Exit rate and jump distribution. Any rate matrix admits the unique decomposition $R_t(i,j) = \lambda_t(i) \cdot r_t(j \mid i)$, where

$$\lambda_t(i) := \sum_{j \neq i} R_t(i,j) \;\; \text{(exit rate)}, \qquad r_t(j \mid i) := \frac{R_t(i,j)}{\lambda_t(i)} \;\; \text{(jump distribution)}. \tag{4}$$

$\lambda_t(i)$ governs the holding time and $r_t(\cdot \mid i)$ is a categorical distribution over jump destinations. This factorization separates *when* to jump from *where* to jump—a structural property of any CTMC.

Reverse process. The time-reversal of the forward process is a CTMC with rate matrix [5]:

$$\hat{R}_t(i,j) = R_t(j,i)\frac{q_t(j)}{q_t(i)}, \quad i \neq j, \tag{5}$$

A neural network parameterizes a reverse CTMC with rate matrix $R_t^\theta(i,j)$ starting from $X_T \sim p_{\text{prior}}$, with the goal of minimizing an upper bound on the negative log-likelihood $-\mathbb{E}_{x_0 \sim p_{\text{data}}}[\log p_\theta(x_0)]$.

Existing parameterizations. Prior work parameterizes R_t^θ through different proxy quantities, each implicitly determining the full reverse rate matrix:

- **Concrete score** [17]: parameterize $s_\theta(x_t, t)_y \approx \frac{q_t(y)}{q_t(x_t)}$, recover $R_t^\theta(i,j) = R_t(j,i) \cdot s_\theta(i,t)_j$. Trained via score entropy.
- **Denoising distribution** [5]: parameterize $p_\theta(x_0 \mid x_t)$, construct $R_t^\theta(i,j) = \sum_{x_0} p_\theta(x_0 \mid i)\, R_t(j,i)\, \frac{q_{t|0}(j|x_0)}{q_{t|0}(i|x_0)}$.
- **Clean data prediction** (x_0-param.) [3, 24, 31]: the model distribution $p_\theta(z_s \mid z_t) = q_{t|s}(z_t \mid z_s)\, \frac{q_s(z_s|x_\theta)}{q_t(z_t|x_\theta)}$, where $x_\theta(z_t, t)$ is a neural network that predicts the clean data.

All three strategies determine $R_t^\theta(i,j)$ without decomposing it into exit rate and jump distribution—which, as we show, correspond to independent learning objectives.

3 Related Work

D3PM [3] and Multinomial Diffusion [12] introduced discrete diffusion using transition matrices in discrete time. Campbell et al. [5] extended this to continuous time via CTMCs, deriving the ELBO from path measure ratios. Subsequent work explored different parameterizations of the reverse rate matrix: SEDD [17] uses a concrete score analogous to continuous score matching [28, 18], MDLM [24] simplifies masked diffusion with x_0-prediction, GIDD [31] unifies these under a general interpolating framework, and discrete flow matching [6, 8] casts CTMC-based discrete generation in a flow-matching framework with multimodal extensions. Recent concurrent work has also explored simplified masked objectives [25, 21], guided generation in discrete spaces [19], and improved sampling via informed correctors [32]. On the variational side, the ELBO for continuous diffusion was established via SDE path measures and Girsanov's theorem [27, 15]; the discrete analog replaces SDEs with CTMCs and Girsanov's theorem with the Campbell–Mecke formula [14], and Sun et al. [29] derived a score-based continuous-time objective for discrete diffusion along these lines.

4 Methodology

4.1 Theory

We retain the setup from Section 2: $\mathcal{S} = \{1, \ldots, S\}$, forward CTMC with rate matrix R_t corrupting $X_0 \sim p_{\text{data}}$ into $X_T \sim p_{\text{prior}}$, with conditionals $q_{t|0}$ and marginals q_t. The true reverse CTMC has

rate matrix $\hat{R}_t(i,j) = R_t(j,i)\,\frac{q_t(j)}{q_t(i)}$. By the decomposition (4), its dynamics factor as:

$$\hat{\lambda}_t(i) = \sum_{j \neq i} \hat{R}_t(i,j) \;\; \text{(exit rate)}, \qquad \hat{r}_t(j \mid i) = \frac{\hat{R}_t(i,j)}{\hat{\lambda}_t(i)} \;\; \text{(jump distribution)}. \tag{6}$$

Rather than parameterizing a proxy for $R_t^\theta(i,j)$ (Section 2.2), we directly parameterize both components via a neural network $\Phi_\theta : \mathcal{S} \times [0,T] \to \mathbb{R}_{>0} \times \Delta^{S-1}$[3]:

$$\Phi_\theta(x_t, t) = \big(\lambda_t^\theta(x_t),\; r_t^\theta(\cdot \mid x_t)\big), \qquad R_t^\theta(i,j) = \lambda_t^\theta(i) \cdot r_t^\theta(j \mid i) \;\; (j \neq i). \tag{7}$$

Crucially, the variational objective inherits this factorization, decomposing into two subproblems: predicting *when to jump* and *where to jump*. We state the main results here and defer proofs to Appendix C.

Lemma 4.1. *Let $\mathbb{Q}_{x_0}$ be the forward path measure starting from x_0, and $\mathbb{P}^\theta$ the neural reverse path measure. For a path ω with jumps at times $0 < t_1 < \cdots < t_n < T$ and states $x_0 \to x_1 \to \cdots \to x_n$:*

$$\log \frac{\mathrm{d}\mathbb{P}^\theta}{\mathrm{d}\mathbb{Q}_{x_0}}(\omega) = \underbrace{\log p_{prior}(x_n)}_{\text{prior}} + \underbrace{\sum_{k=1}^{n} \log \frac{R_{t_k}^\theta(x_k, x_{k-1})}{R_{t_k}(x_{k-1}, x_k)}}_{\text{jump ratio}} + \underbrace{\int_0^T \Big[\lambda_t^{fwd}(X_t) - \lambda_t^\theta(X_t)\Big]\,\mathrm{d}t}_{\text{survival difference}}. \tag{8}$$

The proof follows from taking the ratio of the forward and reverse path densities; see Appendix C. Combining this ratio with importance sampling under $\mathbb{Q}_{x_0}$ and Jensen's inequality yields the following ELBO.

Theorem 4.2. *For any $x_0 \in \mathcal{S}$:*

$$-\log p_\theta(x_0) \leq -\mathbb{E}_{q_{T|0}}[\log p_{prior}(X_T)] + \int_0^T \sum_{i \in \mathcal{S}} q_{t|0}(i \mid x_0)\, \ell_t(i)\,\mathrm{d}t, \tag{9}$$

where $\ell_t(i) := \sum_{j \neq i} \Big[R_t(i,j) \log \frac{R_t(i,j)}{R_t^\theta(j,i)} - R_t(i,j) + R_t^\theta(i,j)\Big]$.

Theorem 4.2 is a conditional bound in which the loss ℓ_t couples the forward rate R_t with the reverse parameterization R_t^θ. Averaging over $x_0 \sim p_{\text{data}}$ marginalizes the conditional $q_{t|0}(\cdot \mid x_0)$ into q_t, after which the θ-dependent part rewrites cleanly as a reverse-process path-space KL up to a θ-independent constant.

Theorem 4.3.

$$\mathcal{E}_{\text{data}}(\theta) := \mathbb{E}_{x_0 \sim p_{data}}\left[-\mathbb{E}_{q_{T|0}}[\log p_{prior}(X_T)] + \int_0^T \sum_{i \in \mathcal{S}} q_{t|0}(i \mid x_0)\, \ell_t(i)\,\mathrm{d}t\right]. \tag{10}$$

$$\mathbb{E}_{x_0 \sim p_{data}}[-\log p_\theta(x_0)] \leq \mathcal{E}_{\text{data}}(\theta), \tag{11}$$

and

$$\mathcal{E}_{\text{data}}(\theta) = \mathrm{KL}(\hat{\mathbb{Q}} \| \mathbb{P}^\theta) + C, \tag{12}$$

where C is independent of θ, $\hat{\mathbb{Q}}$ denotes the path measure of the true reverse CTMC with rate matrix $\hat{R}_t$, $\mathbb{P}^\theta$ denotes the path measure of the parameterized reverse CTMC with rate matrix R_t^θ, and

$$\mathrm{KL}(\hat{\mathbb{Q}} \| \mathbb{P}^\theta) = \int_0^T \sum_{i \in \mathcal{S}} q_t(i) \sum_{j \neq i} \left[\hat{R}_t(i,j) \log \frac{\hat{R}_t(i,j)}{R_t^\theta(i,j)} - \hat{R}_t(i,j) + R_t^\theta(i,j)\right] \mathrm{d}t. \tag{13}$$

Since both rates in (13) now refer to the same reverse direction, the objective admits a timing-direction decomposition.

[3] $\Delta^{S-1} = \{p \in \mathbb{R}^S_{\geq 0} : \sum_j p_j = 1\}$ denotes the $(S-1)$-dimensional probability simplex.

Corollary 4.4. *The reverse process KL decomposes as:*

$$\mathrm{KL}(\hat{\mathbb{Q}}\|\mathbb{P}^\theta) = \int_0^T \sum_{i\in\mathcal{S}} q_t(i) \left[\underbrace{\mathrm{KL}^{Poi}\big(\hat{\lambda}_t(i)\|\lambda_t^\theta(i)\big)}_{\text{when to jump}} + \hat{\lambda}_t(i) \cdot \underbrace{\mathrm{KL}^{Cat}\big(\hat{r}_t(\cdot \mid i)\|r_t^\theta(\cdot \mid i)\big)}_{\text{where to jump}} \right] \mathrm{d}t, \quad (14)$$

where $\mathrm{KL}^{Poi}(\lambda\|\lambda^\theta) := \mathrm{KL}\big(Poi(\lambda)\,\|\,Poi(\lambda^\theta)\big) = \lambda\log\frac{\lambda}{\lambda^\theta} - \lambda + \lambda^\theta$ *is the Poisson KL and* $\mathrm{KL}^{Cat}(\hat{r}\|r^\theta) := \sum_{j\neq i} \hat{r}(j \mid i)\log\frac{\hat{r}(j|i)}{r^\theta(j|i)}$ *is the categorical KL.*

The exit-rate head is trained by a Poisson KL and the jump-distribution head by a categorical KL weighted by $\hat{\lambda}_t(i)$, separating *whether to move* from *where to move next*.

Proposition 4.5. *For fixed t and i, define*

$$g_{t,i}(\lambda, r) := \mathrm{KL}^{Poi}\big(\hat{\lambda}_t(i)\|\lambda\big) + \hat{\lambda}_t(i)\,\mathrm{KL}^{Cat}\big(\hat{r}_t(\cdot \mid i)\|r\big). \quad (15)$$

If $\hat{\lambda}_t(i) > 0$, then $g_{t,i}(\lambda, r) \geq 0$ with equality if and only if $\lambda = \hat{\lambda}_t(i)$ and $r = \hat{r}_t(\cdot \mid i)$. If $\hat{\lambda}_t(i) = 0$, the unique minimizer in λ is $\lambda = 0$ and the choice of r is irrelevant. The training objective (14) *can be estimated via Monte Carlo sampling. Using the cross-entropy identity $\mathrm{KL}^{Cat} = \mathrm{CE} - H$:*

$$\mathcal{L}(\theta) = \mathbb{E}_{t\sim\mathcal{U}(0,T),\, x_t\sim q_t}\Big[\mathrm{KL}^{Poi}\big(\hat{\lambda}_t(x_t)\|\lambda_t^\theta(x_t)\big) + \hat{\lambda}_t(x_t)\cdot\mathrm{CE}\big(\hat{r}_t(\cdot \mid x_t),\, r_t^\theta(\cdot \mid x_t)\big)\Big] + \mathrm{const}, \quad (16)$$

where $\mathrm{CE}(\hat{r}, r^\theta) := -\sum_{j\neq i} \hat{r}(j \mid i)\log r^\theta(j \mid i)$ *is the cross-entropy, and the constant absorbs the entropy of $\hat{r}_t$ (independent of θ).*

Since $q_t(x) = \mathbb{E}_{x_0}[q_{t|0}(x \mid x_0)]$ is intractable, we replace marginal quantities by conditional counterparts via the expectation transformation (Lemma C.7 in Appendix C), yielding the conditional formulation.

Proposition 4.6. *For a fixed data point $x_0 \in \mathcal{S}$, define the conditional reverse rates:*

$$\hat{\lambda}_{t|0}(i \mid x_0) := \sum_{j\neq i} R_t(j,i)\frac{q_{t|0}(j \mid x_0)}{q_{t|0}(i \mid x_0)}, \qquad \hat{r}_{t|0}(j \mid i, x_0) := \frac{R_t(j,i)\frac{q_{t|0}(j|x_0)}{q_{t|0}(i|x_0)}}{\hat{\lambda}_{t|0}(i \mid x_0)}. \quad (17)$$

Then the training loss (16) *can be equivalently written as:*

$$\begin{aligned}\mathcal{L}(\theta) = \mathbb{E}_{\substack{t\sim\mathcal{U}(0,T),\, x_0\sim p_{data}\\ x_t\sim q_{t|0}(\cdot|x_0)}}\Big[&-\sum_{j\neq x_t} R_t(j,x_t)\frac{q_{t|0}(j \mid x_0)}{q_{t|0}(x_t \mid x_0)}\log\big(\lambda_t^\theta(x_t)\, r_t^\theta(j \mid x_t)\big)\\ &- \hat{\lambda}_{t|0}(x_t \mid x_0) + \lambda_t^\theta(x_t)\Big] + \mathrm{const}.\end{aligned} \quad (18)$$

The proof is given in Appendix C.

Theorem 4.7. *Define the* conditional reverse rate *given clean data x_0:*

$$\hat{R}_t(i, j \mid x_0) := R_t(j,i)\,\frac{q_{t|0}(j \mid x_0)}{q_{t|0}(i \mid x_0)}, \quad i \neq j, \quad (19)$$

and the Poisson KL density $f(r,c) := r\log\frac{r}{c} - r + c$. The conditional KL training loss

$$\mathcal{L}_{\mathrm{KL}}(\theta) := \int_0^T \mathbb{E}_{x_0\sim p_{data},\, x_t\sim q_{t|0}(\cdot|x_0)}\left[\sum_{j\neq x_t} f\Big(\hat{R}_t(x_t, j \mid x_0),\; R_t^\theta(x_t, j)\Big)\right]\mathrm{d}t \quad (20)$$

is a tractable upper bound on $\mathrm{KL}(\hat{\mathbb{Q}}\|\mathbb{P}^\theta)$*:*

$$\mathbb{E}_{p_{data}}[-\log p_\theta(x_0)] \leq \mathcal{L}_{\mathrm{KL}}(\theta) + C_{\mathrm{total}}, \quad (21)$$

where C_{total} is independent of θ. Moreover, $\mathcal{L}_{\mathrm{KL}}(\theta)$ equals the loss in (18) *up to a θ-independent constant.*

Theorem 4.7 establishes a tractable objective value upper bound. The next proposition strengthens this by showing that, under standard regularity assumptions, the conditional surrogate differs from the marginal reverse-process KL only by a θ-independent gap.

Proposition 4.8. *Assume: (i) forward quantities (q_t, $q_{t|0}$, $p(x_0 \mid x_t)$) are independent of θ; (ii) $R_t^\theta(i,j) > 0$ and differentiable in θ for all $i \neq j$; and (iii) differentiation can be exchanged with expectation/integration. Then there exists a θ-independent constant $C_{\mathrm{gap}} \geq 0$ such that*

$$\mathcal{L}_{\mathrm{KL}}(\theta) = \mathrm{KL}(\hat{\mathbb{Q}} \| \mathbb{P}^\theta) + C_{\mathrm{gap}},$$

hence

$$\nabla_\theta \mathcal{L}_{\mathrm{KL}}(\theta) = \nabla_\theta \mathrm{KL}(\hat{\mathbb{Q}} \| \mathbb{P}^\theta).$$

In particular,

$$\arg\min_\theta \mathcal{L}_{\mathrm{KL}}(\theta) = \arg\min_\theta \mathrm{KL}(\hat{\mathbb{Q}} \| \mathbb{P}^\theta),$$

and every stationary point of one objective is a stationary point of the other.

Further discussion of the upper-bound interpretation and the information-theoretic form of C_{gap} is deferred to Appendix C.

Corollary 4.9. *Defining $K(a) := a(\log a - 1)$, the losses* (18) *and* (20) *are both equivalent to:*

$$\begin{aligned}\mathcal{L}_{\mathrm{cond}}(\theta) = \mathbb{E}_{t,\, x_0,\, i}\Big[\sum_{j \neq i} \lambda_t^\theta(i) r_t^\theta(j \mid i) - \sum_{j \neq i} R_t(j,i) \frac{q_{t|0}(j \mid x_0)}{q_{t|0}(i \mid x_0)} \log \frac{\lambda_t^\theta(i)\, r_t^\theta(j \mid i)}{R_t(j,i)} \\ + \sum_{j \neq i} R_t(j,i)\, K\Big(\frac{q_{t|0}(j \mid x_0)}{q_{t|0}(i \mid x_0)} \Big) \Big].\end{aligned} \tag{22}$$

For each pair $i \neq j$, the three terms together form the nonnegative Poisson KL density $f\Big(\hat{R}_t(i,j \mid x_0), R_t^\theta(i,j)\Big)$. The first two terms contain all the dependence on θ, while the last term is independent of θ.

