DOES UNIFORM DISCRETE DIFFUSION NEED TIME?

Anonymous authors
Paper under double-blind review

ABSTRACT

Uniform discrete diffusion models (UDMs) commonly use explicit time conditioning, but we find that it can often be unnecessary in practice. In this paper, we first show that the population-optimal UDM predictor generally depends on time: time controls how much the model should trust the observed context. We then show that this dependence can become negligible in finite-data settings relevant to language. When a corrupted training sequence remains much closer to its original clean sequence than to competing training sequences, the empirical-optimal predictor is nearly insensitive to time over most of the diffusion trajectory, where the guarantee weakens toward the high-noise endpoint. Empirically, trained language UDMs exhibit limited time sensitivity over most of the trajectory, while time-agnostic predictors remain competitive with, and often outperform, time-conditioned models across datasets and training objectives. These results challenge the use of explicit time conditioning in UDMs: although the population optimum depends on time, explicitly conditioning on it may often be unnecessary in practice.

1 INTRODUCTION

Diffusion models usually tell the neural network how noisy its input is (Ho et al., 2020; Song et al., 2021; Lai et al., 2025). This information, commonly provided through *time conditioning*, allows the same model to make different predictions at different stages of denoising. But does the model actually need to be told the time? Recent work on continuous diffusion suggests that explicit time conditioning may not always be essential. Sun et al. (2025) show that removing time conditioning causes only a small performance degradation in several settings. Separately, Wang & Du (2025) develop a generative formulation whose neural model is time-independent, further showing that generation need not always be built around an explicitly time-conditioned predictor.

For discrete diffusion models, however, the answer depends strongly on the corruption mechanism. In masked diffusion models (MDMs), corrupted tokens are replaced by a special `[MASK]` symbol. The corrupted sequence therefore reveals which positions have been corrupted, and the population-optimal clean-token predictor does not need the diffusion time explicitly (Zheng et al., 2025). Consistent with this property, time-agnostic MDMs achieve comparable empirical performance (Sahoo et al., 2024) and have been adopted in large language diffusion models (Nie et al., 2025; Ye et al., 2025).

Uniform diffusion models (UDMs) are fundamentally different. A corrupted token is replaced by an ordinary vocabulary token sampled from the uniform distribution, so the model cannot directly tell whether an observed token was preserved from the clean sequence or produced by noise (Liu et al., 2025; Amin et al., 2025). The diffusion time can therefore carry information that is absent from the corrupted sequence itself, by indicating how much the observed context should be trusted. This makes it unclear whether the time-agnostic property of MDMs can extend to UDMs.

This question has become increasingly important as UDMs scale to large language modeling. Recent studies show that scaled UDMs can outperform MDMs in several settings (Sahoo et al., 2026b; von Rütte et al., 2026), followed by the development of large language UDMs (DiffusionGemma Team, 2026; Sahoo et al., 2026a). Some of these models already use time-agnostic predictors, yet it remains unclear why this works, when time actually changes the optimal prediction, and how much explicit time conditioning helps in practice.

Throughout this work, we call a UDM *time-agnostic* when its neural predictor does not receive diffusion time as an explicit input. The forward corruption process, noise schedule, and other analytic time-dependent quantities remain unchanged. We ask:

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

What role does time play in UDMs, and when do UDMs need explicit time conditioning?

First, we characterize why and how the population-optimal UDM predictor depends on time. Uniform corruption creates uncertainty about which observed tokens are reliable. Consequently, every subset of observed tokens constitutes a possible hypothesis about the reliable context. We show that time only controls the relative weights assigned to these hypotheses. Thus, unlike MDMs where corrupted positions are explicitly identified, the population-optimal UDM predictor is generally time-dependent, but its dependence on time has a simple and structured form.

Second, we show that finite training data can strongly suppress this dependence. Population-level time dependence arises from reweighting competing hypotheses about which observed tokens are reliable. In finite language data, however, the enormous discrete sequence space leaves training examples highly separated. A corrupted sequence therefore remains much closer to its originating clean sequence than to competing sequences, causing the posterior to collapse onto few hypotheses and weaken time-dependent reweighting. Only near the high-noise endpoint, where multiple hypotheses become plausible, can time affect the prediction. We make this intuition quantitative, deriving a bound that links the time sensitivity of the empirical oracle to the post-corruption separation margin. We empirically verify that language data exhibit large separation over most noise levels.

Finally, we find that learned language UDMs make limited use of explicit time over most of the diffusion trajectory. The above finite-data result concerns the empirical oracle, not a learned neural network. A learned model may recover additional time dependence through generalization or architectural inductive bias; we therefore separately analyze trained UDMs. Their measured time sensitivity is weak over much of the trajectory and becomes substantially larger in the high-noise regime. The predictive benefit of time conditioning is similarly concentrated near high noise. A hybrid model that enables time conditioning only in this regime performs nearly the same as a fully time-conditioned model, while fully time-agnostic UDMs remain competitive with, and in several settings outperform, their time-conditioned counterparts across datasets and training objectives.

In summary, our results shift the question from whether time matters in UDMs to *where it matters enough to justify explicit conditioning*. This helps explain why time-agnostic UDMs can work well despite a time-dependent population optimum, and suggests a broader design space for simpler architectures, training methods, and samplers that use time only where it is useful.

2 BACKGROUND

We briefly introduce the UDM formulation and the recent variants used in this work. See Appendix A for more details and related work.

Notation. Let $\mathcal{X} = \{\boldsymbol{e}_1, \dots, \boldsymbol{e}_K\} \subset \Delta_K$ be a finite token space, where each token is represented as a one-hot vector and Δ_K is the K-category probability simplex. The vocabulary size is $K = |\mathcal{X}|$, and the uniform prior is $K^{-1}\mathbf{1}_K$. A sequence is $\boldsymbol{x} = (\boldsymbol{x}^1, \dots, \boldsymbol{x}^L) \in \mathcal{X}^L$, and $\boldsymbol{z}_t$ denotes its corrupted version at time $t \in [0, 1]$, where $t = 0$ is data and $t = 1$ is the uniform prior. α_t denotes a decreasing noise schedule with $\alpha_0 \approx 1$ and $\alpha_1 \approx 0$, and $\rho_t \coloneqq K\alpha_t/(1-\alpha_t)$. For simplicity, we use the standard $\alpha_t = 1 - t$ hereafter; formal theoretical results for general schedules are provided in Appendix where the distinction matters. Unless stated otherwise, expectations are over $\boldsymbol{x} \sim p_{\text{data}}$ and $\boldsymbol{z}_t \sim q_t(\cdot \mid \boldsymbol{x})$. $\langle \cdot, \cdot \rangle$, $\odot$, and $\mathbb{1}\{\cdot\}$ denote the dot product, Hadamard product, and indicator function, respectively.

Uniform diffusion models. A UDM independently corrupts each token toward the uniform prior:

$$q_t(\boldsymbol{z}_t^i \mid \boldsymbol{x}^i) = \text{Cat}\left(\boldsymbol{z}_t^i; \alpha_t \boldsymbol{x}^i + (1-\alpha_t)K^{-1}\mathbf{1}_K\right). \tag{1}$$

For $s < t$, the single token posterior $q_{s|0,t}$ follows from Bayes' rule using the $s \to t$ transition and the s-marginal (Eq. (10)). Schiff et al. (2025) parameterize the reverse process by plugging $\boldsymbol{x}_\theta^i(\boldsymbol{z}_t, t)$ into this posterior,

$$p_{s|t}^\theta(\boldsymbol{z}_s^i \mid \boldsymbol{z}_t) \coloneqq q_{s|0,t}\left(\boldsymbol{z}_s^i \mid \boldsymbol{x}^i = \boldsymbol{x}_\theta^i(\boldsymbol{z}_t, t), \boldsymbol{z}_t\right),$$

as detailed in Eq. (3). The standard UDM is trained with the continuous-time NELBO:

$$\mathcal{L}_{\text{UDM}}(\theta) = \int_0^1 \mathbb{E}_{\boldsymbol{x}, \boldsymbol{z}_t} \sum_{i=1}^L \frac{-\alpha_t'}{K\alpha_t} \left[\frac{K}{\bar{x}_{\theta,\ell_i}^i} - \frac{K}{\bar{x}_{\ell_i}^i} + \sum_{j \neq \ell_i} \frac{\bar{x}_j^i}{\bar{x}_{\ell_i}^i} \log \frac{\bar{x}_{\theta,\ell_i}^i \bar{x}_j^i}{\bar{x}_{\theta,j}^i \bar{x}_{\ell_i}^i} \right] \mathrm{d}t, \tag{2}$$

where $\bar{\boldsymbol{\nu}}^i \coloneqq K\alpha_t \boldsymbol{\nu}^i + (1-\alpha_t)\mathbf{1}_K$ and ℓ_i denotes the non-zero index of $\boldsymbol{z}_t^i$.

Recent UDM formulations. Among several recent UDM formulations, we focus on two representative frameworks: Diffusion Duality (Sahoo et al., 2025) and the leave-one-out formulation of Gourevitch et al. (2026). Diffusion Duality retains the same uniform forward process and UDM NELBO, but represents the uniform-state marginal through the $\arg\max$ of a Gaussian corruption and uses a curriculum from continuous relaxations to discrete inputs. In contrast, Gourevitch et al. (2026) show that the neural predictor in the UDM bridge has the *leave-one-out* (LOO) posterior as its population optimum: each clean token is predicted from the corrupted context excluding the observation at that same position, see Eq. (4). They further derive an analytic local-token correction that converts the LOO predictor into the full denoising posterior, allowing the underlying predictor to be trained with a simple cross-entropy loss. We refer to this CE-trained formulation as LOO+CE.

3 The Role of Time at the Population Optimum in UDMs

The corruption mechanism determines whether time needs to enter the clean-token predictor. In MDMs, the special mask symbol reveals which positions have been corrupted, so the population-optimal predictor can rely directly on the visible tokens and is time-agnostic (Zheng et al., 2025).

UDMs are different. Because uniform corruption replaces a token by another ordinary vocabulary token, the model cannot directly tell whether an observed token was copied from the clean sequence or produced by noise. Time can therefore be informative because it changes how likely the observed context is to have been preserved from the clean sequence. This motivates two questions:

1. Is the population-optimal UDM predictor inherently time-dependent?
2. If so, through what structure does time affect its prediction?

The first determines whether removing explicit time conditioning changes the population optimum; the second reveals what information time actually contributes.

3.1 The Population Target of the Uniform Channel

The UDM reverse kernel has a seemingly complicated plug-in form (Gourevitch et al., 2026; Sahoo et al., 2025; Schiff et al., 2025):

$$p^{\theta}_{s|t}\big(\boldsymbol{z}^i_s \mid \boldsymbol{z}_t\big) = q_{s|0,t}(\boldsymbol{z}^i_s \mid \boldsymbol{x}^i = \boldsymbol{x}^i_\theta(\boldsymbol{z}_t, t), \boldsymbol{z}_t) = \mathrm{Cat}\left(\boldsymbol{z}^i_s; \frac{\rho_t \boldsymbol{z}^i_t \odot \boldsymbol{x}^i_\theta(\cdot,t) + \frac{\rho_t}{\rho_s}\boldsymbol{z}^i_t + \big(1-\frac{\rho_t}{\rho_s}\big)\frac{\rho_s \boldsymbol{x}^i_\theta(\cdot,t)+\mathbf{1}_K}{K+\rho_s}}{\rho_t \big\langle \boldsymbol{z}^i_t, \boldsymbol{x}^i_\theta(\boldsymbol{z}_t,t)\big\rangle + 1}\right). \tag{3}$$

Importantly, all neural-model dependence enters through the clean-token prediction $\boldsymbol{x}_\theta(\boldsymbol{z}_t, t)$. Thus, rather than analyzing the reverse kernel directly, we ask what this predictor should output at the population optimum and how that prediction depends on time.

Gourevitch et al. (2026) show that the neural prediction used by the UDM bridge has a *leave-one-out* (LOO) posterior as its population target. Fix a target position $i \in [L]$. Let $X \sim p_{\mathrm{data}}$ denote a random clean sequence, $T \sim \mathrm{Unif}([0,1])$ the corruption time, and $Z \mid (X = \boldsymbol{x}, T = t) \sim q_t(\cdot \mid \boldsymbol{x})$ its corrupted version. We write Z^{-i} for the corrupted sequence with position i removed, and analogously for $\boldsymbol{z}^{-i}$. Thus, $Z^{-i} = \boldsymbol{z}^{-i}$ means that all corrupted tokens except the target position are observed.

The LOO target predicts the clean token at position i from this remaining corrupted context:

$$\boldsymbol{x}^i_\star(\boldsymbol{z}, t) \coloneqq \mathbb{E}\left[X^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right]. \tag{4}$$

Since X^i is one-hot, $\boldsymbol{x}^i_{\star,k}(\boldsymbol{z}, t)$ is the probability that the clean token is $\boldsymbol{e}_k$ given the other corrupted tokens. The local observation $\boldsymbol{z}^i$ can then be incorporated exactly through the known uniform corruption likelihood: $\Pr\big(X^i = \boldsymbol{e}_k \mid Z = \boldsymbol{z}, T = t\big) \propto q_t\big(\boldsymbol{z}^i \mid X^i = \boldsymbol{e}_k\big)\, \boldsymbol{x}^i_{\star,k}(\boldsymbol{z}, t)$. That is, $\boldsymbol{x}^i_\star$ summarizes what the surrounding corrupted tokens imply about the clean token, while the bridge uses the known corruption process to account for the token observed at position i itself.

Importantly, this LOO posterior is exactly the population minimizer of the UDM NELBO in Eq. (2):

$$\boldsymbol{x}_{\theta^\star}(\boldsymbol{z}, t) = \boldsymbol{x}_\star(\boldsymbol{z}, t), \quad \text{for almost every } \boldsymbol{z}, t.$$

Substituting $\boldsymbol{x}_\star$ into Eq. (3) gives the corresponding population-optimal reverse kernel:

$$p^{\theta^\star}_{s|t}\big(\boldsymbol{z}^i_s \mid \boldsymbol{z}_t\big) = q_{s|0,t}(\boldsymbol{z}^i_s \mid \boldsymbol{x}^i = \boldsymbol{x}^i_\star(\boldsymbol{z}_t, t), \boldsymbol{z}_t).$$

Eq. (4) tells us what the optimal predictor is, but it does not yet reveal *how* time changes its prediction: time appears only inside the conditioning distribution. To make this dependence explicit, recall that each token under uniform corruption can be viewed as arising either from the copy branch or from the uniform-noise branch. We therefore decompose the prediction over *hypotheses* about which context positions were copied from the clean sequence, which leads to the following property.

Proposition 1 (Explicit time decomposition of the population-optimal UDM predictor)**.** *Fix $i \in [L]$ and a corrupted observation $\boldsymbol{z}$. For any index set $S \subseteq [L] \setminus \{i\}$, write $\boldsymbol{z}[S] \coloneqq \big(\boldsymbol{z}^j\big)_{j\in S}$, and $X[S] \coloneqq \big(X^j\big)_{j\in S}$. Then the population-optimal predictor satisfies*

$$\boldsymbol{x}^i_\star(\boldsymbol{z}, t) = \sum_{S\subseteq[L]\setminus\{i\}} \Pr(S \mid Z^{-i} = \boldsymbol{z}^{-i}, T = t)\mathbb{E}\left[X^i \,\middle|\, X[S] = \boldsymbol{z}[S]\right], \tag{5}$$

$$\Pr(S \mid Z^{-i} = \boldsymbol{z}^{-i}, T = t) \propto \rho_t^{|S|} \Pr(X[S] = \boldsymbol{z}[S]).$$

Here, $\Pr(X[S] = \boldsymbol{z}[S]) = \sum_{\boldsymbol{x}:\, \boldsymbol{x}[S]=\boldsymbol{z}[S]} p_{\mathrm{data}}(\boldsymbol{x})$ is the data probability mass of the partial observation $\boldsymbol{z}[S]$ under the hypothesis index set S.

Proposition 1 makes the role of time explicit: **time controls how much of the hypothesis $\boldsymbol{z}[S]$ should be trusted**. Given a corrupted context $\boldsymbol{z}^{-i}$, the model does not know which observed tokens were from the clean sequence or from uniform noise. It therefore considers every subset $S \subseteq [L] \setminus \{i\}$ as a possible set of copied positions. If the positions in S were indeed copied, then $\mathbb{E}\big[X^i \,\big|\, X[S] = \boldsymbol{z}[S]\big]$ is the corresponding prediction of the target token from that clean context. The final prediction averages over all such possibilities according to their posterior weights.

We now interpret this time-weighted posterior in more detail. Using $\rho_t = K\alpha_t/(1-\alpha_t)$,

$$\Pr(S \mid Z^{-i} = \boldsymbol{z}^{-i}, T = t) \propto \rho_t^{|S|} \Pr(X[S] = \boldsymbol{z}[S]) = \underbrace{\Pr(X[S] = \boldsymbol{z}[S])}_{\substack{\text{Data}\\ \text{dependent}}} \underbrace{K^{|S|}}_{\substack{\text{Fixed vocabulary}\\ \text{factor}}} \underbrace{\left(\tfrac{\alpha_t}{1-\alpha_t}\right)^{|S|}}_{\substack{\text{Time}\\ \text{dependent}}}.$$

This factorization cleanly separates the sources of the posterior weighting. The data-dependent term measures how compatible the observed values on S are with the data distribution. The time-dependent term changes the relative preference for subsets of different sizes as the corruption level varies, while $K^{|S|}$ is a fixed channel factor that amplifies this size dependence.

