LANGUAGE DIFFUSION MODELS ARE ASSOCIATIVE MEMORIES

Bao Pham
RPI
phamb@rpi.edu

Mohammed J. Zaki
RPI
zaki@cs.rpi.edu

Luca Ambrogioni
Donder's institute of Cognition
luca.ambrogioni@donders.ru.nl

Dmitry Krotov
Independent Researcher
krotov.a.dmitry@gmail.com

Matteo Negri
CY Cergy Paris Université
matteo.negri1@cyu.fr

ABSTRACT

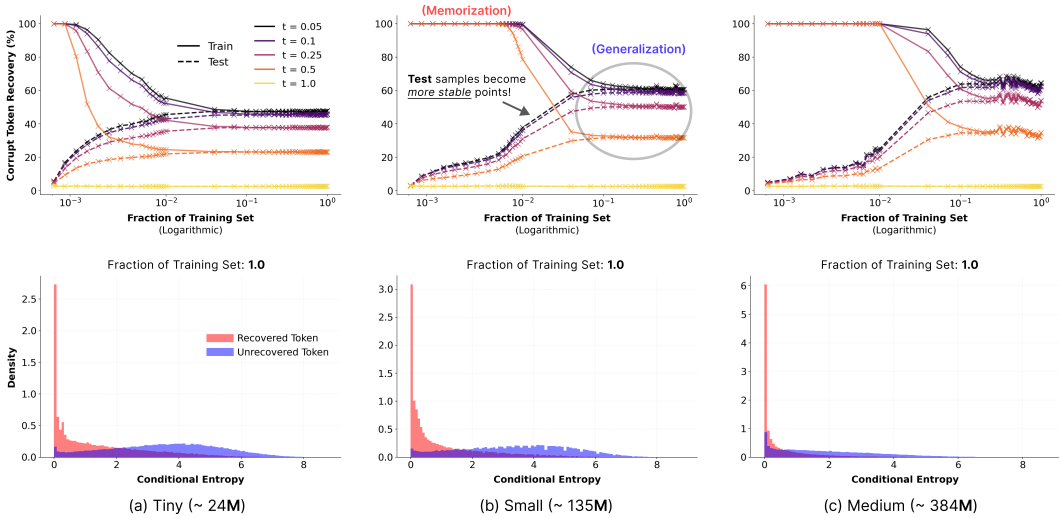
Associative Memory (AM) systems reliably retrieve data points by establishing distinct basins of attraction around them. While historically reliant on explicit and well-defined energy functions, as in Hopfield networks, stable attractors can also be formed via conditional likelihood maximization without the need for such functions. Using this aspect, we demonstrate that Uniform-based Discrete Diffusion Models (UDDMs) behave similarly to AMs via their utilization of conditional likelihood dynamics for sampling and training. By evaluating token recovery, we identify a memorization-to-generalization phase transition governed by training dataset size. With a small amount of training data, UDDMs exhibit a near-perfect memorization, characterized by vanishing conditional entropy. However, as the size of the training set increases, unseen test examples become stable attractors of the system and can be effectively denoised. This behavior highlights an emergent capability, marking the shift to generalization.

1 INTRODUCTION

To function as an Associative Memory (AM), a system must reliably retrieve stored data points by *establishing distinct basins of attraction* around them (Gardner, 1988). Historically, this has been accomplished through the usage of explicit and well-defined energy functions, as seen in the Hopfield networks (Amari, 1972; Hopfield, 1982). Recently, modern versions of these models, Dense Associative Memories (DenseAMs) (Krotov & Hopfield, 2016; 2018; Krotov, 2023), have sparked a scientific resurgence, bridging AM theory with Transformers (Ramsauer et al., 2021; Krotov & Hopfield, 2021; Hoover et al., 2023a; Dehmamy et al., 2025; Bacvanski et al., 2025), and even continuous diffusion models (DMs) (Hoover et al., 2023b; Ambrogioni, 2024; Pham et al., 2025). But, this reliance on explicit and well-defined energy functions is not strictly necessary to guarantee attractor dynamics. As shown by D'Amico et al. (2025), Hopfield networks can maintain their AM characteristics and even retrieve unseen patterns by utilizing pseudo-likelihood (Besag, 1974) as their objective function.

In recent times, continuous DMs (Sohl-Dickstein et al., 2015) have set new standards for image and video generation (Ho et al., 2020; Song & Ermon, 2019; Song et al., 2021; Rombach et al., 2022). However, their mechanics in the discrete domain (particularly for language modeling) remain poorly investigated. This gap of understanding is critical because DMs often face growing scrutiny regarding their tendency to replicate training data (Somepalli et al., 2023a;b; Carlini et al., 2023; Webster, 2023), raising fundamental questions about the interplay between *memorization* and *generalization*. Although recent studies have investigated these phenomena, they predominantly focus on the continuous image domain (Yoon et al., 2023; Kadkhodaie et al., 2023; Biroli et al., 2024; Kamb & Ganguli, 2024; Achilli et al., 2024). This leaves a significant disconnect between the understanding of memorization and generalization, and the rapidly advancing field of text generation.