Through the above derivations, we obtain two equivalent training losses: $\mathcal{L}_{\mathrm{KL}}(\theta)$ (20) and $\mathcal{L}_{\mathrm{cond}}(\theta)$ (22). Under the assumptions stated above, they differ only by a θ-independent constant and therefore induce the same first-order optimization problem.

Corollary 4.10 (Equivalence to MDLM)**.** *If the forward process is the masked (absorbing) process on $\mathcal{S} = \mathcal{V} \cup \{m\}$ with rate $R_t(i,j) = -\frac{\alpha_t'}{\alpha_t}(\boldsymbol{e}_j^\top \boldsymbol{e}_m)\mathbf{1}_{\{i \neq m\}}$, then for any valid noise schedule α_t, the loss* (22) *reduces exactly to the MDLM cross-entropy loss:*

$$\mathcal{L}_{\mathrm{cond}}(\theta) = \mathbb{E}_{t,\, x_0,\, x_t}\left[\frac{\alpha_t'}{1 - \alpha_t}\, \boldsymbol{x}_0^\top \log x_\theta \cdot \mathbf{1}_{\{x_t = m\}} \right], \tag{23}$$

The proof is given in Appendix C. This shows that MDLM is a special case of Neural CTMC's objective restricted to the masked forward process; the uniform forward process used in our experiments is a strict generalization. **In the masked case, r_t^θ reduces to the x_θ predictor in MDLM, while $\lambda_t^\theta(m) = -\frac{\alpha_t'}{1-\alpha_t}$ becomes a constant determined by the noise schedule.**

4.2 Algorithm

Training algorithm. Following the derivation in Section 4.1, we form the conditional reverse targets $\hat{R}_t(x_t,\, j \mid x_0)$ and $\hat{\lambda}_{t|0}(x_t \mid x_0)$ from the forward conditionals (2) and minimize either $\mathcal{L}_{\mathrm{KL}}$ (20) or $\mathcal{L}_{\mathrm{cond}}$ (22); the full procedure is summarized in Algorithm 1.

Algorithm 1 Neural CTMC Training

1: **while** not converged **do**
2: Sample $x_0 \sim p_{\mathrm{data}}$, $t \sim \mathcal{U}(\epsilon, T - \epsilon)$; form $q_{t|0}(\cdot \mid x_0)$ via (2)
3: Sample $x_t \sim q_{t|0}(\cdot \mid x_0)$ and form the conditional targets $\hat{R}_t(x_t, j \mid x_0)$ and $\hat{\lambda}_{t|0}(x_t \mid x_0)$
4: Compute $\big(\lambda_t^\theta(x_t),\, r_t^\theta(\cdot \mid x_t)\big) = \Phi_\theta(x_t, t)$, evaluate $\mathcal{L}_{\mathrm{KL}}(\theta)$ (20) or $\mathcal{L}_{\mathrm{cond}}(\theta)$ (22), and update $\theta \leftarrow \theta - \eta\, \nabla_\theta \mathcal{L}$
5: **end while**

Sampling algorithms. Given a trained model Φ_θ, we generate samples by discretizing the reverse CTMC over $[0, T]$ with step size $\tau = T/N$. We consider two sampling schemes. **τ-Leaping** [9] (Algorithm 2) applies the jump-chain/holding-time view in parallel across sequence positions: each position draws a holding time from its learned exit rate and, if the holding time falls within the current step, jumps according to its learned jump distribution. This naturally exploits the exit-rate/jump-direction decomposition and allows multiple positions to change in a single simulation step. **Euler** (Algorithm 3) constructs a one-step categorical transition from the off-diagonal probabilities $p_j = \lambda_t^\theta(x_t) \cdot r_t^\theta(j \mid x_t) \cdot \tau$ for $j \neq x_t$, with the remaining mass assigned to staying at x_t. Pseudocode for both samplers is given in Appendix D.

5 Experiments

We evaluate Neural CTMC on discrete image and language generation tasks. Neural CTMC uses the uniform forward process (Section 4.2) with $\alpha_t = 1 - t$ and $\beta_t = t$ and a DiT [22] backbone; full hyperparameters are given in Appendix D.

Image Generation: We train Neural CTMC on the MNIST dataset, flattened to sequences of length 28×28 over $\mathcal{S} = \{0, 1, \ldots, 255\}$. Using the same DiT backbone, Figure 3 shows 100 conditional samples (10 per digit class) generated with 128 sampling steps at epoch 80. Both Euler and τ-leaping produce clearly recognizable digits, confirming that the decoupled parameterization learns meaningful structure even on image data treated as a discrete sequence.

Language Modeling: We compare Neural CTMC against SEDD [17], MDLM [24], and GIDD [31] on TinyStories [7] and OpenWebText [23] (OWT). Neural CTMC uses a 12-layer DiT-style Transformer with hidden dimension 768, 12 heads, max_len=512, and approximately 163M parameters. For OpenWebText, Table 1 also includes DUO and records the training-token budget and maximum sequence length of each checkpoint. For GIDD we report results with $p_{\text{unif}} \in \{0.0, 0.1, 0.2\}$, where $p_{\text{unif}} = 0.0$ corresponds to a pure mask process. On OpenWebText, we report mean generative perplexity (PPL) $\pm$ sample standard deviation over five sampling seeds. Every model generates 1,024 unconditional samples of 512 tokens per seed, scored by Gemma 9B and GPT-2 Large. The Gemma 9B results are given in Table 1; the GPT-2 Large results on the same samples are provided in Appendix D, Table 2.

TinyStories. For a fair comparison, we train all methods from their official source code using the same backbone architecture and training budget. We use 50 sampling steps for every method and score all generated samples with the same pretrained Gemma2-9B model. Figure 1 plots generative PPL over the first 45 training epochs. Neural CTMC converges faster and reaches substantially lower perplexity than the baselines. Neural CTMC attains a best PPL of 16.36, compared with 37.60 for GIDD and 42.66 for MDLM.

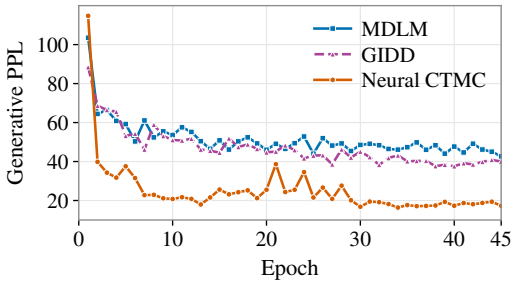


Figure 1: Generative PPL on TinyStories with 50 sampling steps.

OpenWebText. To ensure a fair comparison under uniform forward corruption, we train Neural CTMC, SEDD (uniform), and GIDD (uniform) using the same architecture, training-token budget, and training hyperparameters. All three models use 131B training tokens and a maximum sequence length of 512, and are evaluated under the same protocol. Among all 131B-token checkpoints in Table 1, Neural CTMC achieves the lowest mean generative PPL at all five sampling step counts from 16 to 256, outperforming both the uniform baselines and the masked and mixture GIDD variants. At 32 steps, Neural CTMC achieves a PPL of 265.2, compared with 361.8 for SEDD (uniform) and 483.5 for GIDD (uniform); at 128 steps, the corresponding values are 189.2, 207.3, and 232.1. At 16 and 32 steps, Neural CTMC also outperforms larger-budget masked models: its Gemma 9B PPL is respectively 15.1% and 7.7% lower than MDLM, which uses twice as many training tokens, and 5.5% and 3.8% lower than SEDD (mask), which uses approximately 5.2 times as many. These results demonstrate the effectiveness of Neural CTMC with a pure uniform forward process, including under limited training and sampling budgets.

Table 1: Gemma 9B generative PPL on OpenWebText (mean ± std, lower is better) for varying numbers of sampling steps. **Bold**: best among checkpoints with Train Toks 131B. †: overall best.

Type	Method	Train Toks	Max Len	16	32	64	128	256
mask	SEDD [17]	682B	1024	630.793 ±7.926	275.590 ±4.928	158.236† ±3.333	107.719† ±1.001	79.246 ±0.966
mask	MDLM [24]	262B	1024	702.586 ±12.028	287.290 ±3.186	160.357 ±2.475	107.756 ±1.721	77.767† ±1.526
mask	GIDD [31] ($p_{\text{unif}}=0.0$)	131B	512	2980.247 ±98.176	1036.871 ±33.130	564.206 ±23.818	421.737 ±13.676	368.391 ±11.134
mixture	GIDD [31] ($p_{\text{unif}}=0.1$)	131B	512	708.316 ±14.833	399.776 ±5.931	288.078 ±7.464	251.572 ±7.899	230.243 ±2.287
mixture	GIDD [31] ($p_{\text{unif}}=0.2$)	131B	512	763.013 ±21.389	445.039 ±5.149	346.500 ±8.506	308.073 ±5.285	287.921 ±6.380
continuous	DUO	524B	1024	272.043† ±5.837	197.862† ±1.812	171.522 ±2.292	156.682 ±3.371	150.533 ±3.075
uniform	SEDD [17]	131B	512	942.300 ±43.081	361.819 ±8.608	246.701 ±6.430	207.270 ±0.598	197.105 ±2.578
uniform	GIDD [31]	131B	512	2194.880 ±123.065	483.540 ±12.116	279.374 ±8.078	232.096 ±6.802	209.727 ±2.702
uniform	Neural CTMC (ours)	131B	512	**596.250 ±18.507**	**265.181 ±6.327**	**200.394 ±3.306**	**189.205 ±3.541**	**194.956 ±4.384**

Scaling law. To assess whether the gains of Neural CTMC persist as compute grows, we conduct a scaling study on OpenWebText along three axes—model size {tiny (80.1M), small (169.7M), base (424.5M)}, token budget {100M, 300M, 1B}, and total training FLOPs—with three seeds per configuration. As shown in Figure 2, validation loss decreases monotonically with both parameter count and training tokens, and a clean compute-optimal frontier emerges, with the best run (`base` at 1B tokens) reaching 4.150. The fitted power-law exponent on the frontier ($\alpha \approx 0.16$, $R^2 \approx 0.93$) is in line with mature autoregressive and diffusion-based language models, suggesting that the proposed parameterization scales gracefully and that further gains can be expected at larger compute budgets; see Appendix D for details.

Self-correction. A key advantage of the uniform forward process is that, unlike masked diffusion, generated tokens remain editable during sampling, enabling lightweight post-hoc self-correction. We compare a self-correction (Algorithm 4) against the GIDD self-correction baselines on identical raw OpenWebText samples, with outputs scored by GPT-5.2 on five quality axes. As shown in Table 5, Neural CTMC self-correction at temperature $\tau_{\text{sc}} = 0.1$ achieves the best Grammaticality ($+12.4\%$), Factuality ($+6.2\%$), and Creativity ($+8.0\%$) among all variants, while GIDD ($p_u = 0.2$) leads on Clarity and Writing style; see Appendix E for the full protocol.

6 Conclusion

We present Neural CTMC, a discrete diffusion framework that parameterizes the reverse CTMC through separate exit-rate and jump-distribution heads, aligning the model with the intrinsic timing/direction structure of CTMC dynamics. The resulting ELBO factorizes into a Poisson KL for timing and a categorical KL for direction, and admits a tractable conditional loss that is gradient-equivalent under standard regularity assumptions. Empirically, this decomposition makes the pure uniform forward process competitive on the benchmarks we study: on TinyStories, Neural CTMC reaches generative perplexity 16.36, outperforming the baselines under the same protocol; on OpenWebText, it attains the lowest mean PPL among the compared 131B-token checkpoints at all five sampling step counts from 16 to 256. A scaling study further shows that validation loss decreases monotonically with both model size and tokens, with a clean power-law frontier ($\alpha \approx 0.16$, $R^2 \approx 0.93$). Finally, the editable nature of the uniform process enables a lightweight self-correction step that achieves the best Grammaticality, Factuality, and Creativity scores among all variants we evaluate. The exit-rate/jump-distribution view, together with our open-source uniform-noise checkpoints, offers the community a principled and reusable foundation for future work on scaling, post-training, and controllable generation in discrete diffusion.

7 Limitations

Our results have several limitations, most stemming from compute constraints rather than the framework itself. *(i) Scale.* The largest model we train is 424.5M parameters with a 512-token context; whether the gains persist at billion-parameter scale and longer contexts is left to future work, though the trend in Appendix D (Figure 2) is encouraging. *(ii) Post-training and RL.* We only consider pre-training and do not study instruction tuning, preference optimization, or RL-style fine-tuning on top of the learned exit-rate / jump-direction model. *(iii) Forward processes.* The framework supports general mixture forward processes, but we only train and evaluate the pure uniform process.

Acknowledgments and Disclosure of Funding

Funding. This work received no external funding.

Competing interests. The authors declare no competing interests.

References

[1] Sarah Alamdari, Nitya Thakkar, Rianne van den Berg, Alex X. Lu, Nicolo Fusi, Ava P. Amini, and Kevin K. Yang. Protein generation with evolutionary diffusion: Sequence is all you need. *bioRxiv*, 2023.

[2] David F. Anderson. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. *The Journal of Chemical Physics*, 127(21), 2007.

[3] 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.

[4] Pavel Avdeyev, Chenlai Shi, Yuhao Tan, Kseniia Dudnyk, and Jian Zhou. Dirichlet diffusion score model for biological sequence generation. In *International Conference on Machine Learning*, 2023.

[5] Andrew Campbell, Joe Benton, Valentin De Bortoli, Tom Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. *NeurIPS*, 2022.

[6] Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In *International Conference on Machine Learning*, 2024.

[7] Ronen Eldan and Yuanzhi Li. Tinystories: How small can language models be and still speak coherent english? *arXiv preprint arXiv:2305.07759*, 2023.

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

[9] Daniel T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, 1977.

[10] Ishaan Gulrajani and Tatsunori B. Hashimoto. Likelihood-based diffusion language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

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

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

[13] Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6:695–709, 2005.

[14] Martin Jacobsen. *Point Process Theory and Applications: Marked Point and Piecewise Deterministic Processes*. Birkhäuser, 2006.

[15] Diederik P Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in Neural Information Processing Systems*, 34:21696–21707, 2021.

[16] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. Diffusion-LM improves controllable text generation. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[17] Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *International Conference on Machine Learning*, 2024.

[18] Chenlin Meng, Kristy Choi, Jiaming Song, and Stefano Ermon. Concrete score matching: Generalized score matching for discrete data. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[19] Hunter Nisonoff, Junhao Xiong, Stephan Allenspach, and Jennifer Listgarten. Unlocking guidance for discrete state-space diffusion and flow models. In *International Conference on Learning Representations*, 2025.

[20] James R. Norris. *Markov Chains*. Cambridge University Press, 1997.

[21] 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 *International Conference on Learning Representations*, 2025.

[22] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *International Conference on Computer Vision*, 2023.

[23] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 2019.

[24] Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander M. Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

[25] Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

[26] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265, 2015.

[27] Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. Maximum likelihood training of score-based diffusion models. In *Advances in Neural Information Processing Systems*, 2021.

[28] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[29] Haoran Sun, Lijun Yu, Bo Dai, Dale Schuurmans, and Hanjun Dai. Score-based continuous-time discrete diffusion models. In *International Conference on Learning Representations*, 2023.

[30] Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. DiGress: Discrete denoising diffusion for graph generation. In *International Conference on Learning Representations*, 2023.

[31] Dimitri von Rütte, Janis Fluri, Yuhui Ding, Antonio Orvieto, Bernhard Schölkopf, and Thomas Hofmann. Generalized interpolating discrete diffusion. In *International Conference on Machine Learning*, 2025.

[32] Yixiu Zhao, Jiaxin Shi, Feng Chen, Shaul Druckmann, Lester Mackey, and Scott Linderman. Informed correctors for discrete diffusion models. In *Advances in Neural Information Processing Systems*, 2025.

Appendix

Contents

A Mathematical Preliminaries **12**

A.1 Continuous-Time Markov Chains 12

A.2 Point Process 13

B Formal Problem Setup **13**

B.1 Forward Process 13

B.2 True Reverse Process 13

B.3 Neural Reverse Process 14

B.4 Training Objective 14

C Proofs **14**

D Additional Experimental Details **24**

E Self-correction **26**

F Generated Samples **29**

F.1 Image Generation 30

F.2 TinyStories 30

F.3 OpenWebText 32

A Mathematical Preliminaries

We collect here the full definitions of continuous-time Markov chains and point processes used throughout the paper.

A.1 Continuous-Time Markov Chains

Definition A.1. Let $\mathcal{S} = \{1, \dots, S\}$ be a finite state space and $T > 0$ a time horizon. A *continuous-time Markov chain* (CTMC) on $\mathcal{S}$ over $[0, T]$ is a stochastic process $\{X_t\}_{t\in[0,T]}$ with the Markov property:

$$\mathbb{P}(X_{t+s} = j \mid X_t = i, \{X_u\}_{u<t}) = \mathbb{P}(X_{t+s} = j \mid X_t = i). \tag{24}$$

Definition A.2. A CTMC is characterized by a *rate matrix* $R_t : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ satisfying: (i) $R_t(i, j) \geq 0$ for all $i \neq j$, and (ii) $\sum_{j\in\mathcal{S}} R_t(i, j) = 0$ for all $i \in \mathcal{S}$. The off-diagonal entry $R_t(i, j)$ for $i \neq j$ represents the *transition rate* from state i to state j at time t.

Definition A.3. For state $i \in \mathcal{S}$ at time t:

$$\lambda_t(i) := \sum_{j\neq i} R_t(i, j) \quad \text{(exit rate)}, \qquad r_t(j \mid i) := \frac{R_t(i, j)}{\lambda_t(i)} \quad \text{for } j \neq i \quad \text{(jump distribution)}. \tag{25}$$

The exit rate $\lambda_t(i)$ controls the holding time in state i (exponentially distributed), and $r_t(\cdot \mid i)$ is a probability distribution over the $S - 1$ possible jump destinations.