Near the clean endpoint, ρ_t is large, so the model places more weight on explanations in which many observed tokens were copied from the clean sequence. As noise increases, ρ_t decreases and explanations with fewer copied positions receive relatively more weight. Thus, unlike in MDMs, the population-optimal UDM predictor is generally time-dependent: $\boldsymbol{x}^i_\star(\boldsymbol{z}, \cdot) \not\equiv \mathrm{const}$.

3.2 Time Acts Through the Amount of Matched Context

Proposition 1 expresses the optimal prediction through all 2^{L-1} possible hypotheses of copied context positions. We can further simplify this optimum using the symmetry of the uniform channel (Eq. (1)): every position at which a candidate clean sequence matches the corrupted observation contributes the same factor $1 + \rho_t$. Hence, the likelihood depends on a matching set $M \subseteq [L]$ only through its size, via $(1 + \rho_t)^{|M|}$, regardless of the token identities, the locations of the matches, or whether a matching token was copied from clean sequence or independently produced by uniform corruption. Thus, time acts only through *how many* context positions match. This motivates the following statistic.

Definition 1 (Match count random variable)**.** *For random variables X and Z, and a fixed target position i, define*

$$M^i(X, Z) \coloneqq \sum_{j\in[L]\setminus\{i\}} \mathbb{1}\{X^j = Z^j\}.$$

Thus, M^i counts how many context tokens of a candidate clean sequence agree with the corrupted observation, excluding the target position i, without recording which particular positions match.

Theorem 1 (Time dependence through the matched-context count)**.** *For every target position i,*

$$X^i \perp T \mid Z^{-i}, M^i,$$

i.e., X^i and T are conditionally independent given Z^{-i} and M^i. Moreover, Eq. (4) admits the exact decomposition

$$\boldsymbol{x}^i_\star(\boldsymbol{z}, t) = \sum_{m=0}^{L-1} w^i_m(\boldsymbol{z}, t)\mathbf{h}^i_m(\boldsymbol{z}),$$

$$w^i_m(\boldsymbol{z}, t) \coloneqq \Pr\left(M^i = m \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) \propto (1+\rho_t)^m \Pr\left(M^i(X, \boldsymbol{z}) = m\right),$$

$$\mathbf{h}^i_m(\boldsymbol{z}) \coloneqq \mathbb{E}\left[X^i \,\middle|\, M^i = m, Z^{-i} = \boldsymbol{z}^{-i}\right],$$

where $\Pr(M^i(X, \boldsymbol{z}) = m) = \sum_{M^i(\boldsymbol{x}, \boldsymbol{z}) = m} p_{\text{data}}(\boldsymbol{x})$ is the data probability mass of clean sequences that match $\boldsymbol{z}$ in exactly m context positions.

Theorem 1 gives a compact description of the entire time dependence. Once the corrupted context and its match count M^i are fixed, the distribution of the clean token X^i no longer depends on time. The component prediction $\mathbf{h}^i_m(\boldsymbol{z})$ is therefore time-independent; time only changes how strongly different match counts m are weighted through $(1+\rho_t)^m$. In other words, the population-optimal UDM does not learn an unrelated clean-token prediction at every noise level. Time acts through a single scalar ρ_t that reweights predictions associated with different amounts of matched context.

3.3 Characterizing the Influence of Time in Uniform Diffusion

Theorem 1 shows *where* time enters the population-optimal prediction: it changes the weights over the match count M^i. We next quantify *how strongly* this reweighting changes the prediction.

Proposition 2 (Exact temporal sensitivity)**.** *Let $\eta_t = \log(1+\rho_t)$ and let $\boldsymbol{e}_k$ denote the k-th one-hot vector. Then the temporal derivative of the context posterior is*

$$\frac{\partial}{\partial \eta_t} \boldsymbol{x}^i_\star(\boldsymbol{z}, t) = \mathrm{Cov}_{t,\boldsymbol{z}}\left(\mathbf{h}^i_{M^i}(\boldsymbol{z}), M^i\right),$$

where the covariance is taken componentwise. Equivalently, for any candidate category k with $\boldsymbol{x}^i_{\star,k}(\boldsymbol{z}, t) > 0$,

$$\frac{\partial}{\partial \eta_t} \log \boldsymbol{x}^i_{\star,k}(\boldsymbol{z}, t) = \mathbb{E}_{t,\boldsymbol{z}}\left[M^i \,\middle|\, X^i = \boldsymbol{e}_k\right] - \mathbb{E}_{t,\boldsymbol{z}}[M^i], \tag{6}$$

where the subscript $(t, \boldsymbol{z})$ denotes the posterior distribution $\Pr(\cdot \mid Z^{-i} = \boldsymbol{z}^{-i}, T = t)$.

Proposition 2 makes the effect of time particularly transparent. Larger η_t corresponds to a cleaner channel. Eq. (6) therefore says that a candidate token $\boldsymbol{e}_k$ gains probability toward the clean endpoint when clean sequences predicting $\boldsymbol{e}_k$ have more average matches with the observed context than the posterior average. If they have fewer matches, its probability decreases. Thus, the strength of time dependence is governed by differences in matched-context support across candidate tokens.

Population-level takeaway. The population-optimal UDM predictor is generally time-dependent, but time only reweights predictions across different match counts. Its effect is strongest when competing clean candidates are supported by different amounts of matched context.

4 Time Sensitivity of the Finite-Data Oracle

At the population level, a rich data distribution can assign substantial probability to multiple clean sequences that plausibly explain the same corrupted context. In this regime, changing t can meaningfully reweight these competing explanations and thereby change the optimal prediction.

The situation can be very different under finite training data. An empirical distribution places probability mass on only finitely many sequences in the enormous discrete space $\mathcal{X}^L$. If these sequences are sufficiently separated, a corrupted training sample $\boldsymbol{z}$ can remain much more compatible with its originating sequence $\boldsymbol{x}$ than with any competing sequence that predicts a different token at position i. The posterior may then already be strongly concentrated, leaving little room for the time-dependent reweighting to change the resulting prediction. This motivates the central question:

When does finite training data suppress the time sensitivity of the oracle UDM predictor?

We show that the answer is governed by the geometry of the finite training set: how well separated training sequences are in the enormous discrete sequence space. Our theoretical analysis proceeds in two steps: **1)** when training sequences are sufficiently far apart, the originating sequence is likely to remain separated from competing sequences after corruption, and **2)** whenever this separation is sufficiently large, the oracle Fisher sensitivity for corrupted sequence is strongly suppressed over a broad range of diffusion times, with the guarantee weakening toward the high-noise endpoint.

4.1 When Finite High-Dimensional Data Suppress Time Sensitivity

We now make the finite-data mechanism precise. Given N training sequences $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathcal{X}^L$, define their empirical distribution as $\hat{p}_N \coloneqq \frac{1}{N}\sum_{n=1}^{N} \delta_{\boldsymbol{x}_n}$. This is the finite-data counterpart of p_{data} considered in the previous section, with the oracle supported only on the observed training sequences.

The first key question is how much of the separation between distant training sequences survives even after corruption. Consider a corrupted $\boldsymbol{z}$ generated from a training sequence $\boldsymbol{x}$. For prediction at position i, only training sequences with $\boldsymbol{x}'^i \neq \boldsymbol{x}^i$ can support a different target token. What matters is therefore whether the originating sequence $\boldsymbol{x}$ retains a match-count advantage over every such competitor after corruption. We quantify this advantage through the following separation margin.

Definition 2 (Hamming separation and Δ-separability)**.** *For $\boldsymbol{x} \in \operatorname{supp}(\hat{p}_N)$, define its minimum Hamming distance to the remaining training sequences as*

$$d(\boldsymbol{x}) \coloneqq \min_{\boldsymbol{x}' \in \operatorname{supp}(\hat{p}_N)\setminus\{\boldsymbol{x}\}} \sum_{j=1}^{L} \mathbb{1}\{\boldsymbol{x}'^j \neq \boldsymbol{x}^j\}.$$

For an integer $\Delta \geq 1$, we say that a corrupted sequence $\boldsymbol{z}$ is Δ-separable with respect to $\boldsymbol{x}$ at coordinate i *if*

$$M^i(\boldsymbol{x}, \boldsymbol{z}) - M^i(\boldsymbol{x}', \boldsymbol{z}) \geq \Delta \qquad \forall\, \boldsymbol{x}' \in \operatorname{supp}(\hat{p}_N) \text{ with } \boldsymbol{x}'^i \neq \boldsymbol{x}^i. \tag{7}$$

Thus, Δ is exactly the match-count advantage retained by the originating sequence over every training sequence that would predict a different token at position i. The following result shows that such an advantage survives when the training sequences are sufficiently separated.

Proposition 3 (Δ-separation under UDM corruption (informal))**.** *Given a training sequence $\boldsymbol{x}$, let $T \sim \mathrm{Unif}([0,1])$ and $Z \sim q_T(\cdot \mid \boldsymbol{x})$. For any integer $\Delta \geq 1$ and $i \in [L]$, when $K \gg L$ and $d(\boldsymbol{x})$ is large,*

$$\Pr(Z \text{ is } \Delta\text{-separable w.r.t. } \boldsymbol{x} \text{ at } i) \gtrsim 1 - \frac{\log N + \Delta}{(1-e^{-1})d(\boldsymbol{x})}.$$

This simplified bound exposes the tradeoff directly. If training sequences are sufficiently separated, *i.e.* Hamming separation $d(\boldsymbol{x})$ is large, the source $\boldsymbol{x}$ remains distinguishable after corruption. Conversely, more competing sequences or a larger required margin Δ make the event harder. Importantly, the dependence on dataset size is only logarithmic in N. If $d(\boldsymbol{x})$ is too small relative to $\log N + \Delta$, the bound can become uninformative. The formal finite-K statement is in Appendix B.4.

The second key question is how separation suppresses the time sensitivity of the oracle prediction. For fixed $\boldsymbol{z}$, coordinate i, and time t, define the Fisher time sensitivity

$$\mathcal{I}_t^i(\boldsymbol{z}) \coloneqq \sum_{k:\, \boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t)>0} \boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t)\left(\frac{\partial}{\partial t}\log \boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t)\right)^2 = \sum_{k:\, \boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t)>0} \frac{\left(\frac{\partial}{\partial t}\boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t)\right)^2}{\boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t)}. \tag{8}$$

It measures how quickly the optimal categorical prediction changes when only the diffusion time is varied. Indeed, for infinitesimal δ, we have $\mathcal{D}_{\mathrm{KL}}\left[\boldsymbol{x}^i_\star(\boldsymbol{z},t) \,\middle\|\, \boldsymbol{x}^i_\star(\boldsymbol{z},t+\delta)\right] = \frac{\delta^2}{2}\mathcal{I}_t^i(\boldsymbol{z}) + o(\delta^2)$. Hence, small $\mathcal{I}_t^i(\boldsymbol{z})$ means that nearby times produce nearly indistinguishable oracle predictions.

The next result provides the key link: a large match-count advantage directly limits this time sensitivity.

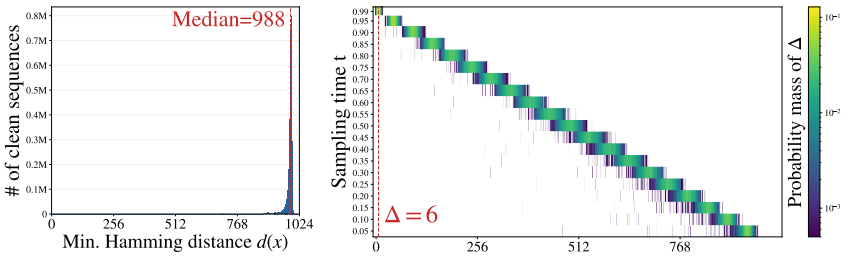


Figure 1: Empirical separation of OWT training data. *Left*: distribution of the minimum Hamming distance $d(\boldsymbol{x})$ from each training sequence to the remaining data, with median $d(\boldsymbol{x}) = 988$. *Right*: distribution of the largest post-corruption separation margin Δ across diffusion times. The dashed line marks $\Delta = 6$, which implies an oracle Fisher-sensitivity bound below 10^{-5} for $t \leq 0.933$. Small separation occurs primarily when $\boldsymbol{z}$ is sampled near the extreme high-noise endpoint.

Theorem 2 (Fisher sensitivity with finite data (informal))**.** *For Δ-separable $\boldsymbol{z}$ with respect to $\boldsymbol{x}$ at coordinate i,*

$$\mathcal{I}_t^i(\boldsymbol{z}) \leq NL^2 \frac{K^2 t^{\Delta-2}}{\big(K-(K-1)t\big)^{\Delta+2}}.$$

Moreover, for $\Delta \geq 2$, the right-hand side is strictly increasing in $t \in (0,1)$.

The bound becomes weaker toward the high-noise endpoint, where corruption makes the originating sequence harder to distinguish from competing sequences. At lower noise levels, a larger separation margin Δ keeps these competitors much less plausible, leaving less room for changes in time to alter the oracle prediction. If $\Delta \geq 2$, a convenient consequence is:

$$\sup_{0<t\leq 1-K^{-1/4}} \mathcal{I}_t^i(\boldsymbol{z}) \leq NL^2 K^{-(3\Delta-2)/4}. \tag{9}$$

For practical LLM vocabulary sizes, the interval $(0, 1-K^{-1/4}]$ already covers most of the diffusion trajectory. More importantly, Eq. (9) can be inverted to ask how much separation is sufficient for a desired sensitivity level. To guarantee $\sup_{0<t\leq 1-K^{-1/4}} \mathcal{I}_t^i(\boldsymbol{z}) \leq \zeta$, it is sufficient that $\Delta \geq \max\left\{2,\ \frac{2}{3} + \frac{4}{3}\log_K \frac{NL^2}{\zeta}\right\}$. Thus, although the Fisher bound contains N and L^2, the required match-count advantage grows only logarithmically with dataset size and sequence length. Each additional unit of Δ further suppresses the bound by a factor of $K^{-3/4}$.

The bounds remain meaningful at LLM scale. Consider a K2-Horizon-375B-scale pretraining configuration (IFM Team, 2026) with $N \approx 1.9 \times 10^9$, $L = 8192$, and $K = 250{,}624$. Here, by Eq. (9), only an integer margin $\Delta \geq 7$ guarantees $\sup_{0<t\leq 0.955} \mathcal{I}_t^i(\boldsymbol{z}) \leq 10^{-5}$. In other words, even with billions of training sequences, the originating sequence only needs to retain a seven-token match-count advantage over every competitor predicting a different target token. By the formal bound in Proposition 3 (Appendix B.4), $d(\boldsymbol{x}) \geq 900$, about 11% of the sequence length, already gives a 0.95 lower bound on 7-separability, while $d(\boldsymbol{x}) \geq 451$ gives a 0.9. Finally, small Fisher sensitivity controls finite changes in the oracle prediction. If $\sup_{0<u\leq 1-\epsilon} \mathcal{I}_u^i(\boldsymbol{z}) \leq \zeta$, then for any $t, s \in (0, 1-\epsilon)$,

$$\left\| \sqrt{\boldsymbol{x}_\star^i(\boldsymbol{z}, t)} - \sqrt{\boldsymbol{x}_\star^i(\boldsymbol{z}, s)} \right\|_2 \leq \tfrac{|t-s|}{2}\sqrt{\zeta}.$$

Thus, once separation makes the Fisher bound small, the finite-data oracle is nearly invariant to time.

4.2 Oracle Has Near-Zero Time Sensitivity on Real Language Data

We now examine whether the finite-data regime predicted by Proposition 3 and Eq. (9) *actually occurs* in real language data. We consider OWT (Gokaslan et al., 2019), a standard benchmark in discrete diffusion language modeling, with $K = 50{,}258$, $L = 1024$, and $N = 8.7 \times 10^6$. For these values, $\sup_{0<t\leq 0.933} \mathcal{I}_t^i(\boldsymbol{z}) \leq \zeta$ with $\zeta = 10^{-5}$ only requires $\Delta \geq 6$. The formal bound in Appendix B.4 further shows that $d(\boldsymbol{x}) \geq 698$ is sufficient for at least a 0.95 lower bound on 6-separability.

We now compute $d(\boldsymbol{x})$ and the post-corruption separation margin Δ directly on OWT. Before diffusion corruption, clean OWT is highly separated in Hamming space. As shown in Figure 1 (*Left*),

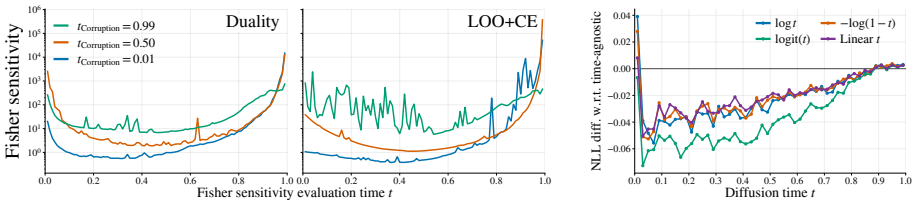


Figure 2: *Left*: Fisher sensitivity of DUALITY and LOO+CE, varying $\boldsymbol{z}$ sampling time $t_{\text{corruption}} \in \{0.01, 0.50, 0.99\}$ and evaluating Fisher sensitivity on every time coordinate. *Right*: Per-time NLL difference between time-agnostic and time-variant UDMs.

the distribution of $d(\boldsymbol{x})$ is concentrated near the maximum sequence length $L = 1024$, with median $d(\boldsymbol{x}) = 988$, well above the sufficient value 698. We next test whether this separation survives UDM corruption. For each $t \in \{0.05, 0.1, \ldots, 0.95, 0.99\}$, we sample 2000 pairs $\boldsymbol{x} \sim \hat{p}_N$ and $\boldsymbol{z}_t \sim q_t(\cdot \mid \boldsymbol{x})$ and measure the largest Δ for which $\boldsymbol{z}_t$ is Δ-separable. Figure 1 (*Right*) shows that the observed margin remains above 6 over almost the entire diffusion trajectory, becoming small primarily when $\boldsymbol{z}$ is sampled near the extreme high-noise endpoint.