In this work, we bridge this gap by establishing that Uniform-based Discrete Diffusion Models (UDDMs) (Austin et al., 2021; Campbell et al., 2024; Gat et al., 2024; Sahoo et al., 2025) fundamentally function as AM systems via conditional likelihood dynamics. Inspired by Pham et al. (2025), which



(a) Tiny (~ 24M) (b) Small (~ 135M) (c) Medium (~ 384M)

Figure 1: UDDMs have memorization and generalization phases analogous to AMs. *Top row*: illustration of the average corrupt token recovery rate (%) as the fraction of the training dataset increases (with different model sizes), comparing the model's ability to recover tokens from the training set (**solid** lines) versus unseen samples from the test set (**dashed** lines) across different timesteps. This visualization captures the memorization to generalization transition, where high recovery on training data drops as the dataset size grows while the recovery rate for test samples increases. Crucially, the recovery rates for both training and test samples converge to the same rate as the training dataset size increases. *Bottom row*: histograms showing the density of *recovered* and *unrecovered* tokens and their respective token-level conditional entropy computed at $t = 0.25$ and using Eq. (12) and three different-size models trained with the full dataset. Here, successfully recovered tokens generally cluster around near-zero entropy, whereas unrecovered tokens are distributed across higher entropy values.

characterized the memorization-generalization transition of continuous DMs from the AM perspective, we also demonstrate that UDDMs undergo a phase transition governed by the training dataset size. Specifically, as the training dataset size increases, the model transitions from a memorization regime, characterized by vanishing conditional token entropy, to a generalization regime where a significant entropy gap emerges and drives the synthesis of novel and diverse text patterns. We also observe a fascinating trend: the token recovery rate for unseen test sequences increases (while that of training sequences decreases) alongside the increasing training dataset size before stabilizing to a particular rate. This convergence of token recovery rates signifies the generalization in UDDMs, where unseen test samples have become attractors to these systems.

2 Preliminary

Consider a clean token $\mathbf{x} \in \mathcal{V}$ drawn from the data distribution q_{data} with the vocabulary $\mathcal{V} = \{\mathbf{x} \in \{0,1\}^K : \sum_{i=1}^K \mathbf{x}_i = 1\}$. In the DDM framework, q_{data} is mapped into a simple distribution through a sequence of Markov states via a forward process that is somewhat akin to the continuous diffusion framework (Austin et al., 2021; Sahoo et al., 2024; 2025):

$$\mathbf{z}_t \sim q_t(\mathbf{z}_t|\mathbf{x}; \alpha_t) = \text{Cat}(\mathbf{z}_t; \alpha_t \mathbf{x} + (1 - \alpha_t)\boldsymbol{\pi}), \tag{1}$$

where $\boldsymbol{\pi} \in \Delta$, $\text{Cat}(\cdot)$ denotes categorical distribution, and Δ denotes K-simplex. Here, $\mathbf{z}_t$ denotes the perturbed token at a time $t \in (0, 1]$, where $\mathbf{z}_0 = \mathbf{x}$. The diffusion parameter $\alpha_t \in [0, 1]$ is a strictly decreasing t-dependent function with the boundary conditions: $\alpha_{t=0} \approx 1$ and $\alpha_{t=1} \approx 0$.

In UDDM, as shown by Austin et al. (2021) and Campbell et al. (2024), the true reverse posterior of a previous timestep $s < t$ corresponding to the forward process (1) is

$$\mathbf{z}_s \sim q_{s|t}(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) = \text{Cat}\left(\mathbf{z}_s; \frac{K\alpha_t \mathbf{z}_t \odot \mathbf{x} + (\alpha_{t|s} - \alpha_t)\mathbf{z}_t}{K\alpha_t \langle \mathbf{z}_t, \mathbf{x} \rangle + (1 - \alpha_t)} + \frac{(\alpha_s - \alpha_t)\mathbf{x} + (1 - \alpha_{t|s})\frac{\mathbf{1}}{K}}{K\alpha_t \langle \mathbf{z}_t, \mathbf{x} \rangle + (1 - \alpha_t)}\right), \tag{2}$$

which defines the approximate reverse posterior $p^\theta_{s|t}(\mathbf{z}_s|\mathbf{z}_t) = q_{s|t}(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x} = \mathbf{x}_\theta(\mathbf{z}_t, t))$, where the diffusion parameter is $\alpha_{t|s} = \frac{\alpha_t}{\alpha_s}$. Here, $\langle \cdot, \cdot \rangle$ denotes the dot product , $\odot$ denotes Hadamard product, and we have a uniform prior over $\mathcal{V}(\boldsymbol{\pi} = \mathbf{1}/K)$ (Sahoo et al., 2024; 2025). Following Eq. (2), we train a neural network $\mathbf{x}_\theta(\mathbf{z}_t, t) \approx \mathbf{x}$ to predict the clean token $\mathbf{x}$ at any time t and optimize it via

the Negative Evidence Lower Bound (NELBO) objective (Austin et al., 2021; Sahoo et al., 2024):

$$\mathbb{E}_q \Big[\underbrace{-\log p_\theta(\mathbf{x}|\mathbf{z}_0)}_{\mathcal{L}_{\text{reconstruction}}} + \underbrace{\sum_{s<t}^{T} D_{\text{KL}}\big[q(\mathbf{z}_s|\mathbf{z}_t,\mathbf{x}) \,||\, p_\theta(\mathbf{z}_s|\mathbf{z}_t)\big]}_{\mathcal{L}_{\text{diffusion}}} \Big] + \underbrace{D_{\text{KL}}\big[q(\mathbf{z}_T|\mathbf{x}) \,||\, p_\theta(\mathbf{z}_T)\big]}_{\mathcal{L}_{\text{prior}}}. \tag{3}$$

The form of UDDMs we used is based on Sahoo et al. (2025). Please see Appx. (C) for more details.