Theorem A.4. *Let $q_t(i) := \mathbb{P}(X_t = i)$ be the marginal distribution at time t. Then q_t satisfies the master equation:*

$$\frac{\mathrm{d}q_t(i)}{\mathrm{d}t} = \sum_{j \in \mathcal{S}} q_t(j) R_t(j,i) = \sum_{j \neq i} q_t(j) R_t(j,i) - q_t(i)\lambda_t(i). \tag{26}$$

Definition A.5. Given a CTMC with rate matrix R_t and marginal distribution $q_t(i)$ at time t, the *time-reversed* CTMC has rate matrix $\hat{R}_t$ given by:

$$\hat{R}_t(i,j) = R_t(j,i)\frac{q_t(j)}{q_t(i)}, \quad i \neq j. \tag{27}$$

A.2 Point Process

Definition A.6. A *point process* on $[0,T]$ is a random collection of points (jump times) $\{t_1, t_2, \ldots\}$ in $[0,T]$. For a CTMC, the jump times form a point process whose intensity at time t is $\lambda_t(X_t)$.

Theorem A.7. *Let $\mathbb{Q}_{x_0}$ be the path measure of a CTMC starting from x_0 with rate matrix R_t. For any measurable function $f : [0,T] \times \mathcal{S} \times \mathcal{S} \to \mathbb{R}$, the expected sum over all jumps satisfies:*

$$\mathbb{E}_{\mathbb{Q}_{x_0}}\left[\sum_{k=1}^{N} f(t_k, X_{t_k^-}, X_{t_k})\right] = \int_0^T \sum_{i \in \mathcal{S}} q_{t|0}(i \mid x_0) \sum_{j \neq i} R_t(i,j) f(t,i,j)\,\mathrm{d}t, \tag{28}$$

where N is the (random) number of jumps, $X_{t_k^-}$ is the state immediately before jump k, X_{t_k} is the state after jump k, and $q_{t|0}(i \mid x_0) := \mathbb{P}_{\mathbb{Q}_{x_0}}(X_t = i)$ is the marginal distribution.

B Formal Problem Setup

We establish here the formal mathematical framework for CTMC-based discrete diffusion models, providing explicit definitions for the objects introduced informally in the main text.

B.1 Forward Process

Definition B.1. The *forward* or *noising* process is a time-dependent CTMC on $\mathcal{S}$ with:

- **Initial distribution**: $X_0 \sim p_{\text{data}}$, the data distribution.
- **Rate matrix**: $R_t : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$.
- **Transition rate**: $R_t(i,j) \geq 0$ for $i \neq j$.
- **Exit rate**: $\lambda_t^{\text{fwd}}(i) := \sum_{j \neq i} R_t(i,j)$.
- **Jump distribution**: $r_t(j \mid i) := \frac{R_t(i,j)}{\lambda_t^{\text{fwd}}(i)}$ for $j \neq i$.

Definition B.2. For the forward process:

$$\begin{aligned} q_{t|0}(x \mid x_0) &:= \mathbb{P}(X_t = x \mid X_0 = x_0), \quad \text{(conditional distribution)} \\ q_t(x) &:= \mathbb{E}_{x_0 \sim p_{\text{data}}}[q_{t|0}(x \mid x_0)], \quad \text{(marginal distribution)}. \end{aligned} \tag{29}$$

B.2 True Reverse Process

Definition B.3. The *true reverse* process is the time-reversal of the forward process, characterized by the rate matrix:

$$\hat{R}_t(i,j) = R_t(j,i)\frac{q_t(j)}{q_t(i)}, \quad i \neq j. \tag{30}$$

This is a CTMC evolving from time 0 to T with:

$$\begin{aligned} \hat{\lambda}_t(i) &:= \sum_{j \neq i} \hat{R}_t(i,j) \quad \text{(true reverse exit rate)}, \\ \hat{r}_t(j \mid i) &:= \frac{\hat{R}_t(i,j)}{\hat{\lambda}_t(i)} \quad \text{(true reverse jump distribution)}. \end{aligned} \tag{31}$$

B.3 Neural Reverse Process

Definition B.4. The *neural reverse process* or *generative model* is a parameterized CTMC with:

- **Initial distribution**: $X_T \sim p_{\text{prior}}$ (reference distribution).
- **Rate matrix**: $R_t^\theta(i,j)$ parameterized by $\theta \in \Theta$.
- **Neural exit rate**: $\lambda_t^\theta(i) := \sum_{j \neq i} R_t^\theta(i,j)$.
- **Neural jump distribution**: $r_t^\theta(j \mid i) := \frac{R_t^\theta(i,j)}{\lambda_t^\theta(i)}$.

Definition B.5. Typically, a neural network $\Phi_\theta : \mathcal{S} \times [0,T] \to \mathbb{R}_{>0} \times \Delta^{S-1}$ outputs:

$$\Phi_\theta(x_t, t) = (\lambda_t^\theta(x_t), r_t^\theta(\cdot \mid x_t)), \tag{32}$$

where Δ^{S-1} is the $(S-1)$-dimensional probability simplex (excluding self-transitions).

B.4 Training Objective

Definition B.6. The goal is to maximize the expected log-likelihood over the data distribution:

$$\max_{\theta \in \Theta} \mathbb{E}_{x_0 \sim p_{\text{data}}}[\log p_\theta(x_0)]. \tag{33}$$

Equivalently, minimize the negative log-likelihood:

$$\min_{\theta \in \Theta} \mathbb{E}_{x_0 \sim p_{\text{data}}}[-\log p_\theta(x_0)]. \tag{34}$$

Remark B.7 (Intractability). Direct evaluation of $p_\theta(x_0)$ requires marginalizing over all possible paths in continuous time, which is computationally intractable. This motivates the ELBO approach developed in Appendix C.

C Proofs

Lemma C.1. *Let $\mathbb{Q}_{x_0}$ denote the path measure induced by a CTMC with rate matrix R_t starting from initial state x_0. For a path ω with jumps at times $0 \le t_1 < \cdots < t_n < T$ and states $x_0 \to x_1 \to \cdots \to x_n$:*

$$\mathrm{d}\mathbb{Q}_{x_0}(\omega) = \left[\prod_{k=1}^{n} R_{t_k}(x_{k-1}, x_k)\right] \exp\left(-\int_0^T \lambda_t(X_t)\,\mathrm{d}t\right). \tag{35}$$

Proof. We decompose the path probability into a product of conditional distributions following the CTMC dynamics. A trajectory starting from x_0 is fully described by jump times $0 \le t_1 < \cdots < t_n < T$ and states $x_0 \to x_1 \to \cdots \to x_n$, where at jump time t_k the chain transitions from x_{k-1} to x_k, and $X_t = x_k$ for $t \in [t_k, t_{k+1})$ (with $t_0 := 0$ and $t_{n+1} := T$).

Step 1: Holding time distribution. When the CTMC is in state x_{k-1} starting from time t_{k-1}, the waiting time until the next jump is governed by a time-inhomogeneous exponential distribution. The cumulative distribution function for the next jump time t_k is:

$$\mathbb{P}(T_k < t \mid T_{k-1} = t_{k-1}) = 1 - \exp\left(-\int_{t_{k-1}}^{t} \lambda_s(x_{k-1})\,\mathrm{d}s\right). \tag{36}$$

Differentiating, the density of the k-th jump time is:

$$p(t_k \mid T_{k-1} = t_{k-1}) = \lambda_{t_k}(x_{k-1}) \exp\left(-\int_{t_{k-1}}^{t_k} \lambda_s(x_{k-1})\,\mathrm{d}s\right). \tag{37}$$

Step 2: Jump destination distribution. Conditional on a jump occurring at time t_k from state x_{k-1}, the destination x_k is chosen according to:

$$\mathbb{P}(X_{t_k} = x_k \mid X_{t_k^-} = x_{k-1}) = r_{t_k}(x_k \mid x_{k-1}) = \frac{R_{t_k}(x_{k-1}, x_k)}{\lambda_{t_k}(x_{k-1})}. \tag{38}$$

Step 3: Combining the contributions. The joint contribution of the k-th jump (both timing and destination) is:

$$\begin{aligned} & p(t_k \mid T_{k-1} = t_{k-1}) \cdot \mathbb{P}(X_{t_k} = x_k \mid X_{t_k^-} = x_{k-1}) \\ &= \lambda_{t_k}(x_{k-1}) \exp\left(- \int_{t_{k-1}}^{t_k} \lambda_s(x_{k-1}) \, \mathrm{d}s \right) \cdot \frac{R_{t_k}(x_{k-1}, x_k)}{\lambda_{t_k}(x_{k-1})} \\ &= R_{t_k}(x_{k-1}, x_k) \exp\left(- \int_{t_{k-1}}^{t_k} \lambda_s(x_{k-1}) \, \mathrm{d}s \right). \end{aligned} \tag{39}$$

The survival probability after the last jump (no further jump in $[t_n, T]$) is:

$$\mathbb{P}(T_{n+1} \geq T \mid T_n = t_n) = \exp\left(- \int_{t_n}^{T} \lambda_s(x_n) \, \mathrm{d}s \right). \tag{40}$$

Step 4: Full path density. Taking the product over all n jumps and the final survival term:

$$\mathrm{d}\mathbb{Q}_{x_0}(\omega) = \prod_{k=1}^{n} R_{t_k}(x_{k-1}, x_k) \cdot \prod_{k=1}^{n} \exp\left(- \int_{t_{k-1}}^{t_k} \lambda_s(x_{k-1}) \, \mathrm{d}s \right) \cdot \exp\left(- \int_{t_n}^{T} \lambda_s(x_n) \, \mathrm{d}s \right). \tag{41}$$

Since $X_t = x_{k-1}$ for $t \in [t_{k-1}, t_k)$, the exponential terms combine into a single integral over the entire time horizon:

$$\prod_{k=1}^{n} \exp\left(- \int_{t_{k-1}}^{t_k} \lambda_s(x_{k-1}) \, \mathrm{d}s \right) \cdot \exp\left(- \int_{t_n}^{T} \lambda_s(x_n) \, \mathrm{d}s \right) = \exp\left(- \int_0^T \lambda_t(X_t) \, \mathrm{d}t \right), \tag{42}$$

which yields the desired result. □

Lemma C.2. *Let $\mathbb{Q}_{x_0}$ be the forward path measure starting from x_0, and $\mathbb{P}^\theta$ be the neural reverse path measure. For a path ω with jumps at times $0 < t_1 < \cdots < t_n < T$ and visited states $x_0 \to x_1 \to \cdots \to x_n$, the log Radon-Nikodym derivative is:*

$$\begin{aligned} \log \frac{\mathrm{d}\mathbb{P}^\theta}{\mathrm{d}\mathbb{Q}_{x_0}}(\omega) = \log p_{\textit{prior}}(x_n) + \sum_{k=1}^{n} \log \frac{R^\theta_{t_k}(x_k, x_{k-1})}{R_{t_k}(x_{k-1}, x_k)} \\ + \int_0^T \left[\lambda_t^{\textit{fwd}}(X_t(\omega)) - \lambda_t^\theta(X_t(\omega)) \right] \mathrm{d}t. \end{aligned} \tag{43}$$

Proof. From Lemma 2.4, the forward path density (conditional on $X_0 = x_0$) is:

$$\mathrm{d}\mathbb{Q}_{x_0}(\omega) = \left[\prod_{k=1}^{n} R_{t_k}(x_{k-1}, x_k) \right] \exp\left(- \int_0^T \lambda_t^{\mathrm{fwd}}(X_t) \, \mathrm{d}t \right). \tag{44}$$

The neural reverse path density for the *same path* (note the index reversal: jumps are $x_k \to x_{k-1}$ in reverse time) is:

$$\mathrm{d}\mathbb{P}^\theta(\omega) = p_{\mathrm{prior}}(x_n) \left[\prod_{k=1}^{n} R^\theta_{t_k}(x_k, x_{k-1}) \right] \exp\left(- \int_0^T \lambda_t^\theta(X_t) \, \mathrm{d}t \right). \tag{45}$$

Taking the ratio:

$$\frac{\mathrm{d}\mathbb{P}^\theta}{\mathrm{d}\mathbb{Q}_{x_0}}(\omega) = p_{\mathrm{prior}}(x_n) \cdot \frac{\prod_{k=1}^{n} R^\theta_{t_k}(x_k, x_{k-1})}{\prod_{k=1}^{n} R_{t_k}(x_{k-1}, x_k)} \cdot \exp\left(\int_0^T \left[\lambda_t^{\mathrm{fwd}}(X_t) - \lambda_t^\theta(X_t) \right] \mathrm{d}t \right). \tag{46}$$

Taking logarithms yields (43). □

Theorem C.3. *For any data point $x_0 \in \mathcal{S}$, the negative log-likelihood satisfies:*

$$-\log p_\theta(x_0) \le -\mathbb{E}_{q_{T|0}}[\log p_{prior}(X_T)] + \int_0^T \sum_{i \in \mathcal{S}} q_{t|0}(i \mid x_0)\, \ell_t(i)\, \mathrm{d}t, \tag{47}$$

where the integrand $\ell_t(i)$ is defined as:

$$\ell_t(i) := \sum_{j \neq i} \left[R_t(i,j) \log \frac{R_t(i,j)}{R_t^\theta(j,i)} - R_t(i,j) + R_t^\theta(i,j) \right]. \tag{48}$$

Proof. **Step 1: Importance sampling.**

Since the forward path measure $\mathbb{Q}_{x_0}$ has $X_0 = x_0$ with probability 1, we can write:

$$\begin{aligned} p_\theta(x_0) &= \int_\Omega \mathbf{1}[X_0(\omega) = x_0]\, \mathrm{d}\mathbb{P}^\theta(\omega) \\ &= \int_\Omega \mathbf{1}[X_0(\omega) = x_0] \cdot \frac{\mathrm{d}\mathbb{P}^\theta}{\mathrm{d}\mathbb{Q}_{x_0}}(\omega)\, \mathrm{d}\mathbb{Q}_{x_0}(\omega) \quad \text{(Radon-Nikodym)} \\ &= \int_\Omega \frac{\mathrm{d}\mathbb{P}^\theta}{\mathrm{d}\mathbb{Q}_{x_0}}(\omega)\, \mathrm{d}\mathbb{Q}_{x_0}(\omega) \quad \text{(since } \mathbb{Q}_{x_0} \text{ starts from } x_0) \\ &= \mathbb{E}_{\mathbb{Q}_{x_0}} \left[\frac{\mathrm{d}\mathbb{P}^\theta}{\mathrm{d}\mathbb{Q}_{x_0}} \right]. \end{aligned} \tag{49}$$

Step 2: Jensen's inequality.

Since $\log$ is concave, we have:

$$\log p_\theta(x_0) = \log \mathbb{E}_{\mathbb{Q}_{x_0}} \left[\frac{\mathrm{d}\mathbb{P}^\theta}{\mathrm{d}\mathbb{Q}_{x_0}} \right] \ge \mathbb{E}_{\mathbb{Q}_{x_0}} \left[\log \frac{\mathrm{d}\mathbb{P}^\theta}{\mathrm{d}\mathbb{Q}_{x_0}} \right]. \tag{50}$$

Negating both sides:

$$-\log p_\theta(x_0) \le -\mathbb{E}_{\mathbb{Q}_{x_0}} \left[\log \frac{\mathrm{d}\mathbb{P}^\theta}{\mathrm{d}\mathbb{Q}_{x_0}} \right]. \tag{51}$$

Step 3: Expand using Lemma 3.1.

Substituting (43) into (51):

$$\begin{aligned} -\log p_\theta(x_0) \le\ & -\mathbb{E}_{\mathbb{Q}_{x_0}}[\log p_{\text{prior}}(X_T)] \\ & - \mathbb{E}_{\mathbb{Q}_{x_0}} \left[\sum_{k=1}^N \log \frac{R_{t_k}^\theta(X_{t_k}, X_{t_k^-})}{R_{t_k}(X_{t_k^-}, X_{t_k})} \right] \\ & - \mathbb{E}_{\mathbb{Q}_{x_0}} \left[\int_0^T (\lambda_t^{\text{fwd}}(X_t) - \lambda_t^\theta(X_t))\, \mathrm{d}t \right], \end{aligned} \tag{52}$$

where N is the random number of jumps.

Step 4: Apply Campbell-Mecke formula (Theorem A.7).

For the jump term, set $f(t,i,j) = \log \frac{R_t(i,j)}{R_t^\theta(j,i)}$:

$$\mathbb{E}_{\mathbb{Q}_{x_0}} \left[\sum_{k=1}^N \log \frac{R_{t_k}(X_{t_k^-}, X_{t_k})}{R_{t_k}^\theta(X_{t_k}, X_{t_k^-})} \right] = \int_0^T \sum_{i \in \mathcal{S}} q_{t|0}(i \mid x_0) \sum_{j \neq i} R_t(i,j) \log \frac{R_t(i,j)}{R_t^\theta(j,i)}\, \mathrm{d}t. \tag{53}$$

For the survival term, by Fubini's theorem:

$$\begin{aligned}\mathbb{E}_{\mathbb{Q}_{x_0}}\left[\int_0^T (\lambda_t^{\text{fwd}}(X_t) - \lambda_t^\theta(X_t))\,\mathrm{d}t\right] &= \int_0^T \mathbb{E}_{\mathbb{Q}_{x_0}}[\lambda_t^{\text{fwd}}(X_t) - \lambda_t^\theta(X_t)]\,\mathrm{d}t \\ &= \int_0^T \sum_{i\in\mathcal{S}} q_{t|0}(i \mid x_0)\left[\lambda_t^{\text{fwd}}(i) - \lambda_t^\theta(i)\right]\mathrm{d}t \\ &= \int_0^T \sum_{i\in\mathcal{S}} q_{t|0}(i \mid x_0) \sum_{j\neq i}\left[R_t(i,j) - R_t^\theta(i,j)\right]\mathrm{d}t.\end{aligned} \tag{54}$$

Step 5: Combine jump and survival terms.