These measurements confirm that OWT lies in the separation regime predicted by our theory: most of the training pairs $(\boldsymbol{x}, \boldsymbol{z})$ satisfies 6-separability, and whenever the observed margin satisfies $\Delta \geq 6$, Eq. (9) gives $\sup_{0<t\leq 0.933} \mathcal{I}_t^i(\boldsymbol{z}) \leq 10^{-5}$, implying negligible oracle time sensitivity over most of the diffusion trajectory. Near the extreme high-noise endpoint, smaller separation margins weaken this guarantee and leave more room for time to influence the oracle prediction.

Finite-data takeaway. When finite training sequences remain well separated after corruption, the originating sequence retains a clear match-count advantage and the oracle becomes nearly insensitive to time over most of the diffusion trajectory. This regime is empirically observed in practical language data, with the separation weakening mainly near the high-noise endpoint.

5 YOU MIGHT NOT NEED TIME IN UDMS FOR TEXT GENERATION

Sections 3 and 4 reveal a tension: the population-optimal UDM predictor generally depends on time, while the finite-data oracle can be nearly time-insensitive over most of the time. The remaining question is whether a learned UDM actually benefits from time conditioning. Two possibilities arise:

- **Time-variant UDMs may benefit from recovering useful time dependence** through generalization or inductive bias beyond the finite-data oracle.
- **Time-agnostic UDMs may benefit from removing largely unnecessary time dependence.** If time dependence remains weak over most of the trajectory, removing explicit time conditioning may simplify the model and improve learning.

We test these directly by measuring the time sensitivity of learned UDMs and comparing time-conditioned and time-agnostic models. Throughout this section, DUALITY denotes the framework of Sahoo et al. (2025), while LOO+CE denotes the LOO denoiser trained with cross-entropy (Gourevitch et al., 2026). See Appendix A.2 for framework details and C.2 for experimental settings.

5.1 LEARNED UDMS EXHIBIT LIMITED TIME DEPENDENCE IN LANGUAGE

Learned time sensitivity follows the finite-data prediction. We first examine whether the time sensitivity of learned UDMs exhibits the qualitative behavior predicted by our finite-data analysis. Using the official OWT checkpoints of DUALITY and LOO+CE, we sample corrupted sequences $\boldsymbol{z}$ at $t_{\text{corruption}} \in \{0.01, 0.50, 0.99\}$. For each fixed $\boldsymbol{z}$, we then vary the time input used for prediction and measure its Fisher sensitivity across the entire diffusion trajectory.

As shown in Figure 2 (*left*), the learned models qualitatively match our finite-data analysis: First, Fisher sensitivity increases with $t_{\text{corruption}}$. Stronger corruption tends to reduce the separation margin Δ, weakening the finite-data suppression guarantee in Eq. (9). Second, for a fixed corrupted input, Fisher sensitivity generally increases with the evaluation time and rises sharply near the high-noise endpoint, where the bound in Theorem 2 becomes weaker. Finally, we use held-out test sequences for this analysis, although our finite-data theory is stated for the empirical training distribution. The

Table 1: Validation perplexity (PPL) across several configurations. Lower is better. Hybrid uses the time-modulation layer only for $t > 0.8$. PPLs lower than the time-variant baseline are underlined.

Dataset	Objective	Variant (139M)	Hybrid (139M)	Agnostic (131M)	Agnostic (139M)
LM1B	DUALITY	29.99	<u>29.92</u>	30.32	<u>29.76</u>
	LOO+CE	30.15	<u>29.67</u>	<u>29.18</u>	<u>28.93</u>
LM1B (Packed)	DUALITY	33.48	<u>33.17</u>	33.69	<u>33.35</u>
	LOO+CE	33.94	<u>33.37</u>	<u>33.24</u>	<u>32.75</u>

similar qualitative behavior on held-out data suggests that limited time sensitivity can extend beyond training examples, while our theory does not directly establish this behavior off the empirical support.

The localized time sensitivity is robust and provides limited practical benefit. We next ask whether this concentration of time sensitivity can be resolved by how time is parameterized to the model. We train conventional UDMs (Schiff et al., 2025) on LM1B. We keep the forward process unchanged and reparameterize only the scalar passed to the time embedding as $\hat{t} = f(t)$, where $f(t) \in \{-\log(1-t), \log t, \mathrm{logit}(t), t\}$. These parameterizations allocate different resolution across diffusion times. Nevertheless, Figure 4 in Appendix shows a consistent pattern across all parameterizations: learned UDMs exhibit weak time sensitivity except near the high-noise endpoint.

We next compare the time-conditioned model with its time-agnostic counterpart. Figure 2 (*Right*) shows that time conditioning improves per-time NLL mainly near $t = 1$, precisely where the learned Fisher sensitivity becomes substantial. Over most of the diffusion trajectory, however, the time-agnostic model achieves lower NLL and also performs better overall. Together, these results suggest that learned UDMs make substantial use of explicit time conditioning only in a limited high-noise region, and that modeling this localized dependence provides limited practical benefit.

5.2 TIME-AGNOSTIC UDM VERSUS TIME-VARIANT UDM

We next compare time-agnostic and time-variant UDMs across broader settings, including different datasets, tokenizers, and training objectives. Beyond the conventional UDM studied in Section 5.1, we now evaluate DUALITY and LOO+CE. We follow widely adopted experimental settings in discrete diffusion language modeling (Sahoo et al., 2025; Arriola et al., 2025; Schiff et al., 2025), evaluating models on LM1B (Chelba et al., 2014), both with and without sentence packing, and OWT, all trained for 1M steps, with the same backbone. See Appendix C.2 for more detailed settings.

Results on LM1B. We compare four setttings for time conditioning: 1) Time-variant model, 2) Time-agnostic model where we remove the time-modulation layers, reducing the parameter count from 139M to 131M, 3) Time-agnostic model with 139M parameters obtained by adding one Transformer block to 2), and 4) Hybrid model that only activates time-modulation layer when $t > 0.8$. Table 1 shows that the hybrid model that only learns time-sensitivity on high-noise regime is almost equivalent to the time-variant model, and moreover, time-agnostic UDMs remain consistently competitive with their time-dependent counterparts across training frameworks.

Results on OWT. We compare time-variant and time-agnostic models with the same parameter size. As shown in Table 2, time-agnostic UDMs are comparable to time-variant UDMs. Generative PPL was comparable for DUALITY, while the time-agnostic model performed better for LOO+CE; see Appendix C.1.

Table 2: Validation PPL on OWT.

	Agnostic	Variant
DUALITY	24.33	25.54
LOO+CE	25.29	25.29

Learned-model takeaway. Time matters mainly near the high-noise endpoint. Consistently, time-agnostic UDMs remain competitive across datasets and objectives, suggesting that explicit time conditioning may offer limited benefit over most of the trajectory.

6 CONCLUSION

We studied the role of time in population optimal UDMs, and why it weakens under finite training data. Consistently, learned UDMs exhibit limited time dependence over most of the trajectory, while time-agnostic models remain competitive across datasets and frameworks. We expect this view opens a door to future research that allows a simpler foundation for architectures, training, and samplers.

AI USE STATEMENT

In this work, we used generative AI tools for writing assistance, including grammar checking, writing polish, checking proofs for errors, and improving the clarity and readability of author-written text. We also used these tools to review and debug code, and develop preliminary figure layouts. We reviewed all AI-assisted works and take responsibility for the final content of this work.

ETHICS STATEMENT

This work studies theoretical and empirical properties of time conditioning in discrete diffusion language models. It does not introduce new datasets, involve human subjects, or deploy models in user-facing settings. The potential societal impacts are therefore similar to those of existing generative language models. We do not anticipate additional ethical risks specific to this work.

REPRODUCIBILITY STATEMENT

We provide complete proofs of our theoretical results in Appendix B. Experimental settings, including datasets, preprocessing, model architectures, training hyperparameters, and additional results, are provided in Appendix C. Our time-agnostic models make only a minimal modification to the official code of the existing baselines (Sahoo et al., 2025; Gourevitch et al., 2026; Schiff et al., 2025). We will release the code upon acceptance.

REFERENCES

Alan Nawzad Amin, Nate Gruver, and Andrew Gordon Wilson. Why masking diffusion works: Condition on the jump schedule for improved discrete diffusion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Marianne Arriola, Aaron Gokaslan, Justin T. Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *International Conference on Learning Representations (ICLR)*, 2025.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Andrew Campbell, Joe Benton, Valentin De Bortoli, Thomas Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One billion word benchmark for measuring progress in statistical language modeling. In *Interspeech 2014*, pp. 2635–2639, 2014. doi: 10.21437/Interspeech.2014-564.

DiffusionGemma Team. DiffusionGemma technical report, 2026. URL https://arxiv.org/abs/2608.00146.

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

Samson Gourevitch, Yazid Janati, Dario Shariatian, Umut Simsekli, Eric Moulines, Eric P. Xing, and Alain Durmus. Uniform diffusion models revisited: Leave-one-out denoiser and absorbing state reformulation, 2026. URL https://arxiv.org/abs/2605.22765.

Zhengfu He, Tianxiang Sun, Qiong Tang, Kuanning Wang, Xuanjing Huang, and Xipeng Qiu. DiffusionBERT: Improving generative masked language models with diffusion models. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023.

Alec Helbling, Sebastian Gutierrez Hernandez, Benjamin Hoover, Duen Horng Chau, and Parikshit Ram. What time is it? how data geometry makes time conditioning optional for flow matching, 2026. URL https://arxiv.org/abs/2605.08344.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Chunsan Hong, Seonho An, Min-Soo Kim, and Jong Chul Ye. Improving discrete diffusion unmasking policies beyond explicit reference policies. In *International Conference on Learning Representations (ICLR)*, 2026.

Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

IFM Team. Introducing k2 horizon: Frontier performance, radically open, 2026. URL https://ifm.ai/blog/k2/.

Chieh-Hsin Lai, Yang Song, Dongjun Kim, Yuki Mitsufuji, and Stefano Ermon. The principles of diffusion models, 2025. URL https://arxiv.org/abs/2510.21890.

Sanghyun Lee, Sunwoo Kim, Seungryong Kim, Jongho Park, and Dongmin Park. Effective test-time scaling of discrete diffusion through iterative refinement, 2025. URL https://arxiv.org/abs/2511.05562.

Sulin Liu, Juno Nam, Andrew Campbell, Hannes Stark, Yilun Xu, Tommi Jaakkola, and Rafael Gomez-Bombarelli. Think while you generate: Discrete diffusion with planned denoising. In *International Conference on Learning Representations (ICLR)*, 2025.

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

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

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 (ICLR)*, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *IEEE International Conference on Computer Vision (ICCV)*, 2023.

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

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

Subham Sekhar Sahoo, Justin Deschenaux, Aaron Gokaslan, Guanghan Wang, Justin Chiu, and Volodymyr Kuleshov. The diffusion duality. In *International Conference on Machine Learning (ICML)*, 2025.

Subham Sekhar Sahoo, Lingjie Chen, Khiem Pham, Jonathan Geuter, Chaitanya Dwivedi, Varad Pimpalkhute, Yash Akhauri, Alexander Moreno, Mikhail Yurochkin, Zhenting Wang, Mostafa Elhoushi, Nolan Dey, Shane Bergsma, Joel Hestness, John Thickstun, Eric Xing, and Zhengzhong Liu. Unlocking lossless speedups in llms via discrete diffusion, 2026a. URL https://arxiv.org/abs/2609.04010.

Subham Sekhar Sahoo, Jean-Marie Lemercier, Zhihan Yang, Justin Deschenaux, Jingyu Liu, John Thickstun, and Ante Jukić. Scaling beyond masked diffusion language models. In *International Conference on Machine Learning (ICML)*, 2026b.

Mojtaba Sahraee-Ardakan, Mauricio Delbracio, and Peyman Milanfar. The geometry of noise: Why diffusion models don't need noise conditioning, 2026. URL https://arxiv.org/abs/2602.18428.

Yair Schiff, Subham Sekhar Sahoo, Hao Phung, Guanghan Wang, Sam Boshar, Hugo Dalla-torre, Bernardo P de Almeida, Alexander M Rush, Thomas Pierrot, and Volodymyr Kuleshov. Simple guidance mechanisms for discrete diffusion models. In *International Conference on Learning Representations (ICLR)*, 2025.

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

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning (ICML)*, 2015.

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 (ICLR)*, 2021.

Qiao Sun, Zhicheng Jiang, Hanhong Zhao, and Kaiming He. Is noise conditioning necessary for denoising generative models? In *International Conference on Machine Learning (ICML)*, 2025.

Dimitri von Rütte, Janis Fluri, Omead Pooladzandi, Bernhard Schölkopf, Thomas Hofmann, and Antonio Orvieto. Scaling behavior of discrete diffusion language models. In *International Conference on Learning Representations (ICLR)*, 2026.

Runqian Wang and Yilun Du. Equilibrium matching: Generative modeling with implicit energy-based models, 2025. URL https://arxiv.org/abs/2510.02300.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models, 2025. URL https://arxiv.org/abs/2508.15487.

Kaiwen Zheng, Yongxin Chen, Hanzi Mao, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. In *International Conference on Learning Representations (ICLR)*, 2025.

CONTENTS

1 Introduction **1**

2 Background **2**

3 The Role of Time at the Population Optimum in UDMs **3**

3.1 The Population Target of the Uniform Channel 3

3.2 Time Acts Through the Amount of Matched Context 4

3.3 Characterizing the Influence of Time in Uniform Diffusion 5

4 Time Sensitivity of the Finite-Data Oracle **5**

4.1 When Finite High-Dimensional Data Suppress Time Sensitivity 6

4.2 Oracle Has Near-Zero Time Sensitivity on Real Language Data 7

5 You Might Not Need Time in UDMs for Text Generation **8**

5.1 Learned UDMs Exhibit Limited Time Dependence in Language 8

5.2 Time-agnostic UDM versus Time-variant UDM 9

6 Conclusion **9**

A Background **14**

A.1 Related Work 14

A.2 Various Uniform Discrete Diffusion Frameworks 15

B Omitted Proofs **18**

B.1 Proof of Proposition 1 18

B.2 Proof of Theorem 1 20

B.3 Proof of Proposition 2 21

B.4 Proof of Proposition 3 22

B.5 Proof of Theorem 2 25

B.6 Proof of Eq. (9) 27

B.7 Discussion on the Finite Data Setting 27

C Experimental Details **29**

C.1 Additional Experimental Results 29

C.2 Experimental Settings 30

C.3 Details on Time-agnostic Models 30

A BACKGROUND

A.1 RELATED WORK

Discrete diffusion models. Diffusion generative modeling was originally introduced by Sohl-Dickstein et al. (2015), and has become a dominant approach for continuous domains such as images (Ho et al., 2020; Song et al., 2021). By defining a time-variant stochastic differential equation (Song et al., 2021), diffusion models establish a continuous path between the data and noise distributions. This success has motivated extensions to discrete domains, leading to discrete diffusion models (Austin et al., 2021; Hoogeboom et al., 2021; Campbell et al., 2022). The forward corruption process is typically designed in one of two ways: (i) *uniform* corruption, which replaces tokens with random vocabulary elements (Lou et al., 2024; Sahoo et al., 2025), or (ii) *masking*-based corruption, which maps tokens to an absorbing `[MASK]` state (Sahoo et al., 2024; Shi et al., 2024).

Uniform discrete diffusion models and plug-in parametrization. Early uniform diffusion models (UDMs), together with masked diffusion models (MDMs), were formulated as discrete diffusion models by specifying particular forward corruption kernels (Austin et al., 2021; Campbell et al., 2022; Lou et al., 2024; Hoogeboom et al., 2021). Among those, the uniform bridge plug-in parameterization was first introduced in multinomial diffusion by Hoogeboom et al. (2021). Using uniform bridge plug-in parameterization, Schiff et al. (2025) developed a continuous-time formulation and derived its NELBO, reporting improved performance over earlier uniform-diffusion baselines. Building on this, Sahoo et al. (2025) reinterpreted the uniform kernel as the $\arg\max$ pushforward of a Gaussian corruption kernel and proposed a curriculum that initially uses a biased continuous-input relaxation before returning to discrete-input UDM training. More recently, Gourevitch et al. (2026) characterized the unrestricted population optimum of the bridge plug-in predictor as a leave-one-out posterior, which enables training with a simple cross-entropy objective under a leave-one-out parametrization.