3 Associative memories from conditional sampling

To establish a connection between AMs and UDDMs, we rely on the *core assumption* (Austin et al., 2021; Hoogeboom et al., 2021; Lou et al., 2024; Sahoo et al., 2024): the denoising process of a sequence $\mathbf{z}^{1:L}$ of length L factorizes for each token $\mathbf{z}^\ell$ as

$$p^\theta_{s|t}(\mathbf{z}_s^{1:L} \mid \mathbf{z}_t^{1:L}) = \prod_{\ell=1}^{L} \psi^\theta_{s|t}(\mathbf{z}_s^\ell \mid \mathbf{z}_t^{1:L}), \tag{4}$$

where $\psi^\theta_{s|t}$ denotes the conditional probability

$$\psi^\theta_{s|t}(\mathbf{z}_s^\ell \mid \mathbf{z}_t^{1:L}) = \text{Cat}\big(\mathbf{z}_s^\ell;\, \text{softmax}_K\big[\beta(t)\, f^\ell_\theta(\mathbf{z}_t^{1:L})\big]\big), \tag{5}$$

$f^\ell_\theta(\cdot)$ are the logits produced from a Diffusion Transformer backbone (Peebles & Xie, 2023) and the softmax is applied over K categories, which produces a probability distribution per position ℓ. Here $\beta(t)$ is a time-dependent inverse temperature, dependent on the diffusion variable $\alpha(t)$, typically increasing as $t \to 0$. These conditional probabilities enter the cross-entropy terms of NELBO (3) and are also used in practice during the denoising process. This reliance on cross-entropy consequently provides a connection to AMs, since it produces *basins of attraction around the training data points* in the dynamics of conditional sampling (D'Amico et al., 2025).

To elaborate, consider a Hopfield network of L binary neurons $\mathbf{s}^\ell \in \{\pm 1\}$ with a non-symmetric coupling matrix $\mathbf{W} \in \mathbb{R}^{L\times L}$, where its *diagonal entries are zero*, we have the following deterministic update rule:

$$\mathbf{s}^\ell_\tau = \text{sgn}\Big(\sum_{m=1}^{L} \mathbf{W}^{\ell m}\, \mathbf{s}^m_{\tau+1} \Big), \tag{6}$$

where we assume the dynamics run backward similarly to UDDMs, while τ denotes a discrete time. It is useful to move from the deterministic update (6) to conditional sampling by reinterpreting it as selecting the most probable state via $\arg\max$ of the conditional distribution (between position ℓ and its *neighborhood* or $1 \dots L$ positions in the spin vector excluding ℓ [1]):

$$\psi_{\tau|\tau+1}(\mathbf{s}^\ell_\tau \mid \mathbf{s}^{1:L}_{\tau+1}; \mathbf{W}^\ell) = \frac{\exp\Big(\mathbf{s}^\ell_\tau f^\ell(\mathbf{s}^{1:L}_{\tau+1})\Big)}{2\cosh\Big(f^\ell(\mathbf{s}^{1:L}_{\tau+1})\Big)}, \tag{7}$$

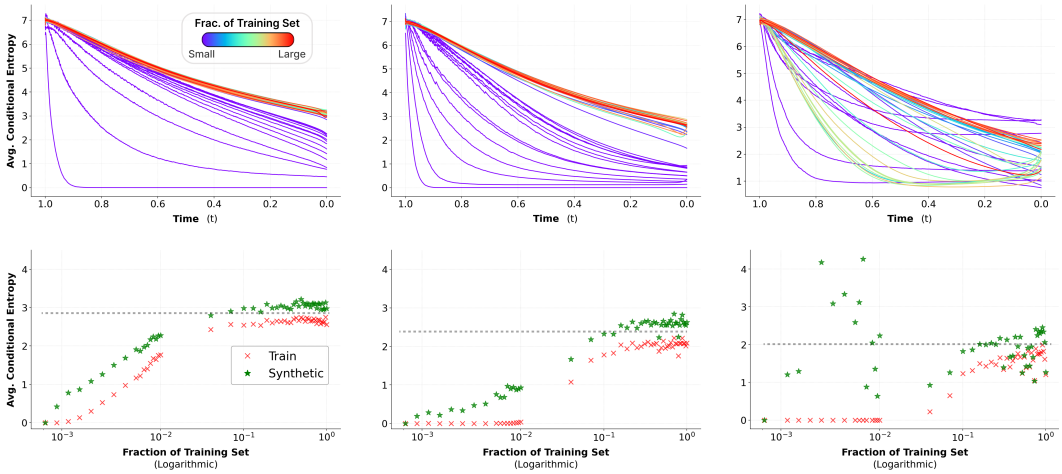
where $f^\ell(\mathbf{s}^{1:L}_{\tau+1}) = \beta \sum_{m=1}^{L} \mathbf{W}^{\ell m}\, \mathbf{s}^m_{\tau+1}$ with a fixed inverse temperature β [2]. For binary variables, Eq. (7) yields a logistic form. But, for generic categorical variables, $f_\ell(\mathbf{s})$ are logits inside $\text{softmax}(\cdot)$ like that of Eq. (5). Here, the couplings $\mathbf{W}$ induced by conditional likelihood do not need to be symmetric like classical AM (D'Amico et al., 2025), and therefore *no explicit global energy function is required*. The existence of attractor-like behavior follows directly from the structure of the conditional probabilities.