From Steps 3 and 4, we need to compute:

$$-\mathbb{E}_{\mathbb{Q}_{x_0}}\left[\log\frac{\mathrm{d}\mathbb{P}^\theta}{\mathrm{d}\mathbb{Q}_{x_0}}\right] = -\mathbb{E}_{\mathbb{Q}_{x_0}}[\log p_{\text{prior}}(X_T)] + I_{\text{jump}} - I_{\text{surv}}, \tag{55}$$

where the jump and survival integrals are (from Step 4):

$$I_{\text{jump}} = \int_0^T \sum_{i\in\mathcal{S}} q_{t|0}(i \mid x_0) \sum_{j\neq i} R_t(i,j)\log\frac{R_t(i,j)}{R_t^\theta(j,i)}\,\mathrm{d}t, \tag{56}$$

$$I_{\text{surv}} = \int_0^T \sum_{i\in\mathcal{S}} q_{t|0}(i \mid x_0) \sum_{j\neq i}\left[R_t(i,j) - R_t^\theta(i,j)\right]\mathrm{d}t. \tag{57}$$

Combining (39) and the survival term:

$$\begin{aligned}I_{\text{jump}} - I_{\text{surv}} &= \int_0^T \sum_{i\in\mathcal{S}} q_{t|0}(i \mid x_0) \sum_{j\neq i}\left[R_t(i,j)\log\frac{R_t(i,j)}{R_t^\theta(j,i)} - \big(R_t(i,j) - R_t^\theta(i,j)\big)\right]\mathrm{d}t \\ &= \int_0^T \sum_{i\in\mathcal{S}} q_{t|0}(i \mid x_0) \sum_{j\neq i}\left[R_t(i,j)\log\frac{R_t(i,j)}{R_t^\theta(j,i)} - R_t(i,j) + R_t^\theta(i,j)\right]\mathrm{d}t \\ &= \int_0^T \sum_{i\in\mathcal{S}} q_{t|0}(i \mid x_0)\,\ell_t(i)\,\mathrm{d}t,\end{aligned} \tag{58}$$

where we define:

$$\ell_t(i) := \sum_{j\neq i}\left[R_t(i,j)\log\frac{R_t(i,j)}{R_t^\theta(j,i)} - R_t(i,j) + R_t^\theta(i,j)\right], \tag{59}$$

which exactly matches (48). This completes the proof. □

Theorem C.4. *The expected negative log-likelihood over the data distribution satisfies:*

$$\mathbb{E}_{x_0\sim p_{data}}[-\log p_\theta(x_0)] \le \mathrm{KL}(\hat{\mathbb{Q}}\|\mathbb{P}^\theta) + C_{total}, \tag{60}$$

where $\hat{\mathbb{Q}}$ is the path measure of the true reverse process, $\mathbb{P}^\theta$ is the path measure of the neural reverse process, and C_{total} is a constant independent of θ. Explicitly:

$$\mathrm{KL}(\hat{\mathbb{Q}}\|\mathbb{P}^\theta) = \int_0^T \sum_{i\in\mathcal{S}} q_t(i) \sum_{j\neq i}\left[\hat{R}_t(i,j)\log\frac{\hat{R}_t(i,j)}{R_t^\theta(i,j)} - \hat{R}_t(i,j) + R_t^\theta(i,j)\right]\mathrm{d}t. \tag{61}$$

Proof. **Step 1: Average over data.**

Taking expectation of both sides of (47) over $x_0 \sim p_{\text{data}}$:

$$\mathbb{E}_{p_{\text{data}}}[-\log p_\theta(x_0)] \le \mathbb{E}_{p_{\text{data}}}[-\mathbb{E}_{q_{T|0}}[\log p_{\text{prior}}(X_T)]] + \mathbb{E}_{p_{\text{data}}}\left[\int_0^T \sum_{i\in\mathcal{S}} q_{t|0}(i \mid x_0)\,\ell_t(i)\,\mathrm{d}t\right]. \tag{62}$$

Step 2: Define marginal distribution.

Define the (unconditional) marginal distribution at time t:

$$q_t(i) := \mathbb{E}_{x_0 \sim p_{\text{data}}}[q_{t|0}(i \mid x_0)] = \int_{\mathcal{S}} p_{\text{data}}(x_0)\, q_{t|0}(i \mid x_0). \tag{63}$$

By Fubini's theorem:

$$\mathbb{E}_{p_{\text{data}}}\left[\int_0^T \sum_{i \in \mathcal{S}} q_{t|0}(i \mid x_0)\, \ell_t(i)\, \mathrm{d}t\right] = \int_0^T \sum_{i \in \mathcal{S}} q_t(i)\, \ell_t(i)\, \mathrm{d}t. \tag{64}$$

Step 3: Transform jump term.

Expand $\ell_t(i)$ from (48):

$$q_t(i)\, \ell_t(i) = \sum_{j \neq i} q_t(i) \left[R_t(i,j) \log \frac{R_t(i,j)}{R_t^\theta(j,i)} - R_t(i,j) + R_t^\theta(i,j)\right]. \tag{65}$$

For the **jump term**, since $\hat{R}_t(j,i) = R_t(i,j)\frac{q_t(i)}{q_t(j)}$, we have $q_t(i)R_t(i,j) = q_t(j)\hat{R}_t(j,i)$. Substitute:

$$q_t(i)R_t(i,j) \log \frac{R_t(i,j)}{R_t^\theta(j,i)} = q_t(j)\hat{R}_t(j,i) \log \frac{R_t(i,j)}{R_t^\theta(j,i)}. \tag{66}$$

Since $R_t(i,j) = \hat{R}_t(j,i)\frac{q_t(j)}{q_t(i)}$:

$$\begin{aligned} q_t(j)\hat{R}_t(j,i) \log \frac{R_t(i,j)}{R_t^\theta(j,i)} &= q_t(j)\hat{R}_t(j,i) \log \frac{\hat{R}_t(j,i)\, q_t(j)/q_t(i)}{R_t^\theta(j,i)} \\ &= q_t(j)\hat{R}_t(j,i) \left[\log \frac{\hat{R}_t(j,i)}{R_t^\theta(j,i)} + \log \frac{q_t(j)}{q_t(i)}\right]. \end{aligned} \tag{67}$$

Summing over i and j (relabel $i \leftrightarrow j$):

$$\begin{aligned} \sum_{i \in \mathcal{S}} \sum_{j \neq i} q_t(i)R_t(i,j) \log \frac{R_t(i,j)}{R_t^\theta(j,i)} = &\sum_{i \in \mathcal{S}} \sum_{j \neq i} q_t(i)\hat{R}_t(i,j) \log \frac{\hat{R}_t(i,j)}{R_t^\theta(i,j)} \\ &+ \underbrace{\sum_{i \in \mathcal{S}} \sum_{j \neq i} q_t(i)\hat{R}_t(i,j) \log \frac{q_t(i)}{q_t(j)}}_{C_t^{(1)} \text{ (independent of } \theta)}. \end{aligned} \tag{68}$$

Step 4: Transform survival term.

For the **survival term** $\sum_{j \neq i}[-R_t(i,j) + R_t^\theta(i,j)]$, since $q_t(i)R_t(i,j) = q_t(j)\hat{R}_t(j,i)$:

$$\sum_{i \in \mathcal{S}} q_t(i)\lambda_t^{\text{fwd}}(i) = \sum_{i \in \mathcal{S}} q_t(i) \sum_{j \neq i} R_t(i,j) = \sum_{i \in \mathcal{S}} \sum_{j \neq i} q_t(j)\hat{R}_t(j,i) = \sum_{i \in \mathcal{S}} q_t(i)\hat{\lambda}_t(i). \tag{69}$$

Therefore:

$$\begin{aligned} \sum_{i \in \mathcal{S}} q_t(i) \sum_{j \neq i}[-R_t(i,j) + R_t^\theta(i,j)] &= -\sum_{i \in \mathcal{S}} q_t(i)\lambda_t^{\text{fwd}}(i) + \sum_{i \in \mathcal{S}} q_t(i)\lambda_t^\theta(i) \\ &= -\sum_{i \in \mathcal{S}} q_t(i)\hat{\lambda}_t(i) + \sum_{i \in \mathcal{S}} q_t(i)\lambda_t^\theta(i) \\ &= \sum_{i \in \mathcal{S}} q_t(i) \sum_{j \neq i}[R_t^\theta(i,j) - \hat{R}_t(i,j)]. \end{aligned} \tag{70}$$

Step 5: Combine jump and survival terms.

Putting everything together from Steps 3 and 4:

$$\int_0^T \sum_{i\in\mathcal{S}} q_t(i)\,\ell_t(i)\,\mathrm{d}t = \int_0^T \sum_{i\in\mathcal{S}} q_t(i) \sum_{j\neq i} \left[\hat{R}_t(i,j)\log\frac{\hat{R}_t(i,j)}{R_t^\theta(i,j)} + R_t^\theta(i,j) - \hat{R}_t(i,j)\right]\mathrm{d}t + \int_0^T C_t^{(1)}\,\mathrm{d}t. \tag{71}$$

The first term can be rewritten as:

$$\int_0^T \sum_{i\in\mathcal{S}} q_t(i) \sum_{j\neq i} \left[\hat{R}_t(i,j)\log\frac{\hat{R}_t(i,j)}{R_t^\theta(i,j)} - \hat{R}_t(i,j) + R_t^\theta(i,j)\right]\mathrm{d}t = \mathrm{KL}(\hat{\mathbb{Q}}\|\mathbb{P}^\theta), \tag{72}$$

and the second term (plus the prior term) is C_{total}, independent of θ. □

Corollary C.5. *Define the true reverse quantities:*

$$\begin{aligned}\hat{\lambda}_t(i) &:= \sum_{j\neq i}\hat{R}_t(i,j) \quad \textit{(exit rate)},\\ \hat{r}_t(j\mid i) &:= \frac{\hat{R}_t(i,j)}{\hat{\lambda}_t(i)} \quad \textit{for } j\neq i \quad \textit{(jump distribution)},\end{aligned} \tag{73}$$

and similarly for the neural reverse:

$$\begin{aligned}\lambda_t^\theta(i) &:= \sum_{j\neq i} R_t^\theta(i,j),\\ r_t^\theta(j\mid i) &:= \frac{R_t^\theta(i,j)}{\lambda_t^\theta(i)}.\end{aligned} \tag{74}$$

Then:

$$\mathbb{E}_{p_{data}}[-\log p_\theta(x_0)] \le \int_0^T \sum_{i\in\mathcal{S}} q_t(i)\left[\mathrm{KL}^{Poi}(\hat{\lambda}_t(i)\|\lambda_t^\theta(i)) + \hat{\lambda}_t(i)\cdot\mathrm{KL}^{Cat}(\hat{r}_t(\cdot\mid i)\|r_t^\theta(\cdot\mid i))\right]\mathrm{d}t + C, \tag{75}$$

where:

$$\begin{aligned}\mathrm{KL}^{Poi}(\lambda\|\lambda^\theta) &:= \lambda\log\frac{\lambda}{\lambda^\theta} - \lambda + \lambda^\theta,\\ \mathrm{KL}^{Cat}(\hat{r}\|r^\theta) &:= \sum_{j\neq i}\hat{r}(j\mid i)\log\frac{\hat{r}(j\mid i)}{r^\theta(j\mid i)}.\end{aligned} \tag{76}$$

Proof. Substitute $\hat{R}_t(i,j) = \hat{\lambda}_t(i)\hat{r}_t(j\mid i)$ and $R_t^\theta(i,j) = \lambda_t^\theta(i)r_t^\theta(j\mid i)$ into (61):

$$\begin{aligned}&\sum_{j\neq i}\left[\hat{R}_t(i,j)\log\frac{\hat{R}_t(i,j)}{R_t^\theta(i,j)} - \hat{R}_t(i,j) + R_t^\theta(i,j)\right]\\ &= \sum_{j\neq i}\hat{\lambda}_t(i)\hat{r}_t(j\mid i)\log\frac{\hat{\lambda}_t(i)\hat{r}_t(j\mid i)}{\lambda_t^\theta(i)r_t^\theta(j\mid i)} - \hat{\lambda}_t(i) + \lambda_t^\theta(i)\\ &= \hat{\lambda}_t(i)\log\frac{\hat{\lambda}_t(i)}{\lambda_t^\theta(i)} + \hat{\lambda}_t(i)\sum_{j\neq i}\hat{r}_t(j\mid i)\log\frac{\hat{r}_t(j\mid i)}{r_t^\theta(j\mid i)} - \hat{\lambda}_t(i) + \lambda_t^\theta(i)\\ &= \mathrm{KL}^{\mathrm{Poi}}(\hat{\lambda}_t(i)\|\lambda_t^\theta(i)) + \hat{\lambda}_t(i)\cdot\mathrm{KL}^{\mathrm{Cat}}(\hat{r}_t(\cdot\mid i)\|r_t^\theta(\cdot\mid i)).\end{aligned} \tag{77}$$

□

Corollary C.6. *The training objective* (14) *can be estimated via Monte Carlo sampling:*

$$\mathcal{L}(\theta) := \mathbb{E}_{t\sim\mathcal{U}(0,T),\, x_t\sim q_t}\left[\mathrm{KL}^{Poi}(\hat{\lambda}_t(x_t)\|\lambda_t^\theta(x_t)) + \hat{\lambda}_t(x_t)\cdot\mathbb{E}_{j\sim\hat{r}_t(\cdot|x_t)}\left[-\log r_t^\theta(j\mid x_t)\right] - \hat{\lambda}_t(x_t)H(\hat{r}_t)\right], \tag{78}$$

where $H(\hat{r}_t) := -\sum_{j\neq i}\hat{r}_t(j\mid i)\log\hat{r}_t(j\mid i)$ *is the entropy of the true reverse jump distribution. Equivalently, using the cross-entropy (CE) identity* $\mathrm{KL}^{Cat} = CE - H$*:*

$$\mathcal{L}(\theta) = \mathbb{E}_{t,x_t}\left[\mathrm{KL}^{Poi}(\hat{\lambda}_t(x_t)\|\lambda_t^\theta(x_t)) + \hat{\lambda}_t(x_t)\cdot CE(\hat{r}_t(\cdot\mid x_t), r_t^\theta(\cdot\mid x_t))\right] + const. \tag{79}$$

Proof. Direct application of Monte Carlo integration to (14):

$$\int_0^T\sum_{i\in\mathcal{S}} q_t(i)\cdot(\cdots)\,\mathrm{d}t = T\cdot\mathbb{E}_{t\sim\mathcal{U}(0,T)}\left[\sum_{i\in\mathcal{S}} q_t(i)\cdot(\cdots)\right] = T\cdot\mathbb{E}_{t\sim\mathcal{U}(0,T),\, x_t\sim q_t}[(\cdots)]. \tag{80}$$

The categorical KL can be written as:

$$\mathrm{KL}^{\mathrm{Cat}}(\hat{r}_t\|r_t^\theta) = \mathbb{E}_{j\sim\hat{r}_t}[-\log r_t^\theta(j\mid x_t)] - H(\hat{r}_t), \tag{81}$$

where the first term is the cross-entropy and the second is the entropy (independent of θ). □

Lemma C.7. *Let* p_0 *be a distribution over* $\mathcal{S}$*, and let* $p(\cdot\mid x_0)$ *be conditional distributions such that* $p(x) = \sum_{x_0\in\mathcal{S}} p_0(x_0)p(x\mid x_0)$ *is the marginal distribution. For any measurable function* $f:\mathcal{S}\times\mathcal{S}\to\mathbb{R}$*, we have:*

$$\mathbb{E}_{x\sim p}\left[\sum_{y\neq x} f(x,y)\frac{p(y)}{p(x)}\right] = \mathbb{E}_{x_0\sim p_0,\, x\sim p(\cdot|x_0)}\left[\sum_{y\neq x} f(x,y)\frac{p(y\mid x_0)}{p(x\mid x_0)}\right]. \tag{82}$$

Proof.

$$\begin{aligned}
\mathbb{E}_{x\sim p}\left[\sum_{y\neq x} f(x,y)\frac{p(y)}{p(x)}\right] &= \sum_{x\in\mathcal{S}} p(x)\sum_{y\neq x} f(x,y)\frac{p(y)}{p(x)}\\
&= \sum_{x\in\mathcal{S}}\sum_{y\neq x} f(x,y)p(y)\\
&= \sum_{x\in\mathcal{S}}\sum_{y\neq x} f(x,y)\sum_{x_0\in\mathcal{S}} p_0(x_0)p(y\mid x_0)\\
&= \sum_{x_0\in\mathcal{S}} p_0(x_0)\sum_{x\in\mathcal{S}}\sum_{y\neq x} f(x,y)p(y\mid x_0)\\
&= \sum_{x_0\in\mathcal{S}} p_0(x_0)\sum_{x\in\mathcal{S}} p(x\mid x_0)\sum_{y\neq x} f(x,y)\frac{p(y\mid x_0)}{p(x\mid x_0)}\\
&= \mathbb{E}_{x_0\sim p_0,\, x\sim p(\cdot|x_0)}\left[\sum_{y\neq x} f(x,y)\frac{p(y\mid x_0)}{p(x\mid x_0)}\right].
\end{aligned} \tag{83}$$

□

Proposition C.8. *For a fixed data point* $x_0\in\mathcal{S}$*, define the conditional reverse rates:*

$$\hat{R}_{t|0}(j,i\mid x_0) := R_t(i,j)\frac{q_{t|0}(i\mid x_0)}{q_{t|0}(j\mid x_0)},\quad i\neq j, \tag{84}$$

$$\hat{\lambda}_{t|0}(i\mid x_0) := \sum_{j\neq i}\hat{R}_{t|0}(i,j\mid x_0) = \sum_{j\neq i} R_t(j,i)\frac{q_{t|0}(j\mid x_0)}{q_{t|0}(i\mid x_0)}, \tag{85}$$

$$\hat{r}_{t|0}(j \mid i, x_0) := \frac{\hat{R}_{t|0}(i, j \mid x_0)}{\hat{\lambda}_{t|0}(i \mid x_0)} = \frac{R_t(j, i)\frac{q_{t|0}(j|x_0)}{q_{t|0}(i|x_0)}}{\hat{\lambda}_{t|0}(i \mid x_0)}, \quad j \neq i. \tag{86}$$

Then the training loss (79) *can be equivalently written as:*

$$\mathcal{L}(\theta) = \mathbb{E}_{t\sim\mathcal{U}(0,T),\, x_0\sim p_{data},\, i\sim q_{t|0}(\cdot|x_0)} \left[-\sum_{j\neq i} R_t(j,i)\frac{q_{t|0}(j \mid x_0)}{q_{t|0}(i \mid x_0)} \log \lambda_t^\theta(i) r_t^\theta(j \mid i) - \hat{\lambda}_{t|0}(i \mid x_0) + \lambda_t^\theta(i) \right] + \textit{const}. \tag{87}$$

Proof. We expand the training loss (79) and apply Lemma C.7 to transform marginal distributions to conditional distributions.