Notably, recent studies have shown that UDMs can outperform MDMs when scaled up in the language domain (Sahoo et al., 2026b; von Rütte et al., 2026), and large UDMs such as DiffusionGemma (DiffusionGemma Team, 2026) and Uno (Sahoo et al., 2026a) have recently been released. Despite growing attention to the application of UDMs in the language domain, it remains unclear whether time conditioning is actually necessary for UDMs. In this regard, our work takes the population-optimum characterization as a starting point and analyzes its dependence on time in detail, showing that, in the language domain, the time sensitivity of UDMs can be largely confined to the endpoint region of t.

Time-agnostic masked diffusion models. Earlier continuous-time formulations for mask corruption kernels commonly used time-conditioned networks to parameterize the reverse process (Campbell et al., 2022; Lou et al., 2024). However, Zheng et al. (2025); Ou et al. (2025) showed that MDMs do not inherently require time conditioning in the model, and Sahoo et al. (2024) demonstrated empirically that time-agnostic MDMs can slightly outperform their time-variant counterparts. Consequently, recent large-scale MDMs (Ye et al., 2025; Nie et al., 2025) have largely adopted time-agnostic architectures. Consequently, time-agnostic MDMs provide a simpler foundation, leading to subsequent developments in inference (Lee et al., 2025) and reinforcement learning (Hong et al., 2026). Accordingly, if time-agnostic modeling is also viable for UDMs, it would provide a simpler and more extensible framework. Unlike MDMs, however, where time-agnostic and time-variant models share the same population optimum, we show that the two optima generally differ for UDMs. Nevertheless, under a finite-data setting, we find that time-agnostic UDMs can still be practically viable.

Time-agnostic continuous diffusion models. Recent studies have investigated when continuous diffusion models can dispense with explicit time conditioning (Sahraee-Ardakan et al., 2026; Sun et al., 2025; Helbling et al., 2026). In continuous diffusion, the state evolves in a differentiable continuous space, and the reverse process is described locally by a score $\nabla_{\boldsymbol{z}} \log p_t(\boldsymbol{z})$ or, equivalently in the ODE view, by a vector field over the continuous state space. The current state can therefore encode *where the sample lies along the generative trajectory*, allowing even a time-independent vector field to evolve according to its state. This continuous dynamics of particles enables geometric explanations for time-agnosticity. For example, Sahraee-Ardakan et al. (2026) study a time-independent vector field and relate the resulting dynamics to a marginal energy landscape and gradient flow. A complementary explanation is that the corrupted observation itself can reveal the noise level. Sun et al. (2025) relate concentration of $T \mid Z = \boldsymbol{z}$ to the gap between time-agnostic and time-conditioned predictors, and further extend their analysis to multiple data points under an i.i.d. Gaussian data model. Similarly,

Helbling et al. (2026) construct a time estimator $\hat{t}(\boldsymbol{z})$ under a spiked-covariance Gaussian model and show that the error from hiding time becomes small in high dimensions.

In contrast, our analysis first reveals the distinct role of time in uniform discrete diffusion: *time tells which subsets of observed tokens are plausibly inherited from the clean sequence*. A key structural distinction from continuous diffusion lies in the corruption kernel. In continuous diffusion, Gaussian noise continuously perturbs every coordinate, so each corrupted coordinate generally retains a partial contribution from the clean state. In discrete diffusion, by contrast, the state evolves through tokenwise jumps, so only a subset of tokens may change. Consequently, a corrupted sequence contains tokens directly inherited from the clean sequence alongside tokens generated independently by noise, with the latter carrying no information about their original tokens. This distinction enables us to clearly understand the specific role of time in uniform discrete diffusion.

Consequently, our analysis reveals a different mechanism through which explicit time conditioning can become unnecessary in uniform discrete diffusion. The key distinction is that the role of time itself can be limited for the prediction of $\boldsymbol{z}$, rather than time being inferred from $\boldsymbol{z}$: for a finite training set, sufficiently separated clean sequences concentrate the empirical posterior onto a small set of compatible hypotheses, making the empirical-optimal predictor nearly invariant to time. We quantify this through the post-corruption separation margin and directly bound the oracle time sensitivity.

Other related works. Liu et al. (2025) decomposes uniform discrete diffusion into a planner that infers latent corruption identities and a denoiser that predicts clean tokens, highlighting the role of corruption-state uncertainty. Relatedly, Amin et al. (2025) conditions discrete diffusion on the jump schedule and connects uniform diffusion to masking diffusion when corruption-event information is revealed. While these works motivate reasoning about latent corruption states, we instead characterize how such ambiguity induces time dependence in the population-optimal UDM predictor and when this dependence becomes negligible.

A.2 Various Uniform Discrete Diffusion Frameworks

We briefly review several representative formulations of uniform discrete diffusion. We first describe the general CTMC view and its posterior- (Campbell et al., 2022) and score-based parameterizations (Lou et al., 2024). We then introduce the work of Schiff et al. (2025), which uses a plug-in UDM parameterization improving empirical performance over these earlier posterior- and score-based parameterizations, and forms the basis of the modern UDM family studied in this work. Finally, we review two recent developments built on this formulation, Diffusion Duality (Sahoo et al., 2025) and the leave-one-out parameterization of Gourevitch et al. (2026).

Posterior and score parameterizations in continuous-time Markov chains. Discrete diffusion can be formulated as a continuous-time Markov chain (CTMC) (Campbell et al., 2022; Lou et al., 2024). We describe here only the single-token process on $\mathcal{X} = \{\boldsymbol{e}_1, \dots, \boldsymbol{e}_K\}$, where $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$ are categorical states represented as one-hot vectors. For sequence-level extensions, we refer to the original works for these constructions. With a slight abuse of notation, write $q_t(\boldsymbol{z} \mid \boldsymbol{x})$ for the forward transition kernel $q_{t|0}(\boldsymbol{z} \mid \boldsymbol{x})$ and define its marginal by

$$p_t(\boldsymbol{z}) = \sum_{\boldsymbol{x} \in \mathcal{X}} p_{\mathrm{data}}(\boldsymbol{x}) q_t(\boldsymbol{z} \mid \boldsymbol{x}).$$

Using the row-vector convention, let

$$\boldsymbol{p}_t = \big(p_t(\boldsymbol{z})\big)_{\boldsymbol{z} \in \mathcal{X}} \in \mathbb{R}^{1 \times K},$$

and let $\boldsymbol{Q}_t \in \mathbb{R}^{K \times K}$ denote the forward transition-rate matrix. The forward dynamics satisfy

$$\frac{\mathrm{d}\boldsymbol{p}_t}{\mathrm{d}t} = \boldsymbol{p}_t \boldsymbol{Q}_t,$$

where the off-diagonal entries of $\boldsymbol{Q}_t$ are nonnegative and each row sums to zero. We use this row-vector convention throughout. This is the transpose of the column-vector convention used by Lou et al. (2024), where $\mathrm{d}\boldsymbol{p}_t/\mathrm{d}t = \boldsymbol{Q}_t \boldsymbol{p}_t$ and the columns of $\boldsymbol{Q}_t$ sum to zero.

For uniform diffusion, the transition kernel and its corresponding rate matrix are

$$\boldsymbol{P}_{t|0} = \alpha_t \boldsymbol{I} + (1 - \alpha_t) K^{-1} \mathbf{1}_K \mathbf{1}_K^\top, \qquad \boldsymbol{Q}_t = -\frac{\alpha_t'}{\alpha_t} \left(K^{-1} \mathbf{1}_K \mathbf{1}_K^\top - \boldsymbol{I} \right).$$

Thus, for two distinct states, the instantaneous transition rate is $-\alpha_t'/(K\alpha_t)$. Equivalently, one may use the unnormalized uniform generator $\mathbf{1}_K\mathbf{1}_K^\top - K\boldsymbol{I}$, as in Lou et al. (2024), with its scale absorbed into the scalar noise schedule. Since the uniform generator is symmetric, the row- and column-vector conventions yield the same numerical generator matrix in this particular case.

The time reversal is again a CTMC. Under our row-vector convention, its exact reverse transition rate from $\boldsymbol{z}$ to $\boldsymbol{z}' \neq \boldsymbol{z}$ is

$$\bar{\boldsymbol{Q}}_t(\boldsymbol{z}, \boldsymbol{z}') = \boldsymbol{Q}_t(\boldsymbol{z}', \boldsymbol{z})\frac{p_t(\boldsymbol{z}')}{p_t(\boldsymbol{z})}.$$

The diagonal entries are determined by requiring each row of $\bar{\boldsymbol{Q}}_t$ to sum to zero. Hence, constructing the reverse process amounts to estimating the marginal ratio $p_t(\boldsymbol{z}')/p_t(\boldsymbol{z})$.

The CTMC denoising formulation of Campbell et al. (2022) obtains this ratio indirectly through the clean posterior. By Bayes' rule,

$$\frac{p_t(\boldsymbol{z}')}{p_t(\boldsymbol{z})} = \sum_{\boldsymbol{x}\in\mathcal{X}} \frac{q_t(\boldsymbol{z}' \mid \boldsymbol{x})}{q_t(\boldsymbol{z} \mid \boldsymbol{x})} q_{0|t}(\boldsymbol{x} \mid \boldsymbol{z}).$$

Since the clean posterior $q_{0|t}(\boldsymbol{x} \mid \boldsymbol{z})$ is generally intractable, it is replaced by a denoising model $\boldsymbol{\mu}_\theta^{\text{Posterior}}(\boldsymbol{x} \mid \boldsymbol{z}, t)$, yielding

$$\bar{\boldsymbol{Q}}_t^\theta(\boldsymbol{z}, \boldsymbol{z}') = \boldsymbol{Q}_t(\boldsymbol{z}', \boldsymbol{z}) \sum_{\boldsymbol{x}\in\mathcal{X}} \frac{q_t(\boldsymbol{z}' \mid \boldsymbol{x})}{q_t(\boldsymbol{z} \mid \boldsymbol{x})} \boldsymbol{\mu}_\theta^{\text{Posterior}}(\boldsymbol{x} \mid \boldsymbol{z}, t), \qquad \boldsymbol{z}' \neq \boldsymbol{z}.$$

Thus, this formulation parameterizes the reverse dynamics through a model of the clean posterior and trains the resulting reverse process using the continuous-time NELBO derived from the forward and reverse CTMCs.

SEDD (Lou et al., 2024), in contrast, directly parameterizes the *concrete score*, i.e., the collection of marginal ratios

$$s_t(\boldsymbol{z})_{\boldsymbol{z}'} \coloneqq \frac{p_t(\boldsymbol{z}')}{p_t(\boldsymbol{z})}, \qquad \boldsymbol{z}' \neq \boldsymbol{z}.$$

Replacing the intractable score by a model $\boldsymbol{\mu}_\theta^{\text{Score}}$ gives, in our row-vector convention,

$$\bar{\boldsymbol{Q}}_t^\theta(\boldsymbol{z}, \boldsymbol{z}') = \boldsymbol{Q}_t(\boldsymbol{z}', \boldsymbol{z})\boldsymbol{\mu}_\theta^{\text{Score}}(\boldsymbol{z}, t)_{\boldsymbol{z}'}, \qquad \boldsymbol{z}' \neq \boldsymbol{z}.$$

SEDD trains these marginal ratios directly using diffusion-weighted denoising score entropy. Thus, both approaches target the same reverse CTMC but differ in how it is parameterized: the CTMC denoising formulation models the clean posterior and obtains the marginal ratio indirectly, whereas SEDD models the marginal ratio itself as a discrete score.

Plug-in parametrized uniform diffusion models. Recent uniform diffusion models largely follow the plug-in parameterization adopted by Schiff et al. (2025), which yields empirical improvements over the score- and posterior-based parameterizations of the CTMC formulations described above. A UDM independently corrupts each token toward the uniform prior:

$$q_t(\boldsymbol{z}_t^i \mid \boldsymbol{x}^i) = \text{Cat}\left(\boldsymbol{z}_t^i; \alpha_t \boldsymbol{x}^i + (1-\alpha_t)K^{-1}\mathbf{1}_K\right).$$

For $s < t$, the single token posterior $q_{s|0,t}$ follows from Bayes' rule using the $s \to t$ transition and the s-marginal:

$$q_{s|0,t}(\boldsymbol{z}_s^i \mid \boldsymbol{x}^i, \boldsymbol{z}_t^i) = \text{Cat}\left(\boldsymbol{z}_s^i; \frac{\rho_t \boldsymbol{z}_t^i \odot \boldsymbol{x}^i + \frac{\rho_t}{\rho_s}\boldsymbol{z}_t^i + (1-\frac{\rho_t}{\rho_s})(\frac{\rho_s \boldsymbol{x}^i + \mathbf{1}_K}{K+\rho_s})}{\rho_t \langle \boldsymbol{z}_t^i, \boldsymbol{x}^i \rangle + 1}\right). \tag{10}$$

To mimic the true posterior, Schiff et al. (2025) design reverse process with plug-in parametrization:

$$\begin{aligned} p_{s|t}^\theta(\boldsymbol{z}_s^i \mid \boldsymbol{z}_t) &= q_{s|0,t}(\boldsymbol{z}_s^i \mid \boldsymbol{x}^i = \boldsymbol{x}_\theta^i(\boldsymbol{z}_t, t), \boldsymbol{z}_t) \\ &= \text{Cat}\left(\boldsymbol{z}_s^i; \frac{\rho_t \boldsymbol{z}_t^i \odot \boldsymbol{x}_\theta^i(\boldsymbol{z}_t, t) + \frac{\rho_t}{\rho_s}\boldsymbol{z}_t^i + \left(1-\frac{\rho_t}{\rho_s}\right)\frac{\rho_s \boldsymbol{x}_\theta^i(\boldsymbol{z}_t,t) + \mathbf{1}_K}{K+\rho_s}}{\rho_t \left\langle \boldsymbol{z}_t^i, \boldsymbol{x}_\theta^i(\boldsymbol{z}_t, t)\right\rangle + 1}\right). \end{aligned}$$

The training of UDM uses the variational upper bound (NELBO), and the discrete-time NELBO is:

$$
\begin{aligned}
&-\log p_\theta(\boldsymbol{x}) \\
&\leq \mathbb{E}\Big[-\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z}_{t(0)}) + \sum_i \mathcal{D}_{\mathrm{KL}}[q_{s(i)|0,t(i)}(\cdot)\|p_\theta(\boldsymbol{z}_{s(i)} \mid \boldsymbol{z}_{t(i)})] + \mathcal{D}_{\mathrm{KL}}[q_{t(T)}(\cdot)\|p_\theta(\boldsymbol{z}_{t(T)})]\Big]
\end{aligned}
$$

The prior and reconstruction loss vanish for the continuous-time, and only the diffusion loss remains:

$$
\mathcal{L}_{\mathrm{UDM}}(\theta) = \int_0^1 \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}_t} \sum_{i=1}^{L} \frac{-\alpha_t'}{K\alpha_t} \left[\frac{K}{\bar{x}^i_{\theta,\ell_i}} - \frac{K}{\bar{x}^i_{\ell_i}} + \sum_{j \neq \ell_i} \frac{\bar{x}^i_j}{\bar{x}^i_{\ell_i}} \log \frac{\bar{x}^i_{\theta,\ell_i}\bar{x}^i_j}{\bar{x}^i_{\theta,j}\bar{x}^i_{\ell_i}} \right] \mathrm{d}t.
$$

where $\bar{\boldsymbol{\nu}}^i \coloneqq K\alpha_t \boldsymbol{\nu}^i + (1-\alpha_t)\mathbf{1}_K$ and ℓ_i denotes non-zero dimension index of $\boldsymbol{z}_t^i$.

Diffusion duality. Diffusion Duality (Sahoo et al., 2025) retains the plug-in parameterization of Schiff et al. (2025), but provides an underlying Gaussian view of the uniform forward process. For each token, consider the Gaussian diffusion

$$
\tilde{q}_t(\boldsymbol{w}_t^i \mid \boldsymbol{x}^i; \tilde{\alpha}_t) = \mathcal{N}\big(\boldsymbol{w}_t^i; \tilde{\alpha}_t \boldsymbol{x}^i, (1-\tilde{\alpha}_t^2)\boldsymbol{I}\big)\,.
$$

Diffusion Duality shows that applying $\arg\max$ to this Gaussian latent recovers the marginal of uniform discrete diffusion:

$$
q_t\big(\boldsymbol{z}_t^i \mid \boldsymbol{x}^i; \mathcal{T}(\tilde{\alpha}_t)\big) = [\arg\max]_\star\, \tilde{q}_t\big(\boldsymbol{w}_t^i \mid \boldsymbol{x}^i; \tilde{\alpha}_t\big)\,, \qquad \alpha_t = \mathcal{T}(\tilde{\alpha}_t),
$$

where $[\arg\max]_\star$ denotes the pushforward under the $\arg\max$ mapping and $\mathcal{T}$ is the diffusion transformation relating the Gaussian and uniform noise schedules. Thus, the uniform diffusion marginal can be viewed as the discrete projection of an underlying Gaussian diffusion.