Specifically, given a set of P examples $\boldsymbol{\Xi} \in \{\pm 1\}^{P\times L}$ where $\mathbf{x} \in \boldsymbol{\Xi}$ is a sequence of L spins. We can train a model by minimizing the following objective based on pseudo-likelihood (Besag, 1974):

$$\mathcal{L}(\mathbf{W}) = -\frac{1}{P} \sum_{\mathbf{x}\,\in\,\boldsymbol{\Xi}} \log \prod_{\ell=1}^{L} \psi_\ell(\mathbf{x}^\ell|\mathbf{x}^{1:L}; \mathbf{W}^\ell) = -\frac{1}{P} \sum_{\mathbf{x}\,\in\,\boldsymbol{\Xi}} \sum_{\ell=1}^{L} \Big[\mathbf{x}^\ell f^\ell(\mathbf{x}^{1:L}) - \log 2\cosh(f^\ell(\mathbf{x}^{1:L})) \Big]. \tag{8}$$

[1] Since the diagonal entries of our coupling matrix $\mathbf{W}$ are zeroed, position ℓ *does not attend to itself*.

[2] Eq. (7) is obtained via the relationship $p(\mathbf{s}^\ell|\mathbf{s}) \propto \exp(-E(\mathbf{s}^\ell|\mathbf{s}))$, where the energy function is $E(\mathbf{s}^\ell|\mathbf{s}) = -\beta \mathbf{s}^\ell \sum_{\ell,m} \mathbf{W}^{\ell m}\mathbf{s}^m = -\beta \mathbf{s}^\ell f(\mathbf{s})$.



(a) Tiny (~ 24M) (b) Small (~ 135M) (c) Medium (~ 384M)

Figure 2: Memorization and generalization phases of UDDMs can be identified by conditional entropy. *Top row* illustrates the trajectory of the average conditional entropy throughout the reverse process for different model sizes (denoted by each column). The horizontal axis represents time (t), starting from the noisy state to the clean state at $t \approx 0$, while the vertical axis tracks the conditional entropy. The various curves are color-coded to represent models trained with various fractions of the training set, ranging from smaller fractions (in purple) to larger fractions (in red). A significant feature highlighted in the graph is that the entropy levels of models trained on different fractions of the training set diverge as $t \to 0$, indicating how the training dataset size influences the model's uncertainty. *Bottom row* depicts the average conditional entropy for both training and synthetic data as the fraction of the training dataset increases, highlighting the existence of an *entropy gap* between the training data and synthetic samples. As the training set size increases, the average conditional entropy stabilizes to a certain value. Crucially, as the model's size increases, that certain entropy value also decreases at high training set sizes. The initial noisy entropy trend of the medium-sized model is also observed in the perplexity scores Fig. (11) computed from `GPT-2 Large` (Radford et al., 2019) in Appx. (E).

If we derive the loss (8) with respect to the coupling matrix $\mathbf{W}$,

$$\frac{\mathrm{d}\,\mathcal{L}(\mathbf{W})}{\mathrm{d}\,\mathbf{W}^{\ell m}} \propto -\frac{1}{P}\sum_{\mathbf{x}\,\in\,\boldsymbol{\Xi}}\Big[\underbrace{\mathbf{x}^{\ell}\mathbf{x}^{m}}_{\text{Hebbian}} - \underbrace{\tanh(f^{\ell}(\mathbf{x}^{1:L}))\mathbf{x}^{m}}_{\text{Margin}}\Big], \tag{9}$$

we can observe two gradient terms, involving the typical Hebbian rule used in (Hopfield, 1982) and another which modifies the classification margin $M(\mathbf{x}^{\ell}) = \mathbf{x}^{\ell} f^{\ell}(\mathbf{x}^{1:L})$ around those points. Furthermore, if we factor out the Hebbian term in Eq. (9), we can see a penalty term which penalizes points with a small margin:

$$\Delta\mathbf{W}^{\ell m} \propto \underbrace{\mathbf{x}^{\ell}\mathbf{x}^{m}}_{\text{Hebbian}}\Big[\underbrace{1-\tanh(M(\mathbf{x}^{\ell}))}_{\approx 2e^{-2M(\mathbf{x}^{\ell})}}\Big]. \tag{10}$$

With just Hebbian learning, it is not sufficient to find an optimal coupling matrix $\mathbf{W}^{*}$ in Eq. (6) to build an AM system – where $\forall \mathbf{x} \in \boldsymbol{\Xi}$ is a fixed point to the system – as it does not imply that there are finite basins of attraction around these examples. To create such basins, a stronger condition is needed to be enforced for the deterministic update (6). Also, as shown by Soudry et al. (2018) and Montanari et al. (2024), training a Perceptron with the cross-entropy loss in the separable regime implicitly solves Eq. (6) with a classification margin, capable of creating large basins of attraction around the training examples $\mathbf{x}^{\mu} \in \boldsymbol{\Xi}$.