Starting from (79), expand the KL divergence and cross-entropy:

$$\begin{aligned}
\mathcal{L}(\theta) &= \mathbb{E}_{t\sim\mathcal{U}(0,T),\, i\sim q_t} \left[\mathrm{KL}^{\mathrm{Poi}}(\hat{\lambda}_t(i) \| \lambda_t^\theta(i)) + \hat{\lambda}_t(i) \cdot CE(\hat{r}_t(\cdot \mid i), r_t^\theta(\cdot \mid i)) \right] + \text{const} \\
&= \mathbb{E}_{t\sim\mathcal{U}(0,T),\, i\sim q_t} \left[\hat{\lambda}_t(i) \log \frac{\hat{\lambda}_t(i)}{\lambda_t^\theta(i)} - \hat{\lambda}_t(i) + \lambda_t^\theta(i) - \hat{\lambda}_t(i) \sum_{j\neq i} \hat{r}_t(j \mid i) \log r_t^\theta(j \mid i) \right] + \text{const}.
\end{aligned} \tag{88}$$

Using $\hat{\lambda}_t(i) = \sum_{j\neq i} \hat{R}_t(i,j) = \sum_{j\neq i} R_t(j,i)\frac{q_t(j)}{q_t(i)}$ and $\hat{\lambda}_t(i)\hat{r}_t(j \mid i) = \hat{R}_t(i,j) = R_t(j,i)\frac{q_t(j)}{q_t(i)}$, we can write:

The first term:

$$\begin{aligned}
\mathbb{E}_{t\sim\mathcal{U}(0,T),\, i\sim q_t} \left[\hat{\lambda}_t(i) \log \frac{\hat{\lambda}_t(i)}{\lambda_t^\theta(i)} \right] &= \mathbb{E}_{t,i} \left[\sum_{j\neq i} R_t(j,i)\frac{q_t(j)}{q_t(i)} \log \frac{\sum_{k\neq i} R_t(k,i)\frac{q_t(k)}{q_t(i)}}{\lambda_t^\theta(i)} \right] \\
&= \mathbb{E}_{t,i} \left[\sum_{j\neq i} R_t(j,i)\frac{q_t(j)}{q_t(i)} \log \left(\sum_{k\neq i} R_t(k,i)\frac{q_t(k)}{q_t(i)} \right) \right] \\
&\quad - \mathbb{E}_{t\sim\mathcal{U}(0,T),\, x_0\sim p_{\text{data}},\, i\sim q_{t|0}(\cdot|x_0)} \left[\sum_{j\neq i} R_t(j,i)\frac{q_{t|0}(j \mid x_0)}{q_{t|0}(i \mid x_0)} \log \lambda_t^\theta(i) \right] \\
&= \text{const} - \mathbb{E}_{t\sim\mathcal{U}(0,T),\, x_0\sim p_{\text{data}},\, i\sim q_{t|0}(\cdot|x_0)} \left[\hat{\lambda}_{t|0}(i \mid x_0) \log \lambda_t^\theta(i) \right].
\end{aligned} \tag{89}$$

The second term:

$$\begin{aligned}
\mathbb{E}_{t\sim\mathcal{U}(0,T),\, i\sim q_t} \left[-\hat{\lambda}_t(i) \right] &= \mathbb{E}_{t,i} \left[-\sum_{j\neq i} R_t(j,i)\frac{q_t(j)}{q_t(i)} \right] \\
&= \mathbb{E}_{t\sim\mathcal{U}(0,T),\, x_0\sim p_{\text{data}},\, i\sim q_{t|0}(\cdot|x_0)} \left[-\sum_{j\neq i} R_t(j,i)\frac{q_{t|0}(j \mid x_0)}{q_{t|0}(i \mid x_0)} \right] \\
&= \mathbb{E}_{t\sim\mathcal{U}(0,T),\, x_0\sim p_{\text{data}},\, i\sim q_{t|0}(\cdot|x_0)} \left[-\hat{\lambda}_{t|0}(i \mid x_0) \right].
\end{aligned} \tag{90}$$

The third term:

$$\begin{aligned}\mathbb{E}_{t\sim\mathcal{U}(0,T),\, i\sim q_t}\left[-\hat{\lambda}_t(i)\sum_{j\neq i}\hat{r}_t(j\mid i)\log r_t^\theta(j\mid i)\right] &= \mathbb{E}_{t,i}\left[-\sum_{j\neq i}\hat{\lambda}_t(i)\hat{r}_t(j\mid i)\log r_t^\theta(j\mid i)\right]\\ &= \mathbb{E}_{t,i}\left[-\sum_{j\neq i}R_t(j,i)\frac{q_t(j)}{q_t(i)}\log r_t^\theta(j\mid i)\right]\\ &= \mathbb{E}_{t,\,x_0,\,i}\left[-\sum_{j\neq i}R_t(j,i)\frac{q_{t|0}(j\mid x_0)}{q_{t|0}(i\mid x_0)}\log r_t^\theta(j\mid i)\right]\end{aligned} \tag{91}$$

Therefore, combining the three terms leads to the conclusion.

□

Proposition C.9. *Assume: (i) forward quantities (q_t, $q_{t|0}$, $p(x_0 \mid x_t)$) are independent of θ; (ii) $R_t^\theta(i,j) > 0$ and differentiable in θ for all $i \neq j$; and (iii) differentiation can be exchanged with expectation/integration. Then there exists a θ-independent constant $C_{\mathrm{gap}} \geq 0$ such that*

$$\mathcal{L}_{\mathrm{KL}}(\theta) = \mathrm{KL}(\hat{\mathbb{Q}}\|\mathbb{P}^\theta) + C_{\mathrm{gap}},$$

hence

$$\nabla_\theta\mathcal{L}_{\mathrm{KL}}(\theta) = \nabla_\theta\mathrm{KL}(\hat{\mathbb{Q}}\|\mathbb{P}^\theta).$$

In particular,

$$\arg\min_\theta \mathcal{L}_{\mathrm{KL}}(\theta) = \arg\min_\theta \mathrm{KL}(\hat{\mathbb{Q}}\|\mathbb{P}^\theta),$$

and every stationary point of one objective is a stationary point of the other.

Proof. For fixed (t,i,j), let $R := \hat{R}_t(i,j \mid X_0)$ with $X_0 \sim p(x_0 \mid x_t = i)$ and let $c := R_t^\theta(i,j)$. Since $\mathbb{E}[R] = \hat{R}_t(i,j)$, expanding $f(r,c) = r\log\frac{r}{c} - r + c$ gives

$$\mathbb{E}[f(R,c)] - f(\mathbb{E}[R],c) = \mathbb{E}[R\log R] - \mathbb{E}[R]\log\mathbb{E}[R] =: \Delta_{t,i,j} \geq 0,$$

where $\Delta_{t,i,j}$ is independent of c and thus independent of θ. Multiplying by $q_t(i)$, summing over $i \in \mathcal{S}$ and $j \neq i$, and integrating over t yields

$$\mathcal{L}_{\mathrm{KL}}(\theta) = \mathrm{KL}(\hat{\mathbb{Q}}\|\mathbb{P}^\theta) + \int_0^T \sum_{i\in\mathcal{S}} q_t(i)\sum_{j\neq i}\Delta_{t,i,j}\,\mathrm{d}t.$$

Setting $C_{\mathrm{gap}} := \int_0^T \sum_{i\in\mathcal{S}} q_t(i)\sum_{j\neq i}\Delta_{t,i,j}\,\mathrm{d}t \geq 0$, under assumptions (i)–(iii), differentiation can be passed through the expectation and integral, so $\nabla_\theta\mathcal{L}_{\mathrm{KL}}(\theta) = \nabla_\theta\mathrm{KL}(\hat{\mathbb{Q}}\|\mathbb{P}^\theta)$. Adding a θ-independent constant does not change the minimizers, and identical gradients imply identical stationary points. □

Corollary C.10. *Defining $K(a) := a(\log a - 1)$, the loss $\mathcal{L}_{\mathrm{KL}}(\theta)$ equals:*

$$\begin{aligned}\mathcal{L}_{\mathrm{cond}}(\theta) = \mathbb{E}_{t,\,x_0,\,i}\Big[&\sum_{j\neq i}\lambda_t^\theta(i)r_t^\theta(j\mid i) - \sum_{j\neq i}R_t(j,i)\frac{q_{t|0}(j\mid x_0)}{q_{t|0}(i\mid x_0)}\log\frac{\lambda_t^\theta(i)\,r_t^\theta(j\mid i)}{R_t(j,i)}\\ &+\sum_{j\neq i}R_t(j,i)\,K\!\left(\frac{q_{t|0}(j\mid x_0)}{q_{t|0}(i\mid x_0)}\right)\Big].\end{aligned} \tag{92}$$

Proof. For fixed (t,x_0,i,j), write

$$\hat{R}_{t|0}(i,j\mid x_0) = R_t(j,i)\,\frac{q_{t|0}(j\mid x_0)}{q_{t|0}(i\mid x_0)} \quad\text{and}\quad R_t^\theta(i,j) = \lambda_t^\theta(i)\,r_t^\theta(j\mid i).$$

Expanding the Poisson KL density gives

$$f\Big(\hat{R}_{t|0}(i,j \mid x_0), R_t^\theta(i,j)\Big) = \hat{R}_{t|0}(i,j \mid x_0)\log\frac{\hat{R}_{t|0}(i,j \mid x_0)}{R_t^\theta(i,j)} - \hat{R}_{t|0}(i,j \mid x_0) + R_t^\theta(i,j)$$
$$= R_t^\theta(i,j) - \hat{R}_{t|0}(i,j \mid x_0)\log\frac{R_t^\theta(i,j)}{R_t(j,i)} \tag{93}$$
$$+ R_t(j,i)\,K\left(\frac{q_{t|0}(j \mid x_0)}{q_{t|0}(i \mid x_0)}\right).$$

Summing over $j \neq i$ and taking expectation over (t, x_0, i) yields $\mathcal{L}_{\mathrm{cond}}(\theta)$. Since $\mathcal{L}_{\mathrm{KL}}(\theta)$ is defined as the expectation of the same pairwise density f, this proves $\mathcal{L}_{\mathrm{cond}}(\theta) = \mathcal{L}_{\mathrm{KL}}(\theta)$. □

Corollary C.11 (Equivalence to MDLM, restated)**.** *If the forward process is the masked (absorbing) process on $\mathcal{S} = \mathcal{V} \cup \{m\}$, then for any valid noise schedule α_t, the loss* (22) *reduces to the MDLM cross-entropy loss* (23)*.*

Proof. We work directly with the full loss $\mathcal{L}_{\mathrm{cond}}(\theta)$ in (22), including its θ-independent term. For the absorbing process,

$$q_{t|0}(\cdot \mid x_0) = \alpha_t \boldsymbol{x}_0 + (1-\alpha_t)\boldsymbol{e}_m, \qquad R_t(i,j) = \rho_t(\boldsymbol{e}_j^\top \boldsymbol{e}_m)\mathbf{1}_{\{i\neq m\}}, \qquad \rho_t := -\frac{\alpha_t'}{\alpha_t}.$$

Define the reverse coefficient

$$\gamma_t := \rho_t \frac{\alpha_t}{1-\alpha_t} = -\frac{\alpha_t'}{1-\alpha_t} > 0.$$

The θ-dependent part of the integrand decomposes as

$$\underbrace{-\sum_{j\neq i}\hat{R}_t(i,j \mid x_0)\log R_t^\theta(i,j)}_{\text{Term ①}} + \underbrace{\sum_{j\neq i} R_t^\theta(i,j)}_{\text{Term ②}} + (\theta\text{-independent terms}),$$

where $\hat{R}_t(i,j \mid x_0) = \gamma_t\, \boldsymbol{e}_j^\top(\boldsymbol{x}_0 - \boldsymbol{e}_m)\mathbf{1}_{\{i=m\}}$ and $R_t^\theta(i,j) = \gamma_t\, \boldsymbol{e}_j^\top(x_\theta - \boldsymbol{e}_m)\mathbf{1}_{\{i=m\}}$.

When $i \neq m$, both rates vanish, so the integrand is zero. For $i = m$, the sum runs over $j \neq m$, and orthogonality $\boldsymbol{e}_j^\top \boldsymbol{e}_m = 0$ simplifies the rates to $\hat{R}_t(m,j \mid x_0) = \gamma_t\, \boldsymbol{e}_j^\top \boldsymbol{x}_0$ and $R_t^\theta(m,j) = \gamma_t\, \boldsymbol{e}_j^\top x_\theta$.

Term ①:

$$① = -\sum_{j\neq m}\gamma_t(\boldsymbol{e}_j^\top \boldsymbol{x}_0)\log\big(\gamma_t\, \boldsymbol{e}_j^\top x_\theta\big) = -\gamma_t\log\gamma_t \underbrace{\sum_{j\neq m}(\boldsymbol{e}_j^\top \boldsymbol{x}_0)}_{=1} - \gamma_t \sum_{j\neq m}(\boldsymbol{e}_j^\top \boldsymbol{x}_0)\log(\boldsymbol{e}_j^\top x_\theta)$$
$$= -\gamma_t\log\gamma_t - \gamma_t\, \boldsymbol{x}_0^\top \log x_\theta,$$

where we used that x_0 is one-hot on $\mathcal{V}$ so $\sum_{j\neq m} \boldsymbol{e}_j^\top \boldsymbol{x}_0 = 1$.

Term ②:

$$② = \sum_{j\neq m}\gamma_t\, \boldsymbol{e}_j^\top x_\theta = \gamma_t \underbrace{\sum_{j\neq m}\boldsymbol{e}_j^\top x_\theta}_{=1} = \gamma_t,$$

using the normalization of x_θ on $\mathcal{V}$.