This duality is further used to construct a low-variance curriculum for training the same plug-in UDM. Diffusion Duality first rewrites the discrete NELBO exactly using Gaussian latents and a hard argmax projection. Its curriculum then replaces the hard input with a tempered softmax, introducing a lower-variance but biased relaxation that approaches the original discrete-input objective as the temperature decreases. More specifically, Diffusion Duality relaxes this input using the identity

$$
\arg\max(\boldsymbol{w}_t^i) = \lim_{\tau \to 0^+} \mathrm{softmax}\left(\frac{\boldsymbol{w}_t^i}{\tau}\right).
$$

During the curriculum, the denoising model therefore receives $\mathrm{softmax}(\boldsymbol{w}_t/\tau)$ instead of the hard $\arg\max(\boldsymbol{w}_t)$ input. A positive temperature preserves more information from the Gaussian latent, making the denoising problem easier and reducing training variance; as τ is annealed toward zero, the model progressively approaches the original discrete UDM training problem. In particular, in their actual experiments, they use the hard $\arg\max$ for last half of training, thereby retaining the original UDM's discrete input and NELBO exactly. The final model retains the same plug-in parameterization and is evaluated as a standard uniform diffusion model.

Leave-one-out parameterization. Gourevitch et al. (2026) characterize the target implicit in the UDM bridge plug-in parameterization. For each position $i \in [L]$, the population-optimal network prediction is the leave-one-out (LOO) posterior

$$
\boldsymbol{x}_\star^i(\boldsymbol{z}, t) = \mathbb{E}\left[X^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right],
$$

rather than the full denoising posterior conditioned on the local observation $Z^i = \boldsymbol{z}^i$. In particular, they show that the UDM NELBO is minimized when the plug-in predictor satisfies

$$
\boldsymbol{x}_{\theta^\star}^i(\boldsymbol{z}, t) = \boldsymbol{x}_\star^i(\boldsymbol{z}, t)
$$

for $P_{T,Z}$-almost every $(t, \boldsymbol{z})$. The local observation $\boldsymbol{z}^i$ can then be incorporated analytically using the known uniform corruption likelihood. By Bayes' rule,

$$
\mathbb{E}\left[X^i \,\middle|\, Z = \boldsymbol{z}, T = t\right] = \frac{\boldsymbol{x}_\star^i(\boldsymbol{z}, t) \odot \left(\alpha_t \boldsymbol{z}^i + \frac{1-\alpha_t}{K}\mathbf{1}_K\right)}{\left\langle \boldsymbol{x}_\star^i(\boldsymbol{z}, t), \alpha_t \boldsymbol{z}^i + \frac{1-\alpha_t}{K}\mathbf{1}_K \right\rangle}.
$$

Thus, the LOO posterior summarizes the information contained in the remaining corrupted context Z^{-i}, while the observation at the target position is inserted exactly through the known forward

corruption process. Replacing the population-optimal LOO posterior $\boldsymbol{x}^i_\star$ by the model prediction $\boldsymbol{x}^i_\theta$ gives the corrected posterior denoiser $\boldsymbol{\mu}^{\mathrm{Posterior},i}_\theta$. Equivalently, using $\rho_t = K\alpha_t/(1-\alpha_t)$, this correction can be written in logit form as

$$\begin{aligned}\boldsymbol{\mu}^{\mathrm{Posterior},i}_\theta(\boldsymbol{z},t) &= \mathrm{softmax}\left(\log \boldsymbol{x}^i_\theta(\boldsymbol{z},t) + \log\left(\alpha_t \boldsymbol{z}^i + \frac{1-\alpha_t}{K}\mathbf{1}_K\right)\right)\\ &= \mathrm{softmax}\left(\log \boldsymbol{x}^i_\theta(\boldsymbol{z},t) + \log(1+\rho_t)\boldsymbol{z}^i\right). \end{aligned} \tag{11}$$

The second equality follows because the uniform-channel likelihood differs between the observed token and all other tokens only by the multiplicative factor $1+\rho_t$, while any common additive logit term vanishes under $\mathrm{softmax}$. This corrected posterior can be trained using the usual denoising cross-entropy,

$$\mathcal{L}_{\mathrm{LOO+CE}}(\theta) = -\int_0^1 \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}_t} \sum_{i=1}^{L} \log\left\langle \boldsymbol{x}^i, \boldsymbol{\mu}^{\mathrm{Posterior},i}_\theta(\boldsymbol{z}_t,t)\right\rangle \mathrm{d}t.$$

At the unrestricted population optimum, the corrected posterior satisfies $\boldsymbol{\mu}^{\mathrm{Posterior},i}_{\theta^\star}(\boldsymbol{z},t) = \mathbb{E}[X^i \mid Z=\boldsymbol{z}, T=t]$, and, crucially, the underlying predictor satisfies

$$\boldsymbol{x}^i_{\theta^\star}(\boldsymbol{z},t) = \boldsymbol{x}^i_\star(\boldsymbol{z},t) = \mathbb{E}\left[X^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T=t\right].$$

Thus, the corrected cross-entropy objective provides a convenient way to train the same LOO predictor required by the UDM plug-in parameterization. Once trained, $\boldsymbol{x}_\theta$ is plugged into the original UDM bridge $p^\theta_{s|t}$ to define the reverse transitions used for generation.

B Omitted Proofs

In this section, we provide omitted proofs in the main paper. Throughout the proofs, we consider $0 < t < 1$ with $0 < \alpha_t < 1$. Conditional expectations on events of zero data probability are assigned arbitrary, time-independent probability vectors, since their corresponding mixture weights are zero.

B.1 Proof of Proposition 1

Proposition 1 (Explicit time decomposition of the population-optimal UDM predictor)**.** *Fix $i \in [L]$ and a corrupted observation $\boldsymbol{z}$. For any index set $S \subseteq [L]\setminus\{i\}$, write $\boldsymbol{z}[S] \coloneqq \left(\boldsymbol{z}^j\right)_{j\in S}$, and $X[S] \coloneqq \left(X^j\right)_{j\in S}$. Then the population-optimal predictor satisfies*

$$\begin{aligned}\boldsymbol{x}^i_\star(\boldsymbol{z},t) &= \sum_{S\subseteq[L]\setminus\{i\}} \Pr(S \mid Z^{-i} = \boldsymbol{z}^{-i}, T=t)\,\mathbb{E}\left[X^i \,\middle|\, X[S] = \boldsymbol{z}[S]\right], \\ \Pr(S \mid Z^{-i} = \boldsymbol{z}^{-i}, T=t) &\propto \rho_t^{|S|}\Pr(X[S] = \boldsymbol{z}[S]).\end{aligned} \tag{5}$$

Here, $\Pr(X[S] = \boldsymbol{z}[S]) = \sum_{\boldsymbol{x}:\,\boldsymbol{x}[S]=\boldsymbol{z}[S]} p_{\mathrm{data}}(\boldsymbol{x})$ is the data probability mass of the partial observation $\boldsymbol{z}[S]$ under the hypothesis index set S.

Proof. Gourevitch et al. (2026) showed that the unrestricted population minimizer of the bridge plug-in UDM NELBO is the leave-one-out posterior. Therefore,

$$\boldsymbol{x}^i_{\theta^\star}(\boldsymbol{z},t) = \boldsymbol{x}^i_\star(\boldsymbol{z},t) = \mathbb{E}\left[X^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T=t\right].$$

By Bayes' rule and marginalization,

$$\boldsymbol{x}^i_\star(\boldsymbol{z},t) = \frac{\mathbb{E}_{p_{\mathrm{data}}}\left[X^i q_t\left(\boldsymbol{z}^{-i} \mid X^{-i}\right)\right]}{\mathbb{E}_{p_{\mathrm{data}}}\left[q_t(\boldsymbol{z}^{-i} \mid X^{-i})\right]}. \tag{12}$$

Since the uniform forward process factorizes across positions,

$$\begin{aligned} q_t\big(\boldsymbol{z}^{-i} \mid X^{-i}\big) &= \prod_{j\in[L]\setminus\{i\}} \left(\alpha_t \mathbb{1}\big\{X^j = \boldsymbol{z}^j\big\} + \frac{1-\alpha_t}{K}\right) \\ &= \left(\frac{1-\alpha_t}{K}\right)^{L-1} \prod_{j\in[L]\setminus\{i\}} \left(1 + \rho_t \mathbb{1}\big\{X^j = \boldsymbol{z}^j\big\}\right) \\ &= \left(\frac{1-\alpha_t}{K}\right)^{L-1} \sum_{S\subseteq[L]\setminus\{i\}} \rho_t^{|S|} \mathbb{1}\{X[S] = \boldsymbol{z}[S]\}\,, \end{aligned} \tag{13}$$

where the last equality follows by expanding the product over all subsets of $[L] \setminus \{i\}$.

Substituting Eq. (13) into Eq. (12) and canceling the common factor $\big((1-\alpha_t)/K\big)^{L-1}$ gives

$$\boldsymbol{x}^i_\star(\boldsymbol{z}, t) = \frac{\displaystyle\sum_{S\subseteq[L]\setminus\{i\}} \rho_t^{|S|} \mathbb{E}_{p_{\text{data}}}\left[X^i \mathbb{1}\{X[S] = \boldsymbol{z}[S]\}\right]}{\displaystyle\sum_{S\subseteq[L]\setminus\{i\}} \rho_t^{|S|} \Pr\left(X[S] = \boldsymbol{z}[S]\right)}.$$

Using

$$\mathbb{E}_{p_{\text{data}}}\left[X^i \mathbb{1}\{X[S] = \boldsymbol{z}[S]\}\right] = \Pr\left(X[S] = \boldsymbol{z}[S]\right) \mathbb{E}\left[X^i \,\middle|\, X[S] = \boldsymbol{z}[S]\right],$$

we obtain

$$\boldsymbol{x}^i_\star(\boldsymbol{z}, t) = \sum_{S\subseteq[L]\setminus\{i\}} \frac{\rho_t^{|S|} \Pr\left(X[S] = \boldsymbol{z}[S]\right)}{\displaystyle\sum_{S'\subseteq[L]\setminus\{i\}} \rho_t^{|S'|} \Pr\left(X[S'] = \boldsymbol{z}[S']\right)} \mathbb{E}\left[X^i \,\middle|\, X[S] = \boldsymbol{z}[S]\right]. \tag{14}$$

It remains to identify the normalized coefficient in Eq. (14). Interpret S as the set of context positions generated through the copy branch of the uniform forward channel. For a fixed S, the joint probability of this copy-set hypothesis and the observation $Z^{-i} = \boldsymbol{z}^{-i}$ is

$$\begin{aligned} \Pr\left(S,\, Z^{-i} = \boldsymbol{z}^{-i} \,\middle|\, T = t\right) &= \alpha_t^{|S|} \left(\frac{1-\alpha_t}{K}\right)^{L-1-|S|} \Pr\left(X[S] = \boldsymbol{z}[S]\right) \\ &= \left(\frac{1-\alpha_t}{K}\right)^{L-1} \rho_t^{|S|} \Pr\left(X[S] = \boldsymbol{z}[S]\right). \end{aligned}$$

Therefore, normalizing over all $S \subseteq [L] \setminus \{i\}$ gives

$$\Pr\left(S \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) = \frac{\rho_t^{|S|} \Pr\left(X[S] = \boldsymbol{z}[S]\right)}{\displaystyle\sum_{S'\subseteq[L]\setminus\{i\}} \rho_t^{|S'|} \Pr\left(X[S'] = \boldsymbol{z}[S']\right)}. \tag{15}$$

Hence,

$$\Pr\left(S \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) \propto \rho_t^{|S|} \Pr\left(X[S] = \boldsymbol{z}[S]\right),$$

and substituting Eq. (15) into Eq. (14) yields

$$\boldsymbol{x}^i_{\theta^\star}(\boldsymbol{z}, t) = \sum_{S\subseteq[L]\setminus\{i\}} \Pr\left(S \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) \mathbb{E}\left[X^i \,\middle|\, X[S] = \boldsymbol{z}[S]\right],$$

which proves the claim. □

B.2 Proof of Theorem 1

Theorem 1 (Time dependence through the matched-context count)**.** *For every target position i,*

$$X^i \perp T \mid Z^{-i}, M^i,$$

i.e., X^i and T are conditionally independent given Z^{-i} and M^i. Moreover, Eq. (4) admits the exact decomposition

$$\boldsymbol{x}^i_\star(\boldsymbol{z}, t) = \sum_{m=0}^{L-1} w^i_m(\boldsymbol{z}, t)\mathbf{h}^i_m(\boldsymbol{z}),$$
$$w^i_m(\boldsymbol{z}, t) \coloneqq \Pr\left(M^i = m \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) \propto (1+\rho_t)^m \Pr\left(M^i(X, \boldsymbol{z}) = m\right),$$
$$\mathbf{h}^i_m(\boldsymbol{z}) \coloneqq \mathbb{E}\left[X^i \,\middle|\, M^i = m, Z^{-i} = \boldsymbol{z}^{-i}\right],$$

where $\Pr(M^i(X, \boldsymbol{z}) = m) = \sum_{M^i(\boldsymbol{x}, \boldsymbol{z}) = m} p_{\text{data}}(\boldsymbol{x})$ is the data probability mass of clean sequences that match $\boldsymbol{z}$ in exactly m context positions.

Proof. By the law of total expectation,

$$\boldsymbol{x}^i_\star(\boldsymbol{z}, t) = \sum_{m=0}^{L-1} \Pr\left(M^i = m \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) \times \mathbb{E}\left[X^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, M^i = m, T = t\right]. \tag{16}$$

Therefore, it remains to show that

$$\Pr\left(M^i = m \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) \propto (1+\rho_t)^m \Pr(M^i(X, \boldsymbol{z}) = m),$$

and

$$\mathbb{E}\left[X^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, M^i = m, T = t\right] = \mathbb{E}\left[X^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, M^i = m\right].$$

Posterior weighting. For any clean sequence $\boldsymbol{x}'$, the uniform forward channel gives

$$q_t\left(\boldsymbol{z}^{-i} \,\middle|\, \boldsymbol{x}'^{-i}\right) = \prod_{j \in [L]\setminus\{i\}} \left[\alpha_t \mathbb{1}\left\{\boldsymbol{x}'^j = \boldsymbol{z}^j\right\} + \frac{1-\alpha_t}{K}\right] = \left(\frac{1-\alpha_t}{K}\right)^{L-1} (1+\rho_t)^{M^i(\boldsymbol{x}', \boldsymbol{z})}. \tag{17}$$

Hence,

$$\begin{aligned}
\Pr\left(M^i = m, Z^{-i} = \boldsymbol{z}^{-i} \,\middle|\, T = t\right) &= \sum_{\substack{\boldsymbol{x}' \in \mathcal{X}^L \\ M^i(\boldsymbol{x}', \boldsymbol{z}) = m}} p_{\text{data}}(\boldsymbol{x}') q_t\left(\boldsymbol{z}^{-i} \,\middle|\, \boldsymbol{x}'^{-i}\right) \\
&= \left(\frac{1-\alpha_t}{K}\right)^{L-1} (1+\rho_t)^m \sum_{\substack{\boldsymbol{x}' \in \mathcal{X}^L \\ M^i(\boldsymbol{x}', \boldsymbol{z}) = m}} p_{\text{data}}(\boldsymbol{x}') \\
&= \left(\frac{1-\alpha_t}{K}\right)^{L-1} \Pr(M^i(X, \boldsymbol{z}) = m)(1+\rho_t)^m.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\Pr\left(M^i = m \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) &= \frac{\Pr\left(M^i = m, Z^{-i} = \boldsymbol{z}^{-i} \,\middle|\, T = t\right)}{\sum_{r=0}^{L-1} \Pr\left(M^i = r, Z^{-i} = \boldsymbol{z}^{-i} \,|\, T = t\right)} \\
&= \frac{\Pr(M^i(X, \boldsymbol{z}) = m)(1+\rho_t)^m}{\sum_{r=0}^{L-1} \Pr(M^i(X, \boldsymbol{z}) = r)(1+\rho_t)^r}.
\end{aligned}$$

and hence

$$\Pr\left(M^i = m \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) \propto \Pr(M^i(X, \boldsymbol{z}) = m)(1+\rho_t)^m.$$

Time-independence of the conditional prediction. For any $\boldsymbol{x}'$ satisfying $M^i(\boldsymbol{x}', \boldsymbol{z}) = m$, Bayes' rule and Eq. (17) give

$$\Pr\left(X = \boldsymbol{x}' \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, M^i = m, T = t\right) = \frac{p_{\text{data}}(\boldsymbol{x}')q_t\left(\boldsymbol{z}^{-i} \,\middle|\, \boldsymbol{x}'^{-i}\right)}{\sum_{\substack{\boldsymbol{x}'' \in \mathcal{X}^L \\ M^i(\boldsymbol{x}'', \boldsymbol{z}) = m}} p_{\text{data}}(\boldsymbol{x}'')q_t\left(\boldsymbol{z}^{-i} \,\middle|\, \boldsymbol{x}''^{-i}\right)}$$
$$= \frac{p_{\text{data}}(\boldsymbol{x}')}{\sum_{\substack{\boldsymbol{x}'' \in \mathcal{X}^L \\ M^i(\boldsymbol{x}'', \boldsymbol{z}) = m}} p_{\text{data}}(\boldsymbol{x}'')} = \frac{p_{\text{data}}(\boldsymbol{x}')}{\Pr(M^i(X, \boldsymbol{z}) = m)},$$

where the second equality follows because all sequences satisfying $M^i(\boldsymbol{x}', \boldsymbol{z}) = m$ have the same uniform-channel likelihood, so the common factor cancels under normalization.