Overall, the intuition is that the cross-entropy terms in the NELBO (3), used to train UDDMs, *enforce correct classification with margin*. Albeit, the pseudo-likelihood objective (8) is more related to the term $\mathcal{L}_{\text{reconstruction}}$ in Eq. (3), since AM systems rely on a fixed temperature value rather than annealing the temperature during its dynamics. Thus, the reverse diffusion process can be interpreted as a stochastic AM retrieval dynamics for categorical variables, similarly in the continuous setting (Ambrogioni, 2024; Pham et al., 2025). Please see Appx. (A) for more mathematical details.

4 MEMORIZATION TO GENERALIZATION

To test aspects of AM within UDDMs, we quantitatively analyze their transition from memorization to generalization on the LM1B dataset (Chelba et al., 2013), via two key metrics as a function of

the training data size: *token recovery rate* and *conditional entropy*. These metrics serve as proxies for the stability of the attractors and the geometry of the energy landscape, respectively. For more details and results, please refer to Appx. (B).

Corrupt Token Recovery. We define the corrupt token recovery rate as the accuracy with which the model recovers the corrupted tokens of a target sequence of length L after applying the reverse process on a noisy sequence defined at a time $t \in (0, 1]$:

$$R(\mathbf{x}^{1:L}, \hat{\mathbf{x}}^{1:L}) = \frac{1}{|\mathcal{M}|} \sum_{j \in \mathcal{M}} \delta(\mathbf{x}^j, \hat{\mathbf{x}}^j), \tag{11}$$

where $\mathbf{x}^{1:L}$ is the original sequence, $\hat{\mathbf{x}}^{1:L}$ is the recovered sequence after running the denoising process, $\mathcal{M} = \{j : \delta(\mathbf{x}^j, \mathbf{z}_t^j) = 0\}$ is the set of indices in the target sequence denoting positions where tokens have been changed after applying the forward process (1), and $\delta(\cdot, \cdot)$ denotes the Kronecker delta function. This metric (11) focuses the correlation between the model's attractors and training and test examples. High recovery on *just* training examples indicates memorization, while a finite recovery rate on *both* training and unseen test examples suggests generalization. The results are shown in Fig. (1).

Here, we observe that as the training data size increases, the model's ability to recover corrupted training tokens diminishes, and stabilizes at a specific rate for different levels of corruption based on t. For example, at $t = 1$, the recovery rate approaches zero, while at $t = 0.25$, the model can recover roughly 50% of the corrupted tokens. Interestingly, for a larger model size, we observe that the drop in recovery rate occurs at a later stage (i.e., at a larger training data size), indicating a delay in the model's memorization to generalization transition. Refer to Appx. (E) for text recovery examples.

Conditional Entropy. While the corrupt recovery rate indicates whether the system returns to an attractor, it does not provide any information about the basins of attraction. Unlike continuous DMs, UDDMs provide direct access to the conditional likelihood, allowing us to probe the geometry of the energy landscape as the training data size grows. Specifically, the conditional entropy of a token $\mathbf{x}^\ell$ in a sequence given its associated perturbation $\mathbf{z}_t^\ell$ at some time t is defined as:

$$\mathcal{H}(\mathbf{x}^\ell \mid \mathbf{z}_t^\ell) = -\sum_{k=1}^{K} \left[p_\theta(\mathbf{x}^\ell \mid \mathbf{z}_t^\ell)\right]_k \cdot \log \left[p_\theta(\mathbf{x}^\ell \mid \mathbf{z}_t^\ell)\right]_k = -\sum_{k=1}^{K} \left[\mathbf{x}_\theta^\ell(\mathbf{z}_t^\ell, t)\right]_k \cdot \log \left[\mathbf{x}_\theta^\ell(\mathbf{z}_t^\ell, t)\right]_k. \tag{12}$$

In AM, the sharpness of the basin dictates retrieval dynamics (Krotov & Hopfield, 2016; Krotov, 2023; Krotov et al., 2025). Here, low entropy implies deterministic transitions into attractors (*memorization*), while high entropy signals a flatter landscape with distributed probability mass (Biroli et al., 2024; Pham et al., 2025). As we observe in Fig. (2), the reverse process of UDDMs intrinsically decreases the conditional entropy as $t \to 0$. Meanwhile, we also observe that successful token recovery is characterized by near-zero conditional entropy in the histograms of Fig. (1). In contrast, increasing the training dataset size prevents this entropy collapse and introduces an *entropy gap* between the model's generated and training samples shown in Fig. (2). Notably, a surprising fraction of low-entropy tokens persists even in the generalization phase, see bottom row of Fig. (1). Please see Appx. (E) for additional results and (D) for our discussion on conditional entropy and energy curvature.