Combining: setting $C_t := -\gamma_t\log\gamma_t + \gamma_t$ (independent of θ):

$$① + ② = -\gamma_t\, \boldsymbol{x}_0^\top \log x_\theta + C_t = \frac{\alpha_t'}{1-\alpha_t}\, \boldsymbol{x}_0^\top \log x_\theta + C_t.$$

It remains to evaluate the θ-independent terms in the full expression (22). For $i = m$, only the clean token contributes, so

$$\sum_{j\neq m}\hat{R}_t(m,j \mid x_0)\log R_t(j,m) + \sum_{j\neq m} R_t(j,m)K\left(\frac{q_{t|0}(j \mid x_0)}{q_{t|0}(m \mid x_0)}\right) = \gamma_t\log\rho_t + \rho_t K\left(\frac{\alpha_t}{1-\alpha_t}\right).$$

Using $K(a) = a(\log a - 1)$ and $\gamma_t = \rho_t \alpha_t / (1 - \alpha_t)$, this becomes

$$\gamma_t \log \rho_t + \gamma_t \left(\log \frac{\alpha_t}{1 - \alpha_t} - 1 \right) = \gamma_t \log \gamma_t - \gamma_t = -C_t.$$

Thus the θ-independent term in (22) exactly cancels the constant C_t produced by Terms ① and ②. Taking the expectation over (t, x_0, x_t) therefore yields (23) with no additive constant. □

Proposition C.12 (Pointwise optimality, restated)**.** *For fixed t and i, define $g_{t,i}(\lambda, r) := \mathrm{KL}^{Poi}\big(\hat{\lambda}_t(i) \| \lambda\big) + \hat{\lambda}_t(i)\, \mathrm{KL}^{Cat}\big(\hat{r}_t(\cdot \mid i) \| r\big)$. If $\hat{\lambda}_t(i) > 0$, then $g_{t,i}(\lambda, r) \geq 0$ with equality if and only if $\lambda = \hat{\lambda}_t(i)$ and $r = \hat{r}_t(\cdot \mid i)$. If $\hat{\lambda}_t(i) = 0$, the unique minimizer in λ is $\lambda = 0$ and the choice of r is irrelevant.*

Proof. Since both terms are non-negative, $g_{t,i}(\lambda, r) \geq 0$. For the Poisson KL, $\mathrm{KL}^{\mathrm{Poi}}(\hat{\lambda} \| \lambda) = \hat{\lambda} \log \frac{\hat{\lambda}}{\lambda} - \hat{\lambda} + \lambda$. Taking the derivative with respect to λ and setting it to zero: $-\hat{\lambda}/\lambda + 1 = 0$, giving $\lambda^* = \hat{\lambda}$. The second derivative $\hat{\lambda}/\lambda^2 > 0$ confirms this is a strict minimum, and $\mathrm{KL}^{\mathrm{Poi}}(\hat{\lambda} \| \hat{\lambda}) = 0$. For the categorical KL, $\mathrm{KL}^{\mathrm{Cat}}(\hat{r} \| r) \geq 0$ by Gibbs' inequality, with equality if and only if $r = \hat{r}$. When $\hat{\lambda}_t(i) = 0$, the Poisson KL reduces to $\lambda \geq 0$ with minimum at $\lambda = 0$, and the categorical term vanishes since it is weighted by $\hat{\lambda}_t(i) = 0$. □

D Additional Experimental Details

Datasets. We evaluate on two datasets: (1) TinyStories [7], a corpus of simple short stories for language modeling; (2) OpenWebText (OWT) [23], a large-scale web text corpus. Both are tokenized with the GPT-2 tokenizer (vocabulary size 50257) and truncated or padded to a maximum sequence length of 512.

Architecture. Neural CTMC uses a 12-layer DiT-style Transformer with hidden dimension 768, 12 attention heads, and a time-conditioning dimension of 128. The total parameter count is approximately 163M.

Training. We use AdamW with a learning rate of 5×10^{-4}, cosine learning rate schedule, and gradient clipping at 1.0. Training is performed in bf16 mixed precision. For TinyStories, we use batch size 256 and 100K training steps with 1K warmup steps. For OpenWebText, we use batch size 512 and 500K training steps with 10K warmup steps.

Algorithm 2 τ-Leaping

Set $\tau = \frac{T}{N}$ and sample $x_T \sim \mathrm{Uniform}(\mathcal{S})^L$
for $n = N$ **downto** 1 **do**
  $t \leftarrow n\tau$
  $\{(\lambda^\theta_{t,\ell}, r^\theta_{t,\ell})\}_{\ell=1}^L \leftarrow \Phi_\theta(x_t, t)$
  Set $x_{t-\tau} \leftarrow x_t$
  for $\ell = 1$ **to** L **do**
    Sample $\Delta_\ell \sim \mathrm{Exp}(\lambda^\theta_{t,\ell})$
    if $\Delta_\ell < \tau$ **then**
      $x^{(\ell)}_{t-\tau} \sim r^\theta_{t,\ell}(\cdot \mid x^{(\ell)}_t)$
    end if
  end for
end for
Return x_0

Algorithm 3 Euler

Set $\tau = \frac{T}{N}$ and sample $x_T \sim \mathrm{Uniform}(\mathcal{S})$
for $n = N$ **downto** 1 **do**
  $t \leftarrow n\tau$
  $\big(\lambda^\theta_t(x_t),\, r^\theta_t(\cdot \mid x_t)\big) \leftarrow \Phi_\theta(x_t, t)$
  Set $p_j \leftarrow \lambda^\theta_t(x_t) \cdot r^\theta_t(j \mid x_t) \cdot \tau$ for $j \neq x_t$
  Set $p_{x_t} \leftarrow 1 - \sum_{j \neq x_t} p_j$
  $x_{t-\tau} \sim \mathrm{Cat}\big(\{p_j\}_{j \in \mathcal{S}}\big)$
end for
Return x_0

Baselines. For TinyStories, we train MDLM and GIDD from their official source code using the same architecture and training budget as Neural CTMC. For OpenWebText, the comparison includes SEDD, MDLM, GIDD, and DUO. Neural CTMC, SEDD (uniform), and all GIDD variants are

trained on 131B tokens with Max Len 512. MDLM, DUO, and SEDD (mask) are trained on 262B, 524B, and 682B tokens, respectively, with Max Len 1024. Tables 1 and 2 report these checkpoint differences explicitly; all models are evaluated with the same sampling-step grid, number and length of generated samples, and scoring models.

Evaluation. On OpenWebText, we report mean generative perplexity (PPL) $\pm$ sample standard deviation over five sampling seeds. For a fair comparison, every model generates 1,024 unconditional samples of 512 tokens per seed; Gemma 9B and GPT-2 Large evaluate the same generated samples. We evaluate each final checkpoint using $\{16, 32, 64, 128, 256\}$ sampling steps. The Gemma 9B results are reported in Table 1 in the main text, and the GPT-2 Large results are given in Table 2 below.

Table 2: GPT-2 Large generative PPL on OpenWebText, evaluated on the same samples as Table 1 (mean $\pm$ std, lower is better). **Bold**: best among checkpoints with Train Toks 131B. †: overall best.

Type	Method	Train Toks	Max Len	16	32	64	128	256
mask	SEDD [17]	682B	1024	230.571 $\pm$1.500	136.543 $\pm$1.691	95.625 $\pm$1.052	74.196 $\pm$0.624	60.045 $\pm$0.400
mask	MDLM [24]	262B	1024	243.047 $\pm$2.291	137.317 $\pm$0.633	93.965† $\pm$0.829	72.243† $\pm$1.035	57.952† $\pm$0.987
mask	GIDD [31] ($p_{\text{unif}} = 0.0$)	131B	512	566.066 $\pm$10.590	297.354 $\pm$4.299	205.251 $\pm$5.606	169.339 $\pm$3.386	155.875 $\pm$3.181
mixture	GIDD [31] ($p_{\text{unif}} = 0.1$)	131B	512	239.518 $\pm$2.855	166.977 $\pm$1.344	136.454 $\pm$2.193	124.020 $\pm$2.357	118.165 $\pm$1.459
mixture	GIDD [31] ($p_{\text{unif}} = 0.2$)	131B	512	245.696 $\pm$4.107	176.264 $\pm$1.199	150.196 $\pm$2.064	140.027 $\pm$2.121	134.277 $\pm$1.256
continuous	DUO	524B	1024	132.835† $\pm$1.205	108.972† $\pm$0.844	100.086 $\pm$1.525	95.529 $\pm$1.502	93.433 $\pm$1.005
uniform	SEDD [17]	131B	512	334.007 $\pm$10.592	171.825 $\pm$2.967	130.487 $\pm$2.213	115.904 $\pm$0.616	112.059 $\pm$0.664
uniform	GIDD [31]	131B	512	574.170 $\pm$15.752	202.289 $\pm$4.054	139.239 $\pm$1.750	123.097 $\pm$1.535	115.513 $\pm$0.909
uniform	Neural CTMC (ours)	131B	512	**227.430 $\pm$3.214**	**135.640 $\pm$2.144**	**113.019 $\pm$0.772**	**109.279 $\pm$0.727**	**111.516 $\pm$1.070**

Unigram entropy. Table 3 reports unigram entropy on the same generated samples as Table 2, with mean and standard deviation over five sampling seeds. Higher entropy indicates that token usage is less concentrated on a few tokens.

Table 3: Unigram entropy on the same samples (mean $\pm$ std).

Type	Method	16	32	64	128	256
mask	SEDD [17]	5.3001 $\pm$0.0047	5.2284 $\pm$0.0054	5.1597 $\pm$0.0022	5.0858 $\pm$0.0063	5.0121 $\pm$0.0072
mask	MDLM [24]	5.3183 $\pm$0.0008	5.2458 $\pm$0.0034	5.1742 $\pm$0.0054	5.1080 $\pm$0.0087	5.0270 $\pm$0.0101
mask	GIDD [31] ($p_{\text{unif}} = 0.0$)	5.2081 $\pm$0.0052	5.1335 $\pm$0.0074	5.0941 $\pm$0.0108	5.1061 $\pm$0.0040	5.1166 $\pm$0.0112
mixture	GIDD [31] ($p_{\text{unif}} = 0.1$)	5.1761 $\pm$0.0033	5.0730 $\pm$0.0043	5.0427 $\pm$0.0061	5.0342 $\pm$0.0109	5.0380 $\pm$0.0083
mixture	GIDD [31] ($p_{\text{unif}} = 0.2$)	5.1815 $\pm$0.0042	5.1245 $\pm$0.0109	5.0963 $\pm$0.0062	5.0954 $\pm$0.0056	5.0876 $\pm$0.0068
continuous	DUO	5.2054 $\pm$0.0033	5.2049 $\pm$0.0011	5.1982 $\pm$0.0053	5.1817 $\pm$0.0053	5.1649 $\pm$0.0064
uniform	SEDD [17]	5.3500 $\pm$0.0111	5.2536 $\pm$0.0063	5.2114 $\pm$0.0059	5.1973 $\pm$0.0023	5.1907 $\pm$0.0026
uniform	GIDD [31]	5.4563 $\pm$0.0064	5.2433 $\pm$0.0060	5.1897 $\pm$0.0046	5.1703 $\pm$0.0029	5.1647 $\pm$0.0027
uniform	Neural CTMC (ours)	5.2380 $\pm$0.0031	5.1616 $\pm$0.0033	5.1187 $\pm$0.0032	5.1072 $\pm$0.0028	5.1138 $\pm$0.0022

Sampling. We evaluate two sampling schemes: Euler (Algorithm 3) and τ-leaping (Algorithm 2). The number of sampling steps is 50 for TinyStories training curves and is chosen from $\{16, 32, 64, 128, 256\}$ for the OpenWebText step-sweep experiment.

Scaling law. To assess whether the benefits of Neural CTMC persist as compute increases, we conduct a scaling-law study on OpenWebText along three axes: model size, training data, and total training FLOPs. We sweep model sizes {tiny (80.1M), small (169.7M), base (424.5M)} and token budgets $\{100\text{M}, 300\text{M}, 1\text{B}\}$, with three random seeds per configuration; the metric is the best validation loss. As shown in Figure 2, the validation loss decreases monotonically with both parameter count and training tokens, and a clear compute-optimal frontier emerges: the best run, `base` at the 1B-token budget, attains a validation loss of 4.150. The fitted power-law exponents on the compute frontier ($\alpha \approx 0.16$, $R^2 \approx 0.93$) indicate that Neural CTMC follows scaling behavior consistent with mature autoregressive and diffusion-based language models, suggesting that the proposed parameterization scales gracefully and that further gains can be expected at larger compute budgets.

Ablation study. We compare Neural CTMC with an R-only variant on OpenWebText to test the effect of learning jump timing and direction with separate heads. Both models use the same uniform forward process, and all other experimental settings are unchanged. The R-only variant directly predicts the complete off-diagonal reverse-rate vector $\{R_t^\theta(i,j)\}_{j\neq i}$; the full model predicts an exit rate $\lambda_t^\theta(i)$ and a jump distribution $r_t^\theta(\cdot \mid i)$, then forms $R_t^\theta(i,j) = \lambda_t^\theta(i) r_t^\theta(j \mid i)$. We do not include an r^θ-only variant with a fixed exit rate because, even under uniform forward corruption, the true reverse exit rate $\hat{\lambda}_t(x_t)$ depends on the current state x_t. Fixing the exit rate would prevent the model from learning this dependence and would misspecify the reverse process. Table 4 reports Gemma 9B PPL as mean $\pm$ standard deviation. The full model lowers mean PPL at all five sampling step counts, with relative reductions of 3.4%–10.6%.

Table 4: Rate-parameterization ablation on OpenWebText (Gemma 9B PPL; mean $\pm$ std, lower is better). **Bold**: lower mean PPL at each step count. Relative reductions are computed from the mean PPL of the R-only variant.

Variant	16	32	64	128	256
R-only (uniform)	617.464 $\pm$10.251	296.667 $\pm$7.374	222.707 $\pm$2.334	211.489 $\pm$1.906	215.839 $\pm$7.456
Full Neural CTMC (λ^θ, r^θ)	**596.250 $\pm$18.507**	**265.181 $\pm$6.327**	**200.394 $\pm$3.306**	**189.205 $\pm$3.541**	**194.956 $\pm$4.384**
Relative PPL reduction	3.4%	10.6%	10.0%	10.5%	9.7%

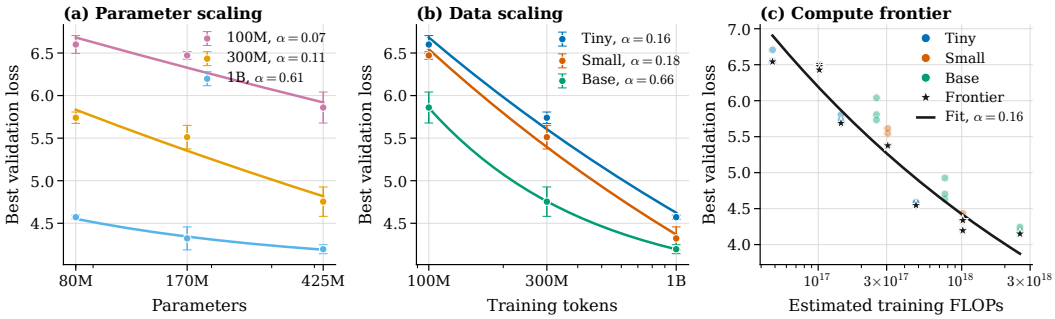


Figure 2: Scaling-law results for Neural CTMC on OpenWebText. Validation loss decreases with model size, training tokens, and total training FLOPs, exhibiting a clear power-law trend on the compute-optimal frontier.

E Self-correction

We additionally consider a lightweight self-correction step that can be applied to a partially denoised sequence during sampling. For sequence data, write $x_t = (x_t^{(1)}, \ldots, x_t^{(L)}) \in \mathcal{S}^L$. The trained model Φ_θ is evaluated on the full sequence and produces the Neural CTMC outputs from (7) at each position.

The step below uses these outputs to recover an auxiliary clean-token distribution and then updates only the most confident proposed correction.

Recovering a clean-token distribution. For the uniform forward process in (2) with $\pi = \frac{1}{S}\mathbf{1}$, the noised distribution associated with a clean-token distribution p_0 has the form $q = \alpha_t p_0 + \beta_t \pi$. At position ℓ, let $i = x_t^{(\ell)}$ and let $\tilde{q}^\theta_{t,\ell}$ denote the model-implied noised distribution recovered from $\Phi_\theta(x_t, t)$. Using the reverse-rate factorization,

$$\lambda_t^\theta(i) \approx \frac{1}{S\alpha_t} \frac{1 - \tilde{q}^\theta_{t,\ell}(i)}{\tilde{q}^\theta_{t,\ell}(i)}, \qquad \lambda_t^\theta(i) r_t^\theta(j \mid i) \approx \frac{1}{S\alpha_t} \frac{\tilde{q}^\theta_{t,\ell}(j)}{\tilde{q}^\theta_{t,\ell}(i)}, \quad j \neq i. \tag{94}$$

Thus we form

$$\tilde{q}^\theta_{t,\ell}(i) = \frac{1}{1 + S\alpha_t \lambda_t^\theta(i)}, \qquad \tilde{q}^\theta_{t,\ell}(j) = \big(1 - \tilde{q}^\theta_{t,\ell}(i)\big) r_t^\theta(j \mid i), \quad j \neq i, \tag{95}$$

and map back to a clean-token distribution by

$$\hat{p}^\theta_{0,t,\ell} = \mathrm{Normalize}\left(\left[\frac{\tilde{q}^\theta_{t,\ell} - \beta_t \pi}{\alpha_t} \right]_+ \right), \tag{96}$$

where $[\cdot]_+$ denotes elementwise clipping at zero. For temperature $\tau_{\mathrm{sc}} > 0$, define

$$\mathrm{Temp}(p; \tau_{\mathrm{sc}})_j := \frac{p_j^{1/\tau_{\mathrm{sc}}}}{\sum_{a \in \mathcal{S}} p_a^{1/\tau_{\mathrm{sc}}}}. \tag{97}$$

Algorithm 4 Self-correction

Require: Partially denoised sequence $x_t = (x_t^{(1)}, \ldots, x_t^{(L)})$; trained Neural CTMC Φ_θ; noise level t; temperature τ_{sc}; maximum updates K
Ensure: Corrected sequence x_t

```
for k = 1 to K do
    for ℓ = 1 to L do
        Let i ← x_t^(ℓ) and compute (λ_t^θ(i), r_t^θ(· | i)) from Φ_θ(x_t, t) at position ℓ
        Recover p̂^θ_{0,t,ℓ} from (λ_t^θ(i), r_t^θ(· | i)) using the equations above
        Sample x̃_0^(ℓ) ~ Cat(Temp(p̂^θ_{0,t,ℓ}; τ_sc))
    end for
    I ← {ℓ ∈ {1, ..., L} : x̃_0^(ℓ) ≠ x_t^(ℓ)}
    if I = ∅ then
        break
    end if
    ℓ* ← arg max_{ℓ∈I} p̂^θ_{0,t,ℓ}(x̃_0^(ℓ))
    x_t^(ℓ*) ← x̃_0^(ℓ*)
end for
Return x_t
```

Quality evaluation. We evaluate the same post-sampling setting on OpenWebText using a common-sample protocol. We first generate 32 raw samples with the Neural CTMC sampler and then apply each self-correction method to exactly the same raw texts. For GIDD, we use the official released checkpoints with $p_u \in \{0.0, 0.1, 0.2\}$ and the self-correction temperatures used in the GIDD study: $\tau_{\mathrm{sc}} = 0.1$ for $p_u = 0.0, 0.1$ and $\tau_{\mathrm{sc}} = 0.5$ for $p_u = 0.2$. For Neural CTMC, we evaluate $\tau_{\mathrm{sc}} \in \{0.1, 0.5\}$ with the same update rule in Algorithm 4. Outputs are judged by GPT-5.2 on a 1–10 scale along five text-quality axes. Table 5 shows that Neural CTMC self-correction is competitive with GIDD self-correction, with the two strongest variants splitting the per-axis wins. Neural CTMC at $\tau_{\mathrm{sc}} = 0.1$ attains the best Grammaticality (4.40, $+12.4\%$, $> 2\sigma$), Factuality (3.09, $+6.2\%$), and

Creativity (5.63, $+8.0\%$, $> 2\sigma$) among all variants, while the hybrid GIDD checkpoint with $p_u = 0.2$ and $\tau_{\text{sc}} = 0.5$ achieves the best Clarity (4.00, $+5.8\%$, $> 2\sigma$) and Writing style (3.88, $+8.4\%$, $> 2\sigma$). The two methods are essentially tied on the five-axis average (4.14 for Neural CTMC at $\tau_{\text{sc}} = 0.1$ vs. 4.13 for GIDD at $p_u = 0.2$), and we note that GIDD ($p_u = 0.2$) shows $> 2\sigma$ improvements on four of the five axes versus two for Neural CTMC ($\tau_{\text{sc}} = 0.1$), so the comparison is best read as complementary strengths rather than a uniform win.