The right-hand side is independent of t. Therefore,

$$X^i \perp T \mid Z^{-i}, M^i,$$

and consequently

$$\mathbb{E}\left[X^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, M^i = m, T = t\right] = \mathbb{E}\left[X^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, M^i = m\right] = \mathbf{h}^i_m(\boldsymbol{z}).$$

Substituting the two results into Eq. (16) gives

$$\boldsymbol{x}^i_\star(\boldsymbol{z}, t) = \sum_{m=0}^{L-1} w^i_m(\boldsymbol{z}, t)\mathbf{h}^i_m(\boldsymbol{z}),$$

with

$$w^i_m(\boldsymbol{z}, t) \propto \Pr(M^i(X, \boldsymbol{z}) = m)(1 + \rho_t)^m,$$

which proves the claim. □

B.3 Proof of Proposition 2

Proposition 2 (Exact temporal sensitivity)**.** *Let $\eta_t = \log(1 + \rho_t)$ and let $\boldsymbol{e}_k$ denote the k-th one-hot vector. Then the temporal derivative of the context posterior is*

$$\frac{\partial}{\partial \eta_t}\boldsymbol{x}^i_\star(\boldsymbol{z}, t) = \text{Cov}_{t,\boldsymbol{z}}\left(\mathbf{h}^i_{M^i}(\boldsymbol{z}), M^i\right),$$

where the covariance is taken componentwise. Equivalently, for any candidate category k with $\boldsymbol{x}^i_{\star,k}(\boldsymbol{z}, t) > 0$,

$$\frac{\partial}{\partial \eta_t} \log \boldsymbol{x}^i_{\star,k}(\boldsymbol{z}, t) = \mathbb{E}_{t,\boldsymbol{z}}\left[M^i \,\middle|\, X^i = \boldsymbol{e}_k\right] - \mathbb{E}_{t,\boldsymbol{z}}[M^i], \tag{6}$$

where the subscript $(t, \boldsymbol{z})$ denotes the posterior distribution $\Pr(\cdot \mid Z^{-i} = \boldsymbol{z}^{-i}, T = t)$.

Proof. By Theorem 1,

$$\boldsymbol{x}^i_\star(\boldsymbol{z}, t) = \sum_{m=0}^{L-1} w^i_m(\boldsymbol{z}, t)\mathbf{h}^i_m(\boldsymbol{z}),$$

where, writing $\eta_t = \log(1 + \rho_t)$,

$$w^i_m(\boldsymbol{z}, t) = \frac{\Pr(M^i(X, \boldsymbol{z}) = m)e^{m\eta_t}}{\sum_{r=0}^{L-1} \Pr(M^i(X, \boldsymbol{z}) = r)e^{r\eta_t}}. \tag{18}$$

Since $\mathbf{h}^i_m(\boldsymbol{z})$ is independent of t, all dependence on η_t is contained in $w^i_m(\boldsymbol{z}, t)$. Differentiating Eq. (18) gives

$$\begin{aligned}\frac{\partial}{\partial \eta_t} w^i_m(\boldsymbol{z}, t) &= w^i_m(\boldsymbol{z}, t) \left(m - \sum_{r=0}^{L-1} r\, w^i_r(\boldsymbol{z}, t) \right) \\ &= w^i_m(\boldsymbol{z}, t) \left(m - \mathbb{E}_{t,\boldsymbol{z}}[M^i] \right).\end{aligned}$$

Therefore,

$$\begin{aligned}\frac{\partial}{\partial \eta_t} \boldsymbol{x}^i_\star(\boldsymbol{z}, t) &= \sum_{m=0}^{L-1} \mathbf{h}^i_m(\boldsymbol{z}) w^i_m(\boldsymbol{z}, t) \left(m - \mathbb{E}_{t,\boldsymbol{z}}[M^i] \right) \\ &= \mathbb{E}_{t,\boldsymbol{z}} \left[\mathbf{h}^i_{M^i}(\boldsymbol{z}) M^i \right] - \mathbb{E}_{t,\boldsymbol{z}} \left[\mathbf{h}^i_{M^i}(\boldsymbol{z}) \right] \mathbb{E}_{t,\boldsymbol{z}}[M^i] \\ &= \mathrm{Cov}_{t,\boldsymbol{z}} \left(\mathbf{h}^i_{M^i}(\boldsymbol{z}), M^i \right),\end{aligned}$$

which proves the claim.

Now fix a candidate token $\boldsymbol{e}_k$ such that $\boldsymbol{x}^i_{\star,k}(\boldsymbol{z}, t) > 0$. Taking the k-th component of the previous expression,

$$\frac{\partial}{\partial \eta_t} \log \boldsymbol{x}^i_{\star,k}(\boldsymbol{z}, t) = \frac{\sum_{m=0}^{L-1} w^i_m(\boldsymbol{z}, t) h^i_{m,k}(\boldsymbol{z}) m}{\sum_{m=0}^{L-1} w^i_m(\boldsymbol{z}, t) h^i_{m,k}(\boldsymbol{z})} - \mathbb{E}_{t,\boldsymbol{z}}[M^i], \tag{19}$$

where $h^i_{m,k}(\boldsymbol{z})$ denotes the k-th component of $\mathbf{h}^i_m(\boldsymbol{z})$.

By Theorem 1, $X^i \perp T \mid Z^{-i}, M^i$, and hence we can include t into the distribution:

$$h^i_{m,k}(\boldsymbol{z}) = \Pr\left(X^i = \boldsymbol{e}_k \,\middle|\, M^i = m, Z^{-i} = \boldsymbol{z}^{-i}, T = t \right).$$

Together with

$$w^i_m(\boldsymbol{z}, t) = \Pr\left(M^i = m \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t \right),$$

Bayes' rule gives

$$\frac{w^i_m(\boldsymbol{z}, t) h^i_{m,k}(\boldsymbol{z})}{\boldsymbol{x}^i_{\star,k}(\boldsymbol{z}, t)} = \Pr\left(M^i = m \,\middle|\, X^i = \boldsymbol{e}_k, Z^{-i} = \boldsymbol{z}^{-i}, T = t \right).$$

Consequently, the first term in Eq. (19) is precisely

$$\mathbb{E}_{t,\boldsymbol{z}} \left[M^i \,\middle|\, X^i = \boldsymbol{e}_k \right],$$

and therefore

$$\frac{\partial}{\partial \eta_t} \log \boldsymbol{x}^i_{\star,k}(\boldsymbol{z}, t) = \mathbb{E}_{t,\boldsymbol{z}} \left[M^i \,\middle|\, X^i = \boldsymbol{e}_k \right] - \mathbb{E}_{t,\boldsymbol{z}}[M^i].$$

This proves Eq. (6). □

B.4 Proof of Proposition 3

We here provide a formal version of Proposition 3 in the main paper and the proof under the empirical data distribution setting.

Assumption 1 (Finite-data setting)**.** *The data distribution is the uniform empirical distribution*[1] $\hat{p}_N = \frac{1}{N} \sum_{n=1}^{N} \delta_{\boldsymbol{x}_n}$ *over* N *sequences* $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathcal{X}^L$ *where* $|\mathcal{X}| = K$.

[1] Note that every theoretical result in our paper holds within Assumption 1; see Appendix B.7 for why.

Proposition 3 (Δ-separable event under UDM training)**.** *Under Assumption 1, fix $\boldsymbol{x} \in \mathrm{supp}(\hat{p}_N)$ and $i \in [L]$, and suppose $d(\boldsymbol{x}) \geq 2$. Let $T \sim \mathrm{Unif}([0,1])$ and, conditioned on $T = t$, let $Z \sim q_t(\cdot \mid \boldsymbol{x})$ under the linear schedule $\alpha_t = 1 - t$. Then, for any integer $\Delta \geq 1$,*

$$\Pr(Z \textit{ is } \Delta\textit{-separable w.r.t. } \boldsymbol{x} \textit{ at } i) \geq \max\left\{0,\, 1 - \frac{\log(N-1) + \Delta + (e + e^{-1} - 2)\frac{L-1}{K}}{(1 - e^{-1})(d(\boldsymbol{x}) - 1)}\right\}. \tag{20}$$

Thus, as given in the informal statement of the main paper, if $K \gg L$ and $d(\boldsymbol{x})$ is sufficiently large, the approximate bound

$$\Pr(Z \text{ is } \Delta\text{-separable w.r.t. } \boldsymbol{x} \text{ at } i) \gtrsim 1 - \frac{\log N + \Delta}{(1 - e^{-1})d(\boldsymbol{x})}$$

holds. Note that we used Eq. (20) to measure quantitative bound in Section 4.2. We now provide the proof of formal statement:

Proof. Fix any $\boldsymbol{x}' \in \mathrm{supp}(\hat{p}_N)$ satisfying $\boldsymbol{x}'^i \neq \boldsymbol{x}^i$, and let

$$\mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}') \coloneqq \left\{j \in [L] \setminus \{i\} : \boldsymbol{x}'^j \neq \boldsymbol{x}^j\right\}.$$

Since $\boldsymbol{x}'^i \neq \boldsymbol{x}^i$ and the Hamming distance between $\boldsymbol{x}$ and $\boldsymbol{x}'$ is at least $d(\boldsymbol{x})$,

$$|\mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')| = \sum_{j \in [L]\setminus\{i\}} \mathbb{1}\left\{\boldsymbol{x}'^j \neq \boldsymbol{x}^j\right\} \geq d(\boldsymbol{x}) - 1. \tag{21}$$

Condition on $T = t$. For each $j \in \mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')$, define

$$Y_j \coloneqq \mathbb{1}\left\{Z^j = \boldsymbol{x}^j\right\} - \mathbb{1}\left\{Z^j = \boldsymbol{x}'^j\right\}.$$

For $j \notin \mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')$, $\boldsymbol{x}$ and $\boldsymbol{x}'$ contribute equally to the match count. Therefore,

$$M^i(\boldsymbol{x}, Z) - M^i(\boldsymbol{x}', Z) = \sum_{j \in \mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')} Y_j. \tag{22}$$

Under the linear schedule $\alpha_t = 1 - t$, the uniform forward kernel gives, for every $j \in \mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')$,

$$\begin{aligned}
\Pr(Y_j = 1 \mid T = t) &= 1 - t + \frac{t}{K}, \\
\Pr(Y_j = 0 \mid T = t) &= \frac{t(K-2)}{K}, \\
\Pr(Y_j = -1 \mid T = t) &= \frac{t}{K}.
\end{aligned}$$

Moreover, the variables $\{Y_j\}_{j \in \mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')}$ are independent conditioned on $T = t$. Hence,

$$\begin{aligned}
\mathbb{E}\left[e^{-Y_j} \mid T = t\right] &= \left(1 - t + \frac{t}{K}\right)e^{-1} + \frac{t(K-2)}{K} + \frac{t}{K}e \\
&= 1 - (1 - e^{-1})(1 - t) + (e + e^{-1} - 2)\frac{t}{K} \\
&\leq \exp\left(-(1 - e^{-1})(1 - t) + (e + e^{-1} - 2)\frac{t}{K}\right),
\end{aligned} \tag{23}$$

where the last inequality comes from $1 + u \leq e^u$.

The sequence Z fails to be Δ-separable from this particular $\boldsymbol{x}'$ whenever

$$M^i(\boldsymbol{x}', Z) > M^i(\boldsymbol{x}, Z) - \Delta.$$

Since the match counts are integer-valued, this is equivalent to

$$M^i(\boldsymbol{x}, Z) - M^i(\boldsymbol{x}', Z) \leq \Delta - 1.$$

Therefore, by Markov's inequality and Eq. (22),

$$\begin{aligned}
&\Pr\left(M^i(\boldsymbol{x}, Z) - M^i(\boldsymbol{x}', Z) \le \Delta - 1 \,\middle|\, T = t\right) \\
&\quad = \Pr\left(\exp\big(-\big(M^i(\boldsymbol{x}, Z) - M^i(\boldsymbol{x}', Z)\big)\big) \ge e^{-(\Delta-1)} \,\middle|\, T = t\right) \\
&\quad \le e^{\Delta-1}\mathbb{E}\left[\exp\big(-\big(M^i(\boldsymbol{x}, Z) - M^i(\boldsymbol{x}', Z)\big)\big) \,|\, T = t\right] \\
&\quad = e^{\Delta-1}\mathbb{E}\left[\exp\left(-\sum_{j \in \mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')} Y_j\right) \,\middle|\, T = t\right] \\
&\quad = e^{\Delta-1} \prod_{j \in \mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')} \mathbb{E}\left[e^{-Y_j} \mid T = t\right],
\end{aligned}$$

Plugging Eq. (23) into the above inequality gives:

$$\begin{aligned}
&\Pr\left(M^i(\boldsymbol{x}, Z) - M^i(\boldsymbol{x}', Z) \le \Delta - 1 \,\middle|\, T = t\right) \\
&\quad \le e^{\Delta-1} \prod_{j \in \mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')} \exp\left(-(1 - e^{-1})(1 - t) + (e + e^{-1} - 2)\frac{t}{K}\right) \\
&\quad = \exp\left(\Delta - 1 - (1 - e^{-1})(1 - t)\,|\mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')| + (e + e^{-1} - 2)\frac{t\,|\mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')|}{K}\right) \\
&\quad \le \exp\left(\Delta - 1 - (1 - e^{-1})(1 - t)(d(\boldsymbol{x}) - 1) + (e + e^{-1} - 2)\frac{L - 1}{K}\right).
\end{aligned}$$

where the last inequality uses Eq. (21), $t \le 1$, and $|\mathcal{H}_i(\boldsymbol{x}, \boldsymbol{x}')| \le L - 1$.

There are at most $N - 1$ empirical sequences distinct from $\boldsymbol{x}$. Thus, by a union bound over all $\boldsymbol{x}' \in \mathrm{supp}(\hat{p}_N)$ satisfying $\boldsymbol{x}'^i \ne \boldsymbol{x}^i$,

$$\begin{aligned}
&\Pr\left(Z \text{ is not } \Delta\text{-separable w.r.t. } \boldsymbol{x} \text{ at } i \,|\, T = t\right) \\
&\quad \le \min\left\{1,\ (N - 1)\exp\left(\Delta - 1 - (1 - e^{-1})(1 - t)(d(\boldsymbol{x}) - 1) + (e + e^{-1} - 2)\frac{L - 1}{K}\right)\right\}.
\end{aligned} \tag{24}$$

It remains to average over the training time $T \sim \mathrm{Unif}([0, 1])$. For any $A \ge 1$ and $B > 0$,

$$\int_0^1 \min\left\{1,\ Ae^{-B(1-t)}\right\} \mathrm{d}t \le \min\left\{1,\ \frac{\log A + 1}{B}\right\}. \tag{25}$$

Indeed, if $\log A \ge B$, the bound follows immediately since the integral is at most one. Otherwise, splitting the integral at $t = 1 - \log A/B$ gives

$$\int_0^1 \min\left\{1,\ Ae^{-B(1-t)}\right\} \mathrm{d}t = \frac{\log A}{B} + \frac{1 - Ae^{-B}}{B} \le \frac{\log A + 1}{B}.$$

Applying Eq. (25) to Eq. (24) with

$$A = (N - 1)\exp\left(\Delta - 1 + (e + e^{-1} - 2)\frac{L - 1}{K}\right)$$

and

$$B = (1 - e^{-1})(d(\boldsymbol{x}) - 1),$$

we obtain

$$\Pr\left(Z \text{ is not } \Delta\text{-separable w.r.t. } \boldsymbol{x} \text{ at } i\right) \le \min\left\{1,\ \frac{\log(N - 1) + \Delta + (e + e^{-1} - 2)\frac{L-1}{K}}{(1 - e^{-1})(d(\boldsymbol{x}) - 1)}\right\}.$$

Taking the complement yields

$$\Pr\left(Z \text{ is } \Delta\text{-separable w.r.t. } \boldsymbol{x} \text{ at } i\right) \ge \max\left\{0,\ 1 - \frac{\log(N - 1) + \Delta + (e + e^{-1} - 2)\frac{L-1}{K}}{(1 - e^{-1})(d(\boldsymbol{x}) - 1)}\right\},$$

which proves the claim. □

B.5 Proof of Theorem 2

We here provide a formal version of Theorem 2 in the main paper under the empirical data distribution setting. We first define the Fisher sensitivity with respect to the noise-schedule parameter α_t:

$$\mathcal{I}^i_{\alpha_t}(\boldsymbol{z}) \coloneqq \sum_{k:\, \boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t)>0} \boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t) \left(\frac{\partial}{\partial \alpha_t} \log \boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t)\right)^2 .$$

Theorem 2 (Fisher sensitivity with finite data)**.** *Suppose that α_t is strictly decreasing in t. Under the finite data assumption 1, for Δ-separable $\boldsymbol{z}$ with respect to $\boldsymbol{x}$ at coordinate i,*

$$\mathcal{I}^i_{\alpha_t}(\boldsymbol{z}) \le (N-1)(L-1)^2 K^2 \frac{(1-\alpha_t)^{\Delta-2}}{(1+(K-1)\alpha_t)^{\Delta+2}}.$$

Moreover, for $\Delta \ge 2$, the right-hand side is strictly increasing in $t \in (0,1)$.

The bound admits the same interpretation as its time-based counterpart. What matters for denoising is how much the oracle prediction changes per unit change in the corruption level, which is measured directly by $\mathcal{I}^i_{\alpha_t}(\boldsymbol{z})$. The theorem shows that as the corruption level increases (i.e., as α_t decreases), the upper bound on this sensitivity becomes progressively weaker. Thus, time dependence is most weakly controlled in the high-noise regime.