5 Conclusion

From the AM perspective, UDDMs function by attempting to recall *stored* patterns given corrupted inputs through conditional sampling, where the failures of token recovery lead to the creation of novel patterns. Specifically, this framework reveals that a UDDM's generative behavior is best characterized by its token recovery rate on perturbed training and unseen test examples, allowing us to not rely on typical metrics that focus on sample quality, like those in continuous DMs. Crucially, the memorization to generalization transition remains detectable via the conditional entropy of the token probabilities and token recovery rate, even without access to reference samples, as low entropy signals successful retrieval of a token while high entropy indicates generalization (the failure of perturbed token recovery). This relationship highlights a possibility of diminishing returns in training beyond the generalization transition and reveals that even in the generative regime, token-level entropy remains diverse.

ACKNOWLEDGMENT

The work of Bao Pham and Mohammed Zaki was funded by the RPI-IBM Future of Computing Research Collaboration program. Part of the work of Dmitry Krotov was performed while he was employed by IBM.

REFERENCES

Beatrice Achilli, Enrico Ventura, Gianluigi Silvestri, Bao Pham, Gabriel Raya, Dmitry Krotov, Carlo Lucibello, and Luca Ambrogioni. Losing dimensions: Geometric memorization in generative diffusion. *arXiv preprint arXiv:2410.08727*, 2024.

S-I Amari. Learning patterns and pattern sequences by self-organizing nets of threshold elements. *IEEE Transactions on computers*, 100(11):1197–1206, 1972.

Luca Ambrogioni. In search of dispersed memories: Generative diffusion models are associative memory networks. *Entropy*, 26(5):381, 2024.

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

Marc Gong Bacvanski, Xincheng You, John Hopfield, and Dmitry Krotov. Dense associative memories with analog circuits. *arXiv preprint arXiv:2512.15002*, 2025.

Marco Benedetti, Enrico Ventura, Enzo Marinari, Giancarlo Ruocco, and Francesco Zamponi. Supervised perceptron learning vs unsupervised hebbian unlearning: Approaching optimal memory retrieval in hopfield-like networks. *The Journal of Chemical Physics*, 156(10), 2022.

Julian Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):192–236, 1974. ISSN 00359246. URL http://www.jstor.org/stable/2984812.

Giulio Biroli, Tony Bonnaire, Valentin de Bortoli, and Marc Mézard. Dynamical regimes of diffusion models. *Nature Communications*, 15(1), November 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-54281-3.

Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. *arXiv preprint arXiv:2402.04997*, 2024.

Nicholas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramèr, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *Proceedings of the 32nd USENIX Conference on Security Symposium*, SEC '23, USA, 2023. USENIX Association. ISBN 978-1-939133-37-3.

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. *arXiv preprint arXiv:1312.3005*, 2013.

Francesco D'Amico, Dario Bocchi, Luca Maria Del Bono, Saverio Rossi, and Matteo Negri. Pseudo-likelihood produces associative memories able to generalize, even for asymmetric couplings. *arXiv preprint arXiv:2507.05147*, 2025.

Nima Dehmamy, Benjamin Hoover, Bishwajit Saha, Leo Kozachkov, Jean-Jacques Slotine, and Dmitry Krotov. Nrgpt: An energy-based alternative for gpt. *arXiv preprint arXiv:2512.16762*, 2025.

BM Forrest. Content-addressability and learning in neural networks. *Journal of Physics A: Mathematical and General*, 21(1):245, 1988.

Elizabeth Gardner. The space of interactions in neural network models. *Journal of physics A: Mathematical and general*, 21(1):257, 1988.

Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky TQ Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. *Advances in Neural Information Processing Systems*, 37: 133345–133385, 2024.

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

Emiel Hoogeboom, Didrik Nielsen, Amir Abdolshahi, and Arash Vahdat. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in Neural Information Processing Systems*, 34, 2021. URL https://arxiv.org/abs/2102.05379.

Benjamin Hoover, Yuchen Liang, Bao Pham, Rameswar Panda, Hendrik Strobelt, Duen Horng Chau, Mohammed Zaki, and Dmitry Krotov. Energy transformer. In A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 27532–27559. Curran Associates, Inc., 2023a.

Benjamin Hoover, Hendrik Strobelt, Dmitry Krotov, Judy Hoffman, Zsolt Kira, and Duen Horng Chau. Memory in plain sight: A survey of the uncanny resemblances between diffusion models and associative memories. *arXiv preprint arXiv:2309.16750*, 2023b.

John J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982. doi: 10.1073/pnas.79.8.2554. URL https://www.pnas.org/doi/abs/10.1073/pnas.79.8.2554.

Zahra Kadkhodaie, Florentin Guth, Eero P Simoncelli, and Stéphane Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representation. *arXiv preprint arXiv:2310.02557*, 2023.

Mason Kamb and Surya Ganguli. An analytic theory of creativity in convolutional diffusion models. *arXiv preprint arXiv:2412.20292*, 2024.

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

Dmitry Krotov. A new frontier for Hopfield networks. *Nature Reviews Physics*, 5(7):366–367, July 2023. doi: 10.1038/s42254-023-00595-y.

Dmitry Krotov and John Hopfield. Dense associative memory is robust to adversarial inputs. *Neural computation*, 30(12):3151–3167, 2018.