Model	Clarity	Grammaticality	Factuality	Writing style	Creativity
Original sample	3.78	3.91	2.91	3.58	5.21
GIDD self-corr. ($p_u = 0.0, \tau_{\text{sc}} = 0.1$)	3.19 (-15.7%)*	3.01 (-23.2%)**	3.03 (+4.3%)	3.08 (-13.9%)*	4.57 (-12.3%)*
GIDD self-corr. ($p_u = 0.1, \tau_{\text{sc}} = 0.1$)	3.81 (+0.6%)	4.00 (+2.2%)	3.08 (+5.7%)	3.63 (+1.5%)	5.19 (-0.4%)
GIDD self-corr. ($p_u = 0.2, \tau_{\text{sc}} = 0.5$)	**4.00** (+5.8%)*	4.25 (+8.6%)*	2.98 (+2.5%)	**3.88** (+8.4%)*	5.54 (+6.2%)*
Neural CTMC self-corr. ($\tau_{\text{sc}} = 0.1$)	3.81 (+0.6%)	**4.40** (+12.4%)*	**3.09** (+6.2%)	3.77 (+5.4%)	**5.63** (+8.0%)*
Neural CTMC self-corr. ($\tau_{\text{sc}} = 0.5$)	3.82 (+1.1%)	4.06 (+3.8%)	3.01 (+3.4%)	3.74 (+4.5%)	5.57 (+6.8%)*

Table 5: Quality ratings after correcting the same raw OpenWebText samples. Percent changes are relative to the original samples; * denotes $> 2\sigma$ and ** denotes $> 5\sigma$ paired difference across seeds.

Examples. To illustrate what the step changes in practice, we apply self-correction to an OpenWebText sample generated by the Euler sampler with 128 denoising steps. We hold the base sample and correction seed fixed, and vary only the update budget K. Table 6 highlights the edited tokens before and after correction. Small budgets tend to make local grammatical or lexical repairs, while larger budgets allow more aggressive changes; this matches the interpretation of K as a controllable post-sampling correction strength.

Table 6: Self-correction examples for update budgets $K \in \{4, 8, 16, 24\}$. The base sample is generated using Euler sampling with 128 steps.

Example 1: self-correction max_updates=4	
Original sample	**After self-correction**
Martin Samuel, who changed his face because she changed his face, is drawn to profound insight by writer John Lewis.	Martin Samuel, who changed his face because she changed his face, is drawn to profound insight by writer John Lewis.
His first experience with cannabis in 2008 and 2009, and later known as Boone Half the Sole, came at the age of 5, when he returned from high school and ingested a pipe.	His first experience with cannabis in 2008 and 2009, and later known as Boone Half the Sole, was at the age of 5, when he returned from high school and ingested a pipe.
The research program lists marijuana as a federal harm reduction drug, and now it has been legal at a high recreational level, and it has surged in all US states.	The research program lists marijuana as a federal harm reduction drug, and now it has become legal at a high recreational level, and it has surged in all US states.
Samuel is celebrating his return from high school and proposes society deserves to stop using cannabis as an excuse, and the exciting and welcome changes, during brief years of life, that are largely reflected in a drug.	Samuel is celebrating his return from high school and proposes society deserves to stop using cannabis as an excuse, and the exciting and welcome changes, during brief years of life, that are largely reflected in a drug.
Recently President Obama said that cannabis was a “vulnerability” in a single age, to which “I believe a novel and means; all forms of human life should not depend one another on marijuana”.	Recently President Obama said that cannabis was a “vulnerability” in a single age, to which “I believe a novel and means; all forms of human life should not depend one another on marijuana”.
Samuel says, “ rivalry old man, who you known as before, you are a very fond and wonderful old person.	Samuel says, “ rivalry old man, who you known as before, you are a very fond and wonderful old person.
“I suffered a rare [cancer] in extremely good health and I eat with it on the streets - and with my bones, every day it’s very strong.”	“I have a rare [cancer] in very good health and I eat with it on the streets - and with my bones, every day it’s very strong.”
He also is a man, who remembers everything every day, and is pivotal with keeping his life “just a completely ordinary life”.	He also is a man, who remembers everything every day, and is pivotal with keeping his life “just a completely ordinary life”.
Example 2: self-correction max_updates=8	
Original sample	**After self-correction**
Martin Samuel, who changed his face because she changed his face, is drawn to profound insight by writer John Lewis.	Martin Samuel, who changed his face because she changed his face, is drawn to profound insight by writer John Lewis.
His first experience with cannabis in 2008 and 2009, and later known as Boone Half the Sole, came at the age of 5, when he returned from high school and ingested a pipe.	His first experience with cannabis in 2008 and 2009, and later known as Boone Half the Sole, was at the age of 5, when he returned from high school and ingested a pipe.
The research program lists marijuana as a federal harm reduction drug, and now it has been legal at a high recreational level, and it has surged in all US states.	The research program lists marijuana as a federal harm reduction drug, and now it has become legal at a high recreational level, and it has surged in all US states.

Continued on next page

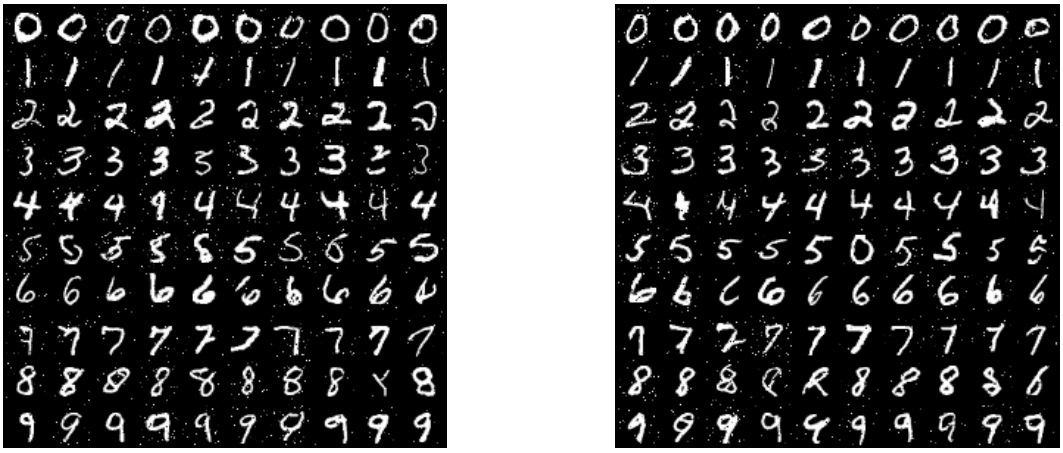
Table 6: Self-correction examples (continued).

Original sample	**After self-correction**
Samuel is celebrating his return from high school and proposes society deserves to stop using cannabis as an excuse, and the exciting and welcome changes, during brief years of life, that are largely reflected in a drug.	Samuel is celebrating his return from high school and proposes society deserves to stop using cannabis as an excuse, for the exciting and welcome changes, during brief years of life, that are largely reflected in the drug.
Recently President Obama said that cannabis was a "vulnerability" in a single age, to which "I believe a novel and means; all forms of human life should not depend one another on marijuana".	Recently President Obama said that cannabis was a "vulnerability" in a single age, to which "I believe a novel and means; all forms of human life should not depend one another on marijuana".
Samuel says, " rivalry old man, who you known as before, you are a very fond and wonderful old person.	Samuel says, " rivalry old man, who you known as before, you are a very fond and wonderful old person.
"I suffered a rare [cancer] in extremely good health and I eat with it on the streets - and with my bones, every day it's very strong."	"I have a rare [cancer] in very good health and I eat with it on the streets - and with my bones, every day it's very strong."
He also is a man, who remembers everything every day, and is pivotal with keeping his life "just a completely ordinary life".	He also is a man, who remembers everything every day, and is pivotal with making his life "just a completely normal life".
Example 3: self-correction max_updates=16	
Original sample	**After self-correction**
Martin Samuel, who changed his face because she changed his face, is drawn to profound insight by writer John Lewis.	Martin Samuel, who changed his face because she changed his face, is inspired to his insight by writer John Lewis.
His first experience with cannabis in 2008 and 2009, and later known as Boone Half the Sole, came at the age of 5, when he returned from high school and ingested a pipe.	His first experience with cannabis in 2008 and 2009, and later known as Boone Half the Sole, was at the age of 5, when he returned from high school and ingested a pipe.
The research program lists marijuana as a federal harm reduction drug, and now it has been legal at a high recreational level, and it has surged in all US states.	The research program lists marijuana as a federal harm reduction drug, but now it has become legal at the high recreational level, and it has surged in all US states.
Samuel is celebrating his return from high school and proposes society deserves to stop using cannabis as an excuse, and the exciting and welcome changes, during brief years of life, that are largely reflected in a drug.	Samuel is celebrating his return from high school and proposes society deserves to stop using cannabis as an excuse, for the exciting and welcome changes, in brief years of life, that are largely reflected in the drug.
Recently President Obama said that cannabis was a "vulnerability" in a single age, to which "I believe a novel and means; all forms of human life should not depend one another on marijuana".	Recently President Obama said that cannabis was a "vulnerability" in a young age, to which "I believe a novel and means; all forms of human life should not depend one another on marijuana".
Samuel says, " rivalry old man, who you known as before, you are a very fond and wonderful old person.	Samuel says, " rivalry old man, who you known as before, you are a very fond and wonderful old person.
"I suffered a rare [cancer] in extremely good health and I eat with it on the streets - and with my bones, every day it's very strong."	"I have a rare [cancer] in very good health and I eat with it on the streets, and with my bones, every day it's very strong."
He also is a man, who remembers everything every day, and is pivotal with keeping his life "just a completely ordinary life".	He also is a man, who remembers everything every day, and is pivotal in making his life "just a completely normal life".
Example 4: self-correction max_updates=24	
Original sample	**After self-correction**
Martin Samuel, who changed his face because she changed his face, is drawn to profound insight by writer John Lewis.	Martin Samuel, who changed his face because she changed his face, is inspired to his insight by writer John Lewis.
His first experience with cannabis in 2008 and 2009, and later known as Boone Half the Sole, came at the age of 5, when he returned from high school and ingested a pipe.	His first experience with cannabis in 2008 in 2009, and later known as Boone Half the Sole, was at the age of 5, when he returned from high school and ingested a pipe.
The research program lists marijuana as a federal harm reduction drug, and now it has been legal at a high recreational level, and it has surged in all US states.	The FDA still lists marijuana as a federal harm reduction drug, but now it has become legal at the high recreational level, and it is legal in all US states.
Samuel is celebrating his return from high school and proposes society deserves to stop using cannabis as an excuse, and the exciting and welcome changes, during brief years of life, that are largely reflected in a drug.	Samuel is celebrating his return from high school and says society deserves to stop using cannabis as an excuse, for the exciting and welcome changes, in brief years of life, that are largely reflected in the drug.
Recently President Obama said that cannabis was a "vulnerability" in a single age, to which "I believe a novel and means; all forms of human life should not depend one another on marijuana".	Recently President Obama said that cannabis was a "vulnerability" at a young age, to which "I believe a novel and means; all forms of human life should not depend one another on marijuana".
Samuel says, " rivalry old man, who you known as before, you are a very fond and wonderful old person.	Samuel says, " rivalry old man, who you known as before, you are a very fond and wonderful old person.
"I suffered a rare [cancer] in extremely good health and I eat with it on the streets - and with my bones, every day it's very strong."	"I have a rare [cancer] and very good health and I eat with it on the streets, and with my bones, every day it's very strong."
He also is a man, who remembers everything every day, and is pivotal with keeping his life "just a completely ordinary life".	He also is a man, who remembers everything every day, and is pivotal in making his life "just a completely normal life".

F Generated Samples

We present generated samples from Neural CTMC, shown without post-processing.

F.1 Image Generation



(a) Euler (128 steps) (b) τ-leaping (128 steps)

Figure 3: Conditional MNIST samples generated by Neural CTMC at epoch 80 with 128 sampling steps. Each row corresponds to one digit class (0–9), with 10 samples per class.

F.2 TinyStories

Sample 1. One day, a girl named Amy went to the park to play. She saw a van full of yummy snacks. Amy's friend Tom wanted to buy some from the store. Amy and Tom went to the store, but they didn't sell the snacks. There were still many things, and Amy was sad. Then they saw a lady in the park. She saw how sad Amy was. The lady said "Amy, why don't you sell the snacks?" Amy thought for a moment and said, "Let's tell the other boy to ask." They walked to a boy and said, "We did this in the shop. Can we buy the food?" The boy sold the food and Tom was happy. He said "thank you, Tom!" Amy smiled and said, "You're welcome Tom! We shared a story and sometimes it's good to tell questions, it can make others happy." Amy and Tom became the best of friends.

Sample 2. Once upon a time there was a stubborn little boy. He liked to play all day, but one day he found a whistle. He decided to pick it up and blow it. He wiggled around and his head started to blow a tune. He was so happy and he was having so much fun. He was playing when a wise old owl came along. The owl asked the whistle: "Dumb, you have some music coming from when you work up close. Watch the practice". The stubborn little boy listened as the owl had said earlier. He just listened to the music and then opened his mouth and blowing it in the air. The whistle was so happy he could whistle louder, but eventually he started to forget how stubborn he was.

Sample 3. Lily and Tom are at the park with their mom. They see a slide and want to go on the big slide. So they take turns to go first. Lily goes first and sits on the slide fast. She says, "Look, I did it, I had fun!" Tom slides down again and again. Lily is sad and angry. She walks to the bench and says, "Take it back! The slide is not fair with all the fun!" But Tom says, "No, mom! Go away, this is my slide!" Mom sees what is happening and comes to the bench. She sees Lily and Tom fighting. She is not happy. She says to them, "You are naughty kids. You cannot play on the slide. You have to ask before you play. The slide is a park, it is a place to play. You have the chance." Lily and Tom feel sorry. They say, "I'm sorry, mom. I was wrong. We will be nice next time. We will wait for you and you will be happy." They decide to go on the slide together. They climb the steps slowly. They make it to the top. Mom smiles. She says, "Good job, kids. You are brave. Let's go home now. Lunch time." Lily and Tom hug mom. They are happy again.

Sample 4. Once upon a time, there was a boy called Timmy. Tommy loved to play outside, so he loaded rocks with him. Tommy especially loved playing in the sand when he was outside. One night, Timmy got so light, he fell asleep by the sea. He ran to the beach to take his shovels in the sand. When he came to the beach, though, he encountered fear. He saw a alive little puppy and that it seemed to be stuck in the sea. Timmy's heart sank and ran with his shovel to help the pup. He

worriedly dug free the little pup, as it couldn't breathe properly. The little puppy wagged its tail and was looking back at Tommy in his eyes. From then on, Timmy never wanted to go back to the beach. He wanted to play and have a new furry friend to explore. The End.

Sample 5. Once upon a time, there was a little girl who liked to touch things. She had a lot of questions and always wondered where she could go because she liked to explore nature. One day in the forest, she came across something very fragile. It was a small button and looked very fascinating. She touched it out and realized it hurt her. The little girl looked around and saw unknown things that she had never seen before. She was so excited as she touched it to show her friends. She quickly ran back home and proudly showed them the fragile button. Her one of her friends said, "That is amazing! That looks fragile. We should keep extra care in the future." The little girl smiled and said "I know, what friendship will come with?" Just that, the little girl and her friends had a wonderful time in the forest!