For ease of interpretation, the main text specializes this result to the linear schedule $\alpha_t = 1 - t$ and expresses the Fisher sensitivity with respect to t. Under this schedule, the theorem reduces to

$$\mathcal{I}^i_t(\boldsymbol{z}) \le (N-1)(L-1)^2 \frac{K^2 t^{\Delta-2}}{\big(K-(K-1)t\big)^{\Delta+2}}.$$

Using $(N-1)(L-1)^2 < NL^2$ gives the simpler bound stated in the main paper:

$$\mathcal{I}^i_t(\boldsymbol{z}) \le NL^2 \frac{K^2 t^{\Delta-2}}{\big(K-(K-1)t\big)^{\Delta+2}}.$$

We now prove the formal version of Theorem 2.

Proof. By Proposition 2, with $\eta_t = \log(1+\rho_t)$, for every k such that $\boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t) > 0$,

$$\frac{\partial}{\partial \alpha_t} \log \boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t) = \frac{\mathrm{d}\eta_t}{\mathrm{d}\alpha_t} \left(\mathbb{E}_{t,\boldsymbol{z}}\left[M^i \,\middle|\, X^i = \boldsymbol{e}_k\right] - \mathbb{E}_{t,\boldsymbol{z}}[M^i]\right).$$

Therefore, by the definition of $\mathcal{I}^i_{\alpha_t}(\boldsymbol{z})$,

$$\begin{aligned}
\mathcal{I}^i_{\alpha_t}(\boldsymbol{z}) &= \left(\frac{\mathrm{d}\eta_t}{\mathrm{d}\alpha_t}\right)^2 \sum_{k:\, \boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t)>0} \boldsymbol{x}^i_{\star,k}(\boldsymbol{z},t) \left(\mathbb{E}_{t,\boldsymbol{z}}\left[M^i \,\middle|\, X^i = \boldsymbol{e}_k\right] - \mathbb{E}_{t,\boldsymbol{z}}[M^i]\right)^2 \\
&= \left(\frac{\mathrm{d}\eta_t}{\mathrm{d}\alpha_t}\right)^2 \mathbb{E}_{t,\boldsymbol{z}}\left[\left(\mathbb{E}_{t,\boldsymbol{z}}\left[M^i \,\middle|\, X^i\right] - \mathbb{E}_{t,\boldsymbol{z}}[M^i]\right)^2\right] \\
&= \left(\frac{\mathrm{d}\eta_t}{\mathrm{d}\alpha_t}\right)^2 \mathrm{Var}_{t,\boldsymbol{z}}\left(\mathbb{E}_{t,\boldsymbol{z}}\left[M^i \,\middle|\, X^i\right]\right). \qquad (26)
\end{aligned}$$

For any choice of center, the variance is upper bounded by the corresponding mean squared deviation. We therefore choose the conditional mean associated with the source token $\boldsymbol{x}^i$, $\mathbb{E}_{t,\boldsymbol{z}}[M^i \mid X^i = \boldsymbol{x}^i]$, where $\boldsymbol{x}$ is the clean sequence from which $\boldsymbol{z}$ was sampled. This choice is convenient because the deviation vanishes whenever $X^i = \boldsymbol{x}^i$. Moreover, since $0 \le M^i \le L-1$, all conditional expectations below lie in $[0, L-1]$. Hence,

$$\begin{aligned}
\mathrm{Var}_{t,\boldsymbol{z}}\left(\mathbb{E}_{t,\boldsymbol{z}}\left[M^i \,\middle|\, X^i\right]\right) &\le \mathbb{E}_{t,\boldsymbol{z}}\left[\left(\mathbb{E}_{t,\boldsymbol{z}}\left[M^i \,\middle|\, X^i\right] - \mathbb{E}_{t,\boldsymbol{z}}\left[M^i \,\middle|\, X^i = \boldsymbol{x}^i\right]\right)^2\right] \\
&\le (L-1)^2 \Pr\left(X^i \ne \boldsymbol{x}^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right). \qquad (27)
\end{aligned}$$

It remains to bound the posterior probability of a different target token. Under Assumption 1,

$$\hat{p}_N = \frac{1}{N}\sum_{n=1}^{N} \delta_{\boldsymbol{x}_n}.$$

From Eq. (17),

$$q_t\left(\boldsymbol{z}^{-i} \,\middle|\, \boldsymbol{x}_n^{-i}\right) = \left(\frac{1-\alpha_t}{K}\right)^{L-1} (1+\rho_t)^{M^i(\boldsymbol{x}_n, \boldsymbol{z})}.$$

Hence, since the empirical prior is uniform,

$$\Pr\left(X^i \neq \boldsymbol{x}^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) = \frac{\sum_{n:\, \boldsymbol{x}_n^i \neq \boldsymbol{x}^i} (1+\rho_t)^{M^i(\boldsymbol{x}_n, \boldsymbol{z})}}{\sum_{r=1}^{N} (1+\rho_t)^{M^i(\boldsymbol{x}_r, \boldsymbol{z})}}. \tag{28}$$

By Δ-separability, every sequence satisfying $\boldsymbol{x}_n^i \neq \boldsymbol{x}^i$ obeys

$$M^i(\boldsymbol{x}_n, \boldsymbol{z}) \leq M^i(\boldsymbol{x}, \boldsymbol{z}) - \Delta.$$

Moreover, the denominator of Eq. (28) contains the contribution of the sampled sequence $\boldsymbol{x}$. Thus,

$$\begin{aligned}\Pr\left(X^i \neq \boldsymbol{x}^i \,\middle|\, Z^{-i} = \boldsymbol{z}^{-i}, T = t\right) &\leq \frac{(N-1)(1+\rho_t)^{M^i(\boldsymbol{x}, \boldsymbol{z})-\Delta}}{(1+\rho_t)^{M^i(\boldsymbol{x}, \boldsymbol{z})}} \\ &= (N-1)(1+\rho_t)^{-\Delta}. \end{aligned} \tag{29}$$

Combining Eq. (26), Eq. (27), and Eq. (29) yields

$$\mathcal{I}^i_{\alpha_t}(\boldsymbol{z}) \leq (N-1)(L-1)^2 \left(\frac{\mathrm{d}}{\mathrm{d}\alpha_t} \log(1+\rho_t)\right)^2 (1+\rho_t)^{-\Delta}. \tag{30}$$

Using

$$\rho_t = \frac{K\alpha_t}{1-\alpha_t},$$

we have

$$1 + \rho_t = \frac{1+(K-1)\alpha_t}{1-\alpha_t}, \tag{31}$$

$$\frac{\mathrm{d}}{\mathrm{d}\alpha_t} \log(1+\rho_t) = \frac{K}{(1-\alpha_t)\,(1+(K-1)\alpha_t)}. \tag{32}$$

Substituting Eq. (31) and Eq. (32) into Eq. (30) gives

$$\mathcal{I}^i_{\alpha_t}(\boldsymbol{z}) \leq (N-1)(L-1)^2 K^2 \frac{(1-\alpha_t)^{\Delta-2}}{(1+(K-1)\alpha_t)^{\Delta+2}}. \tag{33}$$

It remains to establish the monotonicity of the bound. Define

$$g(\alpha) \coloneqq \frac{(1-\alpha)^{\Delta-2}}{(1+(K-1)\alpha)^{\Delta+2}}.$$

For $\Delta \geq 2$ and $\alpha \in (0,1)$,

$$\frac{\mathrm{d}}{\mathrm{d}\alpha} \log g(\alpha) = -\frac{\Delta-2}{1-\alpha} - \frac{(\Delta+2)(K-1)}{1+(K-1)\alpha} < 0.$$

Hence, $g(\alpha)$ is strictly decreasing in α. Since α_t is strictly decreasing in t, $g(\alpha_t)$ is strictly increasing in t. Therefore, the right-hand side of Eq. (33) is strictly increasing in $t \in (0,1)$ for $\Delta \geq 2$, which proves the claim. □

B.6 Proof of Eq. (9)

Proof. By Theorem 2,

$$\mathcal{I}^i_{\alpha_t}(\boldsymbol{z}) \le (N-1)(L-1)^2K^2\frac{(1-\alpha_t)^{\Delta-2}}{(1+(K-1)\alpha_t)^{\Delta+2}}.$$

For $\Delta \ge 2$, we have

$$(1-\alpha_t)^{\Delta-2} \le 1$$

for every $\alpha_t \in (0,1)$. Moreover, suppose

$$\alpha_t \ge K^{-\beta}.$$

Then

$$\begin{aligned} 1+(K-1)\alpha_t &\ge 1+(K-1)K^{-\beta} \\ &= K^{1-\beta}+1-K^{-\beta} \\ &\ge K^{1-\beta}. \end{aligned}$$

Consequently,

$$\begin{aligned} \mathcal{I}^i_{\alpha_t}(\boldsymbol{z}) &\le (N-1)(L-1)^2K^{2-(1-\beta)(\Delta+2)} \\ &= (N-1)(L-1)^2K^{-\Delta+\beta(\Delta+2)}. \end{aligned}$$

Taking the supremum over all t satisfying $\alpha_t \ge K^{-\beta}$ gives

$$\sup_{t:\,\alpha_t \ge K^{-\beta}} \mathcal{I}^i_{\alpha_t}(\boldsymbol{z}) \le (N-1)(L-1)^2K^{-\Delta+\beta(\Delta+2)}.$$

The exponent is negative whenever

$$0 < \beta < \frac{\Delta}{\Delta+2}.$$

Thus, for any $0 < \beta < \frac{\Delta}{\Delta+2}$ and $\Delta \ge 2$, the Fisher sensitivity is polynomially suppressed throughout the region $\alpha_t \ge K^{-\beta}$.

Choosing $\beta = 1/4$, which is valid for every $\Delta \ge 2$, gives

$$\sup_{t:\,\alpha_t \ge K^{-1/4}} \mathcal{I}^i_{\alpha_t}(\boldsymbol{z}) \le (N-1)(L-1)^2K^{-(3\Delta-2)/4}.$$

This provides a useful intuition for a general schedule α_t. If $\boldsymbol{z}$ is Δ-separable with $\Delta \ge 2$, its Fisher sensitivity is suppressed by a negative power of K whenever the expected clean-token fraction exceeds $K^{-1/4}$. Thus, irrespective of the particular scheduler, a Δ-separable $\boldsymbol{z}$ has low empirical-optimal Fisher sensitivity with respect to the noise coordinate throughout this regime. Finally, under the linear schedule $\alpha_t = 1-t$, the condition $\alpha_t \ge K^{-\beta}$ is equivalent to

$$0 < t \le 1-K^{-\beta},$$

and $\mathcal{I}^i_{\alpha_t}(\boldsymbol{z}) = \mathcal{I}^i_t(\boldsymbol{z})$. Therefore,

$$\sup_{0<t\le 1-K^{-1/4}} \mathcal{I}^i_t(\boldsymbol{z}) \le NL^2K^{-(3\Delta-2)/4},$$

where we intentionally upper-bound $(N-1)(L-1)^2 < NL^2$ to give a simpler and more intuitive bound in the main paper. □

B.7 Discussion on the Finite Data Setting

In Section 4, we specialize the data distribution to the finite empirical distribution $\hat{p}_N$ under Assumption 1. Here, we verify that this finite-data formulation is fully consistent with the theoretical results in previous works and in Section 3. For example, deriving the continuous-time UDM NELBO from its discrete-time counterpart could in principle rely on assumptions on the data distribution, such as

full support; here, we explicitly verify that the finite empirical distribution does not invalidate these results. Specifically, we show that (i) the UDM NELBO remains a valid variational objective under $\hat{p}_N$, (ii) its unrestricted optimum remains the uniform-channel leave-one-out posterior characterized in Proposition 1, and consequently, (iii) Theorem 1 and Proposition 2 continue to hold under the finite empirical distribution.

Validity of the UDM NELBO under finite data. We first verify that the UDM NELBO remains a valid variational objective under the finite empirical distribution in Assumption 1. The derivation of the UDM NELBO (Schiff et al., 2025) is pointwise in the clean sequence $\boldsymbol{x}$ and does not require any particular form of the outer data distribution. In particular, for a finite discretization with T diffusion steps, define $t(i) = \frac{i+1}{T+1}$, $s(i) = \frac{i}{T+1}$, for $i = 0, \dots, T$. Then, for every $\boldsymbol{x} \in \mathcal{X}^L$,

$$-\log p_\theta(\boldsymbol{x}) \leq \mathbb{E}_q \Bigg[-\log p_\theta\big(\boldsymbol{x} \mid \boldsymbol{z}_{t(0)}\big) + \sum_{i=1}^{T} \mathcal{D}_{\mathrm{KL}}\big[q\big(\boldsymbol{z}_{s(i)} \mid \boldsymbol{z}_{t(i)}, \boldsymbol{x}\big) \,\big\|\, p_\theta\big(\boldsymbol{z}_{s(i)} \mid \boldsymbol{z}_{t(i)}\big)\big] \Bigg] + \mathcal{D}_{\mathrm{KL}}\big[q\big(\boldsymbol{z}_{t(T)} \mid \boldsymbol{x}\big) \,\big\|\, p_\theta\big(\boldsymbol{z}_{t(T)}\big)\big] .$$

Crucially, this bound holds pointwise for every clean sequence $\boldsymbol{x} \in \mathcal{X}^L$. Its derivation does not depend on the support, cardinality, or any other property of the data distribution from which $\boldsymbol{x}$ is drawn. Therefore, the discrete-time NELBO remains valid when the data distribution is replaced by the finite empirical distribution $\hat{p}_N$ in Assumption 1. We now analyze whether the continuous-time limit of the discrete-time NELBO derived by Schiff et al. (2025) holds in finite-data setting. Their derivation takes $T \to \infty$ for each fixed clean sequence $\boldsymbol{x}$. In particular, the prior loss vanishes because $\alpha_{t(T)} = 0$ implies

$$q\Big(\boldsymbol{z}^i_{t(T)} \mid \boldsymbol{x}\Big) = \mathrm{Cat}(\cdot; K^{-1}\mathbf{1}_K),$$

independently of $\boldsymbol{x}$, so that setting $p_\theta(\boldsymbol{z}^i_{t(T)}) = K^{-1}\mathbf{1}_K$ gives zero prior KL. Likewise, as $t(0) = 1/(T+1) \to 0$ and $\alpha_{t(0)} \to 1$,

$$q\Big(\boldsymbol{z}^i_{t(0)} \mid \boldsymbol{x}\Big) \longrightarrow \mathrm{Cat}(\cdot; \boldsymbol{x}^i),$$

and the endpoint denoising parameterization copies its input, making the reconstruction loss vanish. Neither argument depends on the support of the data distribution. Thus, the remaining diffusion term converges pointwise in $\boldsymbol{x}$ to

$$\mathcal{L}_{\mathrm{UDM}}(\boldsymbol{x}; \theta) = \int_0^1 \mathbb{E}_{\boldsymbol{z}_t \sim q_t(\cdot \mid \boldsymbol{x})} \sum_{i=1}^{L} \frac{\alpha_t'}{K\alpha_t} \left[\frac{K}{\bar{x}^i_{\ell_i}} - \frac{K}{\bar{x}^i_{\theta,\ell_i}} - \sum_{j \neq \ell_i} \frac{\bar{x}^i_j}{\bar{x}^i_{\ell_i}} \log \frac{\bar{x}^i_{\theta,\ell_i} \bar{x}^i_j}{\bar{x}^i_{\theta,j} \bar{x}^i_{\ell_i}} \right] \mathrm{d}t,$$

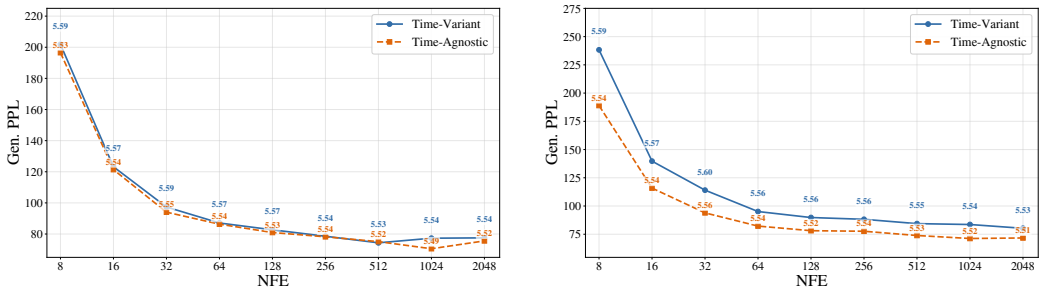
where $\bar{\boldsymbol{\nu}}^i = K\alpha_t \boldsymbol{\nu}^i + (1-\alpha_t)\mathbf{1}_K$ and ℓ_i denotes the non-zero dimension of $\boldsymbol{z}^i_t$. Taking expectation of $\boldsymbol{x}$ then gives:

$$\mathcal{L}_{\mathrm{UDM}}(\theta) = \int_0^1 \mathbb{E}_{\boldsymbol{x}, \boldsymbol{z}_t} \sum_{i=1}^{L} \frac{\alpha_t'}{K\alpha_t} \left[\frac{K}{\bar{x}^i_{\ell_i}} - \frac{K}{\bar{x}^i_{\theta,\ell_i}} - \sum_{j \neq \ell_i} \frac{\bar{x}^i_j}{\bar{x}^i_{\ell_i}} \log \frac{\bar{x}^i_{\theta,\ell_i} \bar{x}^i_j}{\bar{x}^i_{\theta,j} \bar{x}^i_{\ell_i}} \right] \mathrm{d}t,$$

where $\boldsymbol{x}$ comes from p_{data} or $\hat{p}_N$.Hence, the continuous-time UDM NELBO is likewise well defined for each individual clean sequence, irrespective of the support of the data distribution.