Dmitry Krotov and John J. Hopfield. Dense associative memory for pattern recognition. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/eaae339c4d89fc102edd9dbdb6a28915-Paper.pdf.

Dmitry Krotov and John J Hopfield. Large associative memory problem in neurobiology and machine learning. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=X4y_10OX-hX.

Dmitry Krotov, Benjamin Hoover, Parikshit Ram, and Bao Pham. Modern methods in associative memory. *arXiv preprint arXiv:2507.06211*, 2025.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Masked diffusion models are masked language models. *arXiv preprint arXiv:2406.07524*, 2024. URL https://arxiv.org/abs/2406.07524.

Andrea Montanari, Yiqiao Zhong, and Kangjie Zhou. Tractability from overparametrization: The example of the negative perceptron. *Probability Theory and Related Fields*, 188(3–4): 805–910, 2024. doi: 10.1007/s00440-023-01248-y. URL https://doi.org/10.1007/s00440-023-01248-y. arXiv:2110.15824.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Bao Pham, Gabriel Raya, Matteo Negri, Mohammed J Zaki, Luca Ambrogioni, and Dmitry Krotov. Memorization to generalization: Emergence of diffusion models from associative memory. *arXiv preprint arXiv:2505.21777*, 2025.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=tL89RnzIiCd.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

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

Subham Sekhar Sahoo, Justin Deschenaux, Aaron Gokaslan, Guanghan Wang, Justin T Chiu, and Volodymyr Kuleshov. The diffusion duality. *International Conference on Machine Learning*, 42, 2025. URL https://openreview.net/forum?id=9P9Y8FOSOk.

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

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6048–6058, 2023a.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Understanding and mitigating copying in diffusion models. *Advances in Neural Information Processing Systems*, 36:47783–47803, 2023b.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

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

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19(70): 1–57, 2018.

Ryan Webster. A reproducible extraction of training images from diffusion models. *arXiv preprint arXiv:2305.08694*, 2023.

TaeHo Yoon, Joo Young Choi, Sehyun Kwon, and Ernest K Ryu. Diffusion probabilistic models generalize when they fail to memorize. In *ICML 2023 Workshop on Structured Probabilistic Inference Generative Modeling*, 2023.

APPENDIX

A Additional Details on Pseudo-Likelihood and Classification Margin

To begin, it is important to note that the dynamics in AMs are entirely deterministic in contrast to the stochastic dynamics in UDDMs. Using the conditional probability (7) defined in Sec. (3), we want to sample the next probable state following the deterministic update (6):

$$\mathbf{s}_\tau^\ell = \arg\max_{\mathbf{s}^\ell} \; \psi_{\tau|\tau+1}\big(\mathbf{s}_\tau^\ell \mid \mathbf{s}_{\tau+1}^{1:L}; \mathbf{W}_\ell\big). \tag{13}$$

Although the dynamics between these two systems are different, the fact that UDDMs rely on cross-entropy in NELBO (3) to learn their neural network – highlights the existence of classification margin which enables basins of attraction around training data points, and even unseen test samples, as shown by our token recovery rate results in Fig. (1).

To further elaborate on the changing margin during the gradient descent of Eq. (9), we can define the local classification margin of $\mathbf{x}^\ell$ as $M(\mathbf{x}^\ell) = \mathbf{x}^\ell f^\ell(\mathbf{x}^{1:L})$, where $\mathbf{x}^\ell \in \{\pm 1\}$ is a spin and

$$f^\ell(\mathbf{x}^{1:L}) = \beta \sum_{m=1}^{L} \mathbf{W}^{\ell m}\, \mathbf{x}^m, \tag{14}$$

where we omit the time subscripts for conciseness. Using the relationship $\tanh(f(\mathbf{x}^\ell)) = \tanh(\mathbf{x}^\ell M(\mathbf{x}^\ell))$, we can factor out the Hebbian term in Eq. (9):

$$\Delta \mathbf{W}^{\ell m} \propto \underbrace{\mathbf{x}^\ell \mathbf{x}^m}_{\text{Hebbian}} \underbrace{\Big[1 - \tanh(M(\mathbf{x}^\ell))\Big]}_{\text{Penalty}} \tag{15}$$

where $\tanh(\mathbf{x}^\ell M(\mathbf{x}^\ell)) = \mathbf{x}^\ell \tanh(M(\mathbf{x}^\ell))$ since $\mathbf{x}^\ell$ is simply a sign flip. We can observe the gradient penalty $1 - \tanh(M(\mathbf{x}^\ell)) \approx 2e^{-2M(\mathbf{x}^\ell)}$ decays exponentially for correctly classified patterns with wide margins. Consequently, based on this fact, the pseudo-likelihood objective (8) suppresses updates for the stored data points and concentrates learning on patterns with the smallest margins as the weight magnitudes diverge in the separable regime (D'Amico et al., 2025), indicating that the learned model stores its data points with the possible maximum margin for each point. Hence, the observations made from Eq. (9) are very essential to linking AMs and UDDMs via conditional dynamics.