Sample 6. Once upon a time there lived a girl called Mia. She was three years old and had big ears. All the little kids near the beach, Mia liked to whistle with her lip in the sandpit. One day, Mia put on her swim suit and went to the beach. She looked around and saw a little box full of toys in it. Mia picked up the toys and tried to shout. Nothing she could hear, Mia tried and tried, but she couldn't. Then, Mia heard a noise behind her. She turned and saw a man in a funny hat was walking towards her. He said, "hello Mia!" Mia was embarrassed but she decided not to be scared, and she managed to pay down the sound of whistle. The man came out of the car and approached them both. Mia said, "I'm Mia". The man smiled and said, "I'm Henry". Mia's mom was so happy to meet Mia and the man. They all became friends and had a lot of fun. Mia even bought a new toy. Mia smiled and waved goodbye as the man left, feeling Mia would never forget her day.

Sample 7. Jack woke ready today one morning. He put on his hat, and on his lovely shoes. He grabbed his lunch bag because he knew he was going. At the park, Jack saw a big tank that he had never seen before. He was so excited and he ran over to see the tank, but when he got the tank down in front of him, he was very sad. He saw worms, bugs, and shout. But then a friendly voice shouted, "Don't worry Jack. I can help you". Jack looked up and saw a friend. It was his friend John. John said, "Let's go explore the tank together!" So they went in, and Jack was so happy. Jack and his friend explored together all night and eventually they discovered a big aquarium. They played games together and saw fish swimming in the tank. Jack was so happy. Jack started it was time to go home, but instead he said "I want to be friends?" John just smiled and hugged his best friend. They both ran back to the park and they had a great day.

Sample 8. Lily and Ben were twins and liked to play outside with their toys. They had a red cart that they always filled with tape and toys. They liked to make roads and houses with blocks and pretend they were lions and horses. One day, they decided to go with their cart to the park. They asked their mom to let them have fun. She said yes and let them. They pushed their cart on the road, holding hands, jumping and laughing. Ben was driving the cart, but Lily was faster. She saw that their cart had a big stick that had been cut off. She tried to cut it, but it was too tight. She pushed it off and ran away. Ben said that was not fair and ran after it. They both took the cart with all their might. But they did not see a big puddle of mud behind. They did not hear how mom was coming to pull them out. Their mom was not happy. She made them wash the mud and the cart. She told them to take away all their toys and go home. She told them to say sorry and share their toys, and never to play with their cart for a week. Ben and Lily felt sad and sorry. They realized they did not mean to be polite. They wished they had never fought over the stick again. They wished they had been more polite.

Sample 9. Once upon a time there was a bear and a rabbit in their den in the forest. In the forest there was a big cake. The bear and rabbit were curious about the cake so they decided to go. Slowly, they bounced and skipped on their way. They were starting to feel very happy. Soon they realized they were in a place to get to the cake. "Where are we going?" shouted the tree. The rabbit and bear looked around. Suddenly a wise voice said back, "Why won't you return to a group of humans? They will compete for the best cake at the competition!" The bear in the forest cheered. The bear was so excited and wanted to join the competition. Together they went on the stage and cheered as they won.

Sample 10. Once upon a time there was a curious frog. He lived in a green and peaceful forest. One day, the frog decided to go for a swim. He went to the pond but his body settled down the water.

He liked to see the water and decided to take a closer look. But afterwards all the way closer, he came across a keeper that was causing the water. The keeper didn't like having the water frog along the pond so he agreed to arrest him. The frog was sad but he was determined to give himself a try. He jumped and splashed with the animals for a long time. Suddenly, something wonderful happened. The gust of wind blew very hard and the little frog was blown away from the pond. The frog was so happy and quickly swam away. The other animals watched him to drink the pond from above, happy he had helped his friends.

F.3 OpenWebText

Sample 1. Let's examine Golden State's next in this year-end battle: allowing people aspiring to buy health-care policies from companies to enjoy access to one for everyone. The commission whose leading role in acquiring health insurance is considering that an exchange delivered from state over state to Golden State be moved to the new opportunity to compete to policies denied consumers by government — resulting in the health insurance industry's loss of interest — through 2017. While that leaves Sacramento residents looking for an option, a free, tangible savings account appears insufficient because they don't think it could help any of them. "If the general democracy was against it, we would be reluctant to get our health insurance in that state and go through that state," Tourer said. Now that leaves an alternative: "Making sure you don't out and buy insurance and anything about our state we love is equity and fairness and everything else, now we have to decide whether not for you and you want to have subsidies and or not," he said. More close-to-your-time credit or financial security is what the current system does. But that financial stability has recently led to many types of miners copying and rewriting health insurance policies, as well as an increasing interest in the markets becoming the norm. That, unlike some things in the health care markets, are terms that certainly aren't too complicated in fact. The federal government can explain why and how it changed years ago. Of course, the new Consumer Financial Protection Bureau report provided that some of the support changed came from the same agency that had commissioned a statewide campaign to force the 10-percent health care burden toward poor and rich Southern California. It argues that it means that the bottom line comes in with those of our new neighbors.

Sample 2. After taking off a plane Monday morning, Alex Campbell met with the coach himself Monday night, as the Flames haven't played under Campbell this year. In addition to autographs by former Flames-line player and right-arm captain Alexander Grad (who was the Stanley Cup team's affiliate with hockey team Arizona last year) we've taken a look at the life of city's top scorer, who moves the third straight. "I think we end up a pretty good overall record in a team. Really here, during that period when people come to a wedding or dinner, who take in season trophies, he does play remarkably well overall," said Westberg, who is coach in the North Central Lakes Helacs. Former GM Simon Keller spent an extended 20 years in the NHL but has been slowly to build into a guy as an 18-year-old 13 years. He's pretty well engaged and has a good story. He began to really early grow as a person, and was definitely a contributor in a few of them before he went to the Devils in 2004. That fierce sense of humor is useful to those in our organization as well as him. "That's cognac stuff. And we are a great, centered person. It needs some energy and variety, he's great. And he's supposed to be a proper player. But we might deliver some tactical stuff."

Sample 3. As Japan's tech industry has coalesced their startup talent and created so much future there will be computer generated labor for entrepreneurs to do in Japanese. Concern among those around the country is about how best to train workers to complex tasks such as to for goods during limited shifts. For some resigned, tech employees in this sector have been pushing to increase the labor base to more than expected, which could help boost corporations' share of corporate workers from a projected low 600,000 a year, according to USA Today, with some regulations to protect businesses from worker making human workforce. Long-time automation — as a result, the controversy surrounding an AI company by the Open House artificial intelligence Institute called Magnii, based on a capitalists' vision, has been heavily development by Nune Labs and its Tokyo headquarters. Though Tokyo, with its 11 million population, features the country's top job market, over a million Japanese entrepreneurs lack any longer work. Nvo Labs had a disparate interest in AI that led to the head of police department Yumsuki backing away with Niigueu in fall of 2013. The Lab has committed to introducing autonomy and run AI development programs at warehouse laboratories in villages, comparable to an ITTA cooperative. She also continued to pursue efforts to install local government staff that can better handle high-end robots, like the AITs.

Sample 4. Chief thing here today from my employer, Barlan Yblin. You've got to be the newest newcomer to the limelight, Bernie Sanders. I've long been paying attention to the Sanders' argument too much. As CNBC reports, the American economy has lost roughly $3.6 million worth of jobs per month in June, or $37 million — making up the rest, such as child care and childcare for mental veterans. Sanders seems to be not making claims that have put the economy "under far greater strain." Case: In 2009, since is fourth year, a data center report showed the net worth of all U.S. households had plunged. The report insisted that in 2009, the entire economy lost roughly $1 trillion more. This was down by a little over a tenth of a share compared with five years. For comparison to the past, incomes were consistently diminished with little improvement in value, and earnings analyses found no improvement. But jobs seem to have seemed to have a great impact. Ultimately, any company looking to hire workers in this is notoriously insubordinate. Now the Washington Post's library of political papers has covered none of this. We mention that Mr. Sanders has said that many companies lowered their payloss because of the country's improving economy and the currently difficult times.

Sample 5. For the past decade, Google has attempts to take the leap in building up a truly free WiFi-only hardware platform given that it lacks single-party cross-network support, and that its OS will allow all to get a full unexperimental operating system in the modern day world through an incredibly revolutionary privacy approach. "We woke up the OEM crowd with maybe 1 chance so they realise that Google's not afraid of it," explains Johannes Systant, vice-president of GTH. This isn't really too much different for what hardware Google, though, are trying to do. Cross-network devices are the first to lead connected platforms in terms of improving our devices by allowing all computers to interact with data, talk and play. Android is part of many a very powerful advancement of the Internet through the same way that Chromebooks pick up their favourite Android app for free. Google is basically plugging the Internet into the plug out of incorporate common USB devices. The key for Google to deliver on its traditional model is where networks and handhelds succeed. Using this technology, you can comfortably control a USB device to a network device to connect a USB DN port, or other abstractly or remotely, to the network. Through open standards, paired between two USB ports, you can securely connect your Ethernet device connected to a cryptographic service and from any USB port. Using such devices, you'll be able to encrypt your computer's pointing code and transfer that data right out into your home computer's additional encryption chip.

Sample 6. Down on Labor? The Radical Coalition says "came cut off" from Federal Broadcasting Commission investigation. A UN conference urging Labor to join an inquiry into violations of public broadcasting freedom has turned up and out. The inquiry's focus is an ongoing effort to examine the internal climate science submissions debate that led to the Rudd government's denial of access to the Danish and Dutch Science Centre (EK) in the Delta area. It gives retreat access to report to be chaired by Mike Smith, UN wildlife director, the morning of a press conference. It was revealed former Labor backbencher Gordon Bensonberg, from the WA Liberals and now the head of the environmental campaign of the Greens, is in step wrong for the activities of a UN inquiry into the practice. "As I reported, there is a regime in the Australian people have so far known that the issue of using this column to a UN investigation regarding the conversations between the EK and the Australian organisation was — a legally relevant issue, that the land owners were not worthy to answer," Bensonberg told a hallway questions account. Senior members of the committee of the inquiry were not immediately available to comment for this report. The United Party of Labor spokesperson, Elizabeth Raffre, told the ABC the group would not join the UN inquiry: "Currently, we are carrying out what they deal with at that, but we know that's not a part of investigation that they'll get to us." The conference stressed it had not been told how the Radical Coalition gathered for the human rights briefings in September.

Sample 7. Chinese Super League Poland have won for the first time in a World Cup. The Lions are expected to win this past Champions International in 2010 after three successive competitions. Clant-Club City and Cardiff were bracing for plenty after a disappointing losing performance at the 2015 Cup. Nevertheless, with the on-and-out post-season August 1 International, them should have the opportunity to compete. The World Championship match will no doubt irritate the young squad. However a huge performance is not always the answer, however who knew it was an unsuccessful visit to England. With many hoping there is a red-and-red affair aimed at Tony Bois, it's unlikely that there will. The players do have plenty of prying over this fixture, though it should fit the eye better for Yi Xiaakis and the rest of the emerging squad. Speaking after this clash, Tony Bois told Daily Football that "it can't be enough ambition". The 30-year-old believes that now with China the

organisation is listening. He couldn't force the fight after the court in a commercial clash. Although China is Poland's problem, Hoo still has a motivation that his team will get better. They have won a lot in a little over three seasons at the top, and I will defend another great old team.

Sample 8. WASHINGTON — The Huffington Post recently reported that Connecticut Arms Owners Protection (DFA) background checks are necessary for gun purchases related to the Newtown shooting, leading to law-abiding pistol-regulators and killing people. Per the report on Friday, government-sponsored gun legislation, employing a concealed weapon legislation that protects gun transactions and convictions for possessing the concealed weapons, had by 2013 violated federal and legal authority for concealed weapon holders. In 2013, when DFA along with additional gun restrictions went on the books, the power of state lawmakers and manufacturers to manufacture and prohibit concealed weapon holders from carrying guns for personal use would create a precedent for the act of mass murder of someone fatally using a gun or a handgun, as well as committing a killing for dealers and other gun dealers. On Saturday, The Washington Post came out with the news, carrying an account of the shooting of 20 people killed on Boston streets. It is not clear whether the shooting chose to use a federal TPP agreement to bring concealed handguns outside the U.S. after Newtown. The story did not attract concerned attention. It should have become clear that the story is either the whole story or, with its many parts, however, it was a first-hand experience that would have sparked outrage among only the front organs of the newspaper.

Sample 9. In northeast Wisconsin's second Congressional district, Rep. Frank Miller (R-WI) was making out to reporters in a pricey real estate section of Bloomberg headquarters. He called himself a physicist — like the humble servant that found iron Aug. 25 on Jay Jones and addressing his fans at the CEO Leadership Leaders Summit in San Francisco. The clue is silly and incomplete, intended to unify conservatives with a 2016 election season over the horizon. It bears out a lot of never-ending confusion about whether the president's plan, which would impose new corporate income taxes, is simply flying under the radar. The stuff hasn't changed and it's a make-believe policy package. It doesn't necessarily matter because you are an investment banker. In the way the plan specifically considers, you're the one who doesn't pay income tax. But in the wake of last month's results, the president has not proposed a taxcut, but it's to promote the dream of personal deductions. And the bottom line: you won't get an income-cut for millions of people. Miller's website has a similar message, but in reality, it helps the people who don't understand it understand the roots. The billionaire businessman's brand rests on revenue and its business model — not Bloomberg, for three local newspapers.

Sample 10. New York J Committee Giants Nats Jets 10-3 Set To Play. That's the common sense for the Jets offense, common with a defensive powerhouse feeling out after making an unexpected draw in overtime against the Giants. Despite a healthy quarterback that was named doubtful for an 80-plus-point rally — the Jets had an outstanding defensive performance in the home defeat. The Jets go on a limb even if it can be said that the Jets pushed offenses into the brawl after the trade of Charlie Mario Bisolo near the end of the team's game. "I'll be here tonight," Mike Gonzalez, the Green Bay and defensive tackle's overall skill, told a Giants coach. But then something changed. "The problem with that, we go to make sure you can get true wins, you can have interceptions at all times," Gonzalez said. The Jets will need more aggressive defense when sometimes and power are paid. "Without reaching the distance, you'll pretty much never be playing," Gonzalez said. "But I think, they've always had their defensive players tell me, they can't give you a challenge. No, I said, you can't score if you have no, you can't be good, you cannot be effective from the line I got on the edge."

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The three claims in the abstract and Section 1—decomposed exit-rate/jump-distribution parameterization, ELBO factorization into a Poisson KL and a categorical KL, and competitive performance of the pure uniform forward process on TinyStories and OpenWebText against mask-based baselines under matched training budget—are supported by Section 4.1 (theory) and Section 5 (experiments).

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 7 discusses three limitations: (i) scale—the largest model trained is 424.5M parameters with a 512-token context; (ii) post-training and RL are not studied; (iii) we only train and evaluate the pure uniform forward process, while general mixture forward processes are unexplored.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All theorems, lemmas, propositions, and corollaries (e.g., Proposition 4.8 explicitly lists assumptions (i)–(iii)) state their assumptions, and full proofs are given in Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The architecture (12-layer DiT, 768 hidden dim, 12 heads, 163M params), training configuration (AdamW, learning rate, batch size, schedule, bf16), and sampling pseudocode (Algorithms 2 and 3) are fully specified in Section 5 and Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: We release the trained checkpoints together with sampling code, which is sufficient to reproduce the generative perplexity numbers in Table 1; however, the training code is not released at submission time. The architecture and training procedure are nevertheless fully specified in Section 5 and Appendix D to support reproducibility.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Optimizer (AdamW), learning rate (5×10^{-4}), cosine schedule, batch size, training steps, mixed-precision setting, and the OpenWebText evaluation protocol (five sampling seeds, 1,024 samples of 512 tokens per seed, scored by Gemma 9B and GPT-2 Large) are reported in Section 5 and Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: For the OpenWebText PPL comparisons, we report the mean and sample standard deviation of the five per-seed PPL values. Each model generates 1,024 unconditional samples of 512 tokens per seed, and Gemma 9B and GPT-2 Large score the same samples. The reported standard deviations capture sampling variability from the initial distribution and categorical transitions at a fixed checkpoint. The scaling-law study (Appendix D, Figure 2) uses 3 random seeds per configuration, with error bars across seeds shown in the figure, and the compute-optimal frontier is fit with a power law of $R^2 \approx 0.93$.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We used 8×H100 (80GB) GPUs for training. The Neural CTMC checkpoint evaluated on OpenWebText was trained on approximately 131B tokens, as reported in Table 1; the batch size and training-step count are given in Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research uses publicly available datasets (TinyStories, OpenWebText) and trains a small-scale generative model; it does not involve human subjects or sensitive data, and conforms to the NeurIPS Code of Ethics.

   Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This is foundational research on a training objective for discrete diffusion models; the released 163M-parameter checkpoint produces text quality clearly below comparable open-source language models, and we therefore do not foresee societal impacts beyond those generally associated with small open-source language models.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: The released checkpoints are distributed with a license and explicit terms of use restricting redistribution and specifying acceptable use.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the official sources of SEDD [17], MDLM [24], and GIDD [31] (whose released code and checkpoints we use under their original licenses), as well as Gemma2-9B (used as the external scoring model), TinyStories [7], OpenWebText [23], the GPT-2 tokenizer, and the DiT architecture [22].

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The released Neural CTMC checkpoints are accompanied by a README, sampling configuration files, and a license describing usage terms.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve research with human subjects, so IRB approval is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: We use pretrained Gemma2-9B to compute generative perplexity on TinyStories and OpenWebText, and GPT-2 Large to compute generative perplexity on the same OpenWebText samples (Section 5 and Appendix D). We also use GPT-5.2 to rate self-corrected OpenWebText samples on five text-quality axes (Appendix E). These models are used only for evaluation and are not components of the proposed method.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.