Validity of the leave-one-out optimum under finite data. We next verify that the leave-one-out characterization of the UDM NELBO optimum remains valid under Assumption 1. We refer the readers to Appendix A.1 of Gourevitch et al. (2026) for a full understanding. The proof of their optimum results applies directly when p_{data} is replaced by $\hat{p}_N$. Their argument relies only on Bayes' rule, the Chapman–Kolmogorov identity, factorization of the bridge across coordinates, and the Bayes-compatible simplex extension of the plug-in bridge. None of these steps imposes any assumption on the cardinality or support structure of the clean-data distribution.

Validity of the theoretical results in Section 3. Having established that the UDM NELBO and its leave-one-out optimum remain valid under Assumption 1, the remaining results in Section 3 follow without any additional assumption on the data distribution. Proposition 1 is obtained from the



(a) DUALITY (b) LOO+CE

Figure 3: Sampling performance of time-variant and time-agnostic UDMs across different numbers of function evaluations (NFEs). We compare generative perplexity as the sampling budget varies under **(a)** DUALITY and **(b)** LOO+CE; the annotated values report the entropy of generated samples.

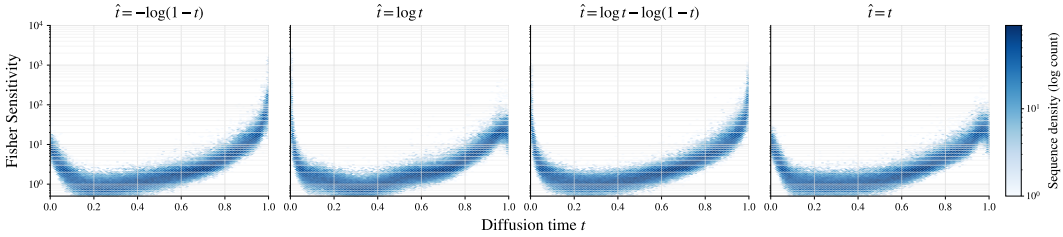


Figure 4: Fisher sensitivity under different time-input reparameterizations, where $\boldsymbol{z}$ sampling time and Fisher sensitivity evaluation time are equal. All sensitivities are measured with respect to the original diffusion time t, not reparametrized time $\hat{t}$ for model input.

leave-one-out posterior using only Bayes' rule, factorization of the uniform forward channel across coordinates, and a finite subset expansion. These arguments apply identically after replacing p_{data} by $\hat{p}_N$.

Likewise, Theorem 1 only groups the probability mass of the data distribution according to the matched-context count M^i and uses the fact that sequences with the same match count have the same uniform-channel likelihood. Proposition 2 then follows by differentiating the resulting finite normalized mixture weights. None of these derivations requires any assumption on the cardinality or support structure of p_{data}. Therefore, Proposition 1, Theorem 1, and Proposition 2 all remain valid under the finite empirical distribution in Assumption 1.

C EXPERIMENTAL DETAILS

C.1 ADDITIONAL EXPERIMENTAL RESULTS

We provide generative perplexity results for DUALITY and LOO+CE trained on OWT, which were omitted from the main paper. For each NFE, we generate 256 samples. Following Zheng et al. (2025) and DUALITY (Sahoo et al., 2025), we use float64 precision for sampling, as low-precision sampling can lead to reduced diversity and misleading generative perplexity in discrete diffusion models. As shown in Figure 3 (*Left*), time-agnostic and time-variant DUALITY achieve nearly identical generative perplexity across a wide range of NFEs. Their sample entropies are also closely matched, indicating similar sampling quality and diversity. On the other hand, as shown in Figure 3 (*Right*), LOO+CE exhibits similar sample entropy for the time-agnostic and time-variant models, while the time-agnostic model achieves lower generative perplexity.

C.2 Experimental Settings

Settings on training experiments. We use exactly the same experimental settings as the original UDM (Schiff et al., 2025) and Diffusion Duality (Sahoo et al., 2025). We detokenize the One Billion Words dataset following Lou et al. (2024); Sahoo et al. (2024), whose official code can be found. We tokenize LM1B using the BERT-BASE-UNCASED tokenizer, consistent with He et al. (2023). We then concatenate and pack the sequences to a fixed length of 128 (Raffel et al., 2020). For OWT, we use the GPT-2 tokenizer and similarly concatenate and pack sequences to length 1,024; during packing, we insert an `eos` token between consecutive documents. As OWT does not provide an official validation split, we reserve the last 100k documents for validation. We parameterize the diffusion backbone of all models with the modified diffusion transformer architecture (Peebles & Xie, 2023) from Lou et al. (2024); Sahoo et al. (2024). For time-variant models, we use 12 layers, a hidden dimension of 768, 12 attention heads. For the diffusion backbone, we use the AdamW optimizer with a batch size of 512, constant learning rate warmup from 0 to a learning rate of 3e-4 for 2,500 steps, following prior works (Sahoo et al., 2024; Lou et al., 2024; Sahoo et al., 2025). We use a dropout rate of 0.1. See Appendix C.3 for time-agnostic and hybrid models.

Experimental setting for the Hamming-distance and Δ analyses (Figure 1) The minimum hamming distance histogram is obtained by exhaustively comparing all $8{,}730{,}966$ OWT training rows of length $L = 1024$. For each row, we compute its exact minimum Hamming distance as L minus the maximum position-wise match count over all other rows, excluding its own row index. Special tokens are treated like ordinary tokens throughout.

For each $t \in \{0.05, 0.10, \ldots, 0.95, 0.99\}$, we independently sample $2{,}000$ source rows uniformly with replacement and corrupt each token by copying it with probability $1-t$ or drawing uniformly from the full $50{,}258$-token vocabulary, including the added padding token, with probability t. For each pair, we independently sample $i \sim \mathrm{Unif}([L])$ and compute $\Delta_i = M^i(\boldsymbol{x}, \boldsymbol{z}_t) - \max_{\boldsymbol{x}' : \boldsymbol{x}'^i \neq \boldsymbol{x}^i} M^i(\boldsymbol{x}', \boldsymbol{z}_t)$ by scanning the entire training set, where M^i counts matches only outside i. The probability mass at each time is estimated from the empirical frequencies of these $2{,}000$ margins.

Both measurements use exact exhaustive GPU scans, without approximate nearest-neighbor search. We store the entire token matrix losslessly in 16-bit integers (approximately 17.9 GB) on each H100 GPU and compute integer match counts across training rows.

Experimental setting for the learned time-sensitivity analysis (Figure 2 (*left*)). We use the public checkpoint both for DUALITY and LOO+CE trained on OWT. We uniformly sample 1000 held-out OWT sequences of length $L = 1024$ without replacement. For each $t_{\text{corruption}} \in \{0.01, 0.50, 0.99\}$, we independently draw one corrupted sequence per clean sequence from $q_{t_{\text{corruption}}}$ and reuse identical inputs for both models. Each corrupted input $\boldsymbol{z}$ is then held fixed while the evaluation time varies over $t \in \{0.01, 0.02, \ldots, 0.99\}$. We apply Eq. (8) to the raw network softmax prediction $\boldsymbol{x}_\theta^i(\boldsymbol{z}, t)$ in place of the oracle, before any analytic local-token correction. Derivatives are computed using automatic differentiation with respect to raw t. Each curve reports the arithmetic mean over all valid token positions, including unchanged tokens, followed by the mean over the 1000 sequences. In particular, for LOO+CE, we measure Fisher sensitivity with respect to $\boldsymbol{x}_\theta$, the LOO denoiser actually used for generation and in the plug-in reverse kernel, rather than $\boldsymbol{\mu}_\theta^{\text{Posterior}}$, which is introduced through the LOO reparameterization in Eq. (11).

Perplexity measure. In all experiments, we estimate validation/test perplexity by exponentiating the UDM NELBO. That is, we report perplexity that corresponds to an upper bound on the true perplexity, following the standard convention in most discrete diffusion language modeling works (Sahoo et al., 2024; 2025; Schiff et al., 2025; Gourevitch et al., 2026; Shi et al., 2024). Likewise, the time-resolved NLL plots report the expected token-normalized NELBO integrand at time t.

C.3 Details on Time-agnostic Models

Time-agnostic UDMs. Across all frameworks—including the conventional UDM (Schiff et al., 2025), Diffusion Duality (Sahoo et al., 2025), and the leave-one-out denoiser trained with cross-entropy (Gourevitch et al., 2026)—the only modification we make is to render the model time-agnostic. All other framework-specific components, including the loss function, forward process, noise schedule, and curriculum learning procedure, are kept exactly as originally proposed. More specifically, we directly build upon the original codebases and leave their framework-specific implementations

unchanged, except for removing time conditioning from the model. We refer the reader to Appendix A.2 for a detailed description of these frameworks. For experiments on LM1B, we compared 1) time-variant model with 12 Transformer layers, 2) time-agnostic model with 12 Transformer layers, 3) time-agnostic model with 13 Transformer layers that matches parameter count of 1), and 4) hybrid model. For experiments on OWT, we compared the time-variant model with 12 Transformer layers and the time-agnostic model with 13 Transformer layers that matches the parameter count of the time-variant counterpart. All architectures are explained in detail below.

Time-agnostic architecture. We use the same Transformer backbone for the time-conditioned and time-agnostic models, differing only in how the normalization and residual modulation parameters are specified. For the time-conditioned model, we follow the backbone of Sahoo et al. (2025); Schiff et al. (2025), where each Transformer block uses the adaLN-Zero parameterization: a timestep embedding produces time-dependent scale, shift, and residual-gating parameters. For time-agnostic models, we simply replaced time-dependent parameters with single learnable parameters.

More precisely, let $\mathbf{c}(t) \in \mathbb{R}^{d_c}$ denote the timestep embedding, where d_c is the conditioning dimension, and let d denote the Transformer hidden dimension. For the ℓ-th Transformer block, the modulation parameters are generated as

$$\big(\boldsymbol{\beta}_{\ell,\mathrm{attn}}(t), \mathbf{s}_{\ell,\mathrm{attn}}(t), \mathbf{g}_{\ell,\mathrm{attn}}(t), \boldsymbol{\beta}_{\ell,\mathrm{mlp}}(t), \mathbf{s}_{\ell,\mathrm{mlp}}(t), \mathbf{g}_{\ell,\mathrm{mlp}}(t)\big) = \mathbf{W}_\ell \mathbf{c}(t) + \mathbf{b}_\ell,$$

where $\mathbf{W}_\ell \in \mathbb{R}^{6d \times d_c}$, $\mathbf{b}_\ell \in \mathbb{R}^{6d}$, and each modulation vector lies in $\mathbb{R}^d$. Here, $\boldsymbol{\beta}$ and $\mathbf{s}$ denote the channel-wise shift and scale modulation of LayerNorm, respectively, while $\mathbf{g}$ denotes the channel-wise residual gate.

Let $\mathbf{h}_\ell \in \mathbb{R}^{L \times d}$ denote the sequence of hidden representations entering the ℓ-th block. Ignoring dropout for notational simplicity, the block is

$$\widetilde{\mathbf{h}}_\ell = \mathbf{h}_\ell + \mathbf{g}_{\ell,\mathrm{attn}}(t) \odot \mathrm{Attn}_\ell\left(\big(\mathbf{1}_d + \mathbf{s}_{\ell,\mathrm{attn}}(t)\big) \odot \mathrm{LN}_{\mathbf{w}_{\ell,\mathrm{attn}}}(\mathbf{h}_\ell) + \boldsymbol{\beta}_{\ell,\mathrm{attn}}(t)\right),$$
$$\mathbf{h}_{\ell+1} = \widetilde{\mathbf{h}}_\ell + \mathbf{g}_{\ell,\mathrm{mlp}}(t) \odot \mathrm{MLP}_\ell\left(\big(\mathbf{1}_d + \mathbf{s}_{\ell,\mathrm{mlp}}(t)\big) \odot \mathrm{LN}_{\mathbf{w}_{\ell,\mathrm{mlp}}}(\widetilde{\mathbf{h}}_\ell) + \boldsymbol{\beta}_{\ell,\mathrm{mlp}}(t)\right).$$

Here, $\mathbf{w}_{\ell,\mathrm{attn}}, \mathbf{w}_{\ell,\mathrm{mlp}} \in \mathbb{R}^d$ are the learned channel-wise LayerNorm scales. All element-wise operations above act along the hidden dimension, with the d-dimensional modulation vectors broadcast across sequence positions. Thus, time enters the Transformer backbone only through the scale, shift, and residual-gating parameters of each block.

For the time-agnostic model, we remove the timestep embedding $\mathbf{c}(t)$ entirely and directly learn a time-independent set of modulation parameters for each block:

$$\big(\boldsymbol{\beta}_{\ell,\mathrm{attn}}, \mathbf{s}_{\ell,\mathrm{attn}}, \mathbf{g}_{\ell,\mathrm{attn}}, \boldsymbol{\beta}_{\ell,\mathrm{mlp}}, \mathbf{s}_{\ell,\mathrm{mlp}}, \mathbf{g}_{\ell,\mathrm{mlp}}\big) \in \mathbb{R}^{6d},$$

where each of the six vectors belongs to $\mathbb{R}^d$.The resulting static normalization modulation is equivalent to a standard affine LayerNorm. More precisely, for a token representation $\mathbf{h} \in \mathbb{R}^d$, define

$$\mathrm{LN}_0(\mathbf{h}) = \frac{\mathbf{h} - \mu(\mathbf{h})\mathbf{1}_d}{\sqrt{\mathrm{Var}(\mathbf{h}) + \epsilon}}, \quad \mathrm{LN}_{\mathbf{w}}(\mathbf{h}) = \mathbf{w} \odot \mathrm{LN}_0(\mathbf{h}).$$

Then, for time-independent $\mathbf{s}, \boldsymbol{\beta} \in \mathbb{R}^d$,

$$(\mathbf{1}_d + \mathbf{s}) \odot \mathrm{LN}_{\mathbf{w}}(\mathbf{h}) + \boldsymbol{\beta} = \big(\mathbf{w} \odot (\mathbf{1}_d + \mathbf{s})\big) \odot \mathrm{LN}_0(\mathbf{h}) + \boldsymbol{\beta} = \boldsymbol{\gamma} \odot \mathrm{LN}_0(\mathbf{h}) + \boldsymbol{\beta},$$

where $\boldsymbol{\gamma} = \mathbf{w} \odot (\mathbf{1}_d + \mathbf{s}) \in \mathbb{R}^d$. The last expression is precisely standard affine LayerNorm with learned channel-wise scale $\boldsymbol{\gamma}$ and shift $\boldsymbol{\beta}$. Thus, the time-agnostic block is equivalent to

$$\widetilde{\mathbf{h}}_\ell = \mathbf{h}_\ell + \mathbf{g}_{\ell,\mathrm{attn}} \odot \mathrm{Attn}_\ell\left(\mathrm{LN}_\ell^{\mathrm{attn}}(\mathbf{h}_\ell)\right),$$
$$\mathbf{h}_{\ell+1} = \widetilde{\mathbf{h}}_\ell + \mathbf{g}_{\ell,\mathrm{mlp}} \odot \mathrm{MLP}_\ell\left(\mathrm{LN}_\ell^{\mathrm{mlp}}(\widetilde{\mathbf{h}}_\ell)\right),$$

where $\mathrm{LN}_\ell^{\mathrm{attn}}$ and $\mathrm{LN}_\ell^{\mathrm{mlp}}$ are standard learned affine LayerNorms with channel-wise scale and shift in $\mathbb{R}^d$, and $\mathbf{g}_{\ell,\mathrm{attn}}, \mathbf{g}_{\ell,\mathrm{mlp}} \in \mathbb{R}^d$ are learned, time-independent residual gates initialized to zero.

We apply the same replacement to the time-conditioned modulation in the final output layer.

Hybrid architecture. The hybrid model combines the time-conditioned and time-agnostic parameterizations above, using explicit time conditioning only in the high-noise regime. Specifically, for each

Transformer block, we retain both the original time-modulation layer and a separate set of directly learned, time-independent modulation parameters. The six modulation vectors are given by

$$\left(\boldsymbol{\beta}_{\ell,r}, \mathbf{s}_{\ell,r}, \mathbf{g}_{\ell,r}\right)_{r\in\{\text{attn,mlp}\}} = \begin{cases} \mathbf{W}_\ell \mathbf{c}(t) + \mathbf{b}_\ell, & t > 0.8, \\ \left(\boldsymbol{\beta}^0_{\ell,r}, \mathbf{s}^0_{\ell,r}, \mathbf{g}^0_{\ell,r}\right)_{r\in\{\text{attn,mlp}\}}, & t \le 0.8. \end{cases}$$

where the six vectors in the second branch are learned directly and do not depend on time. Thus, the model is identical to the time-conditioned architecture for $t > 0.8$ and to the time-agnostic architecture for $t \le 0.8$, with all other components left unchanged.