As mentioned prior, it is not sufficient to find an optimal coupling matrix $\mathbf{W}^*$ in Eq. (6) to build an AM system, because $\mathbf{W}^*$ does not imply that there are finite basins of attraction around these examples. To create such basins, a stronger condition is needed to be enforced for the deterministic update (6):

$$\mathbf{x}^\ell = \operatorname{sgn}\Big(\sum_{m=1}^{L} \mathbf{W}^{\ell m}\mathbf{x}^m + \kappa\Big), \qquad \forall \ell = 1, \dots, L \tag{16}$$

so that a classification margin $\kappa \in \mathbb{R}^+$ ensures that each fixed point is robust to a finite amount of variable flips from the deterministic update rule (Gardner, 1988; Forrest, 1988; Benedetti et al., 2022). Here, larger the κ implies larger basins. Given the load $\gamma = P/L$, there exists a maximum margin $\kappa_{\max}(\gamma)$. As shown by Soudry et al. (2018) and Montanari et al. (2024), training a Perceptron with the cross-entropy loss in the separable regime implicitly solves Eq. (16) with $\kappa = \kappa_{\max}(\gamma)$, creating large basins of attraction around the training examples $\mathbf{x}^\mu \in \boldsymbol{\Xi}$.

B Additional Details on Memorization to Generalization

Setup. For our experiments, showcased in Sec. (4), we trained two sets of UDDMs, labeled as *tiny* and *small*, utilizing the code base and approach of Sahoo et al. (2025). For more details on this variant of UDDMs, please refer to the discussion in Appx. (C) below. Meanwhile, the backbone of our trained UDDMs is the diffusion transformer architecture from Peebles & Xie (2023). The configurations of our *tiny*, *small*, and *medium* models are described in Tab. (1).

Meanwhile, there are a total of 180 models, or 50 models for each of the two sets (i.e., tiny and small sets), we have trained. All models are trained up to 1 million training iterations following Sahoo

Table 1: A table showing hyperparameters of the tiny and small UDDMs.

	Model Type		
Hyperparameters	Tiny	Small	Medium
Hidden Size	256	768	1024
Conditioning Size	128	128	128
Length	128	128	128
Num. of Blocks	8	12	24
Num. of Heads	8	12	16
Scale by Sigma[1]	True	True	True
Dropout	0.1	0.1	0.1

[1] Note: scale by sigma indicates that the model takes the inverse temperature or the appropriate diffusion scheduling parameter at time t instead of the typical approach of conditioning on t.

et al. (2025). For the selection of the fraction of the training dataset sizes, we initially start with the fraction $n = 0.01$ and increment it by $\Delta n = 0.03$ all the way to the full dataset. However, to further magnify the memorization phase, we train more points using linearly spacing (of 17 points, inclusively) starting at 10^{-4} to 10^{-2}. Lastly, our models are trained on the LM1B dataset (Chelba et al., 2013), where our model handles the block size (or sequence length) of 128, and all of them are initialized from the same random seed.

Corrupt Token Recovery. To obtain the results in the top row of Fig. (1), we utilized our trained models, from the tiny and small sets, compute the analysis of randomly chosen 5000 samples belonging to their respective training set (in accordance with their fraction of training dataset size) and the unseen test set. Here, we perform the perturbation using the forward process (20) at time t and run the reverse process (2) starting at that time t back to a small time $\epsilon = 10^{-5}$. Then, to measure our recovery rate, we applied Eq. (11) from the main text, which measures the rate of perturbed tokens being recovered.

Conditional Entropy. Similarly, in the bottom row of Fig. (1), we identified which tokens in a sequence is recovered and those have failed to be recovered, and then compute the conditional entropy of each class of tokens using Eq. (12) in the main text. For this result, we do not average the conditional entropy. However, in the case of results in top panel of Fig. (2), we measured the average conditional entropy of 5000 training samples (which is also averaged over the sequence length) of the reverse process starting from $t = 1$ to $\epsilon = 10^{-5}$. Similarly, perturbation is first applied and the reverse process is performed afterwards.

Finally, we are interested the effects of training examples and generated samples from our models: whether if there are differences in their conditional entropy. For the results in the bottom panel of Fig. (2), we generated a synthetic set of 100000 samples for each of the trained models and performed our analyses (also, with at most 100000 training sequences). See Figs. (6)-(8) for the full histograms of conditional entropy on training versus generated samples, and Figs. (3)-(5) for conditional entropy histograms of unrecovered versus recovered tokens shown in Fig. (1).

Hardware. The training of the UDDMs are done using NVIDIA Tesla V100 GPUs. Each GPU has 32GB of memory and is linked with Power9 processors, clocking at 3.15 GHz maximum. For each model, we used 4 GPUs and an effective total batch size of 512 samples. For each GPU, the local batch size is set as 64, requiring 2 gradient accumulation steps.

C Uniform-state Discrete Diffusion and Duality with Gaussian

Duality of Uniform and Gaussian. In continuous diffusion modeling, we typically rely on the diffusion mapping of a data distribution q_{data} to a simple prior distribution that is often the standard Gaussian distribution $\mathcal{N}(0, \mathbf{I}_K)$. The marginal distribution of the noisy latent variable $\mathbf{w}_t \sim \tilde{q}_t(\cdot|\mathbf{x})$ at time t is defined as:

$$\mathbf{w}_t \sim \tilde{q}_t(\mathbf{w}_t|\mathbf{x}; \tilde{\alpha}_t) = \mathcal{N}(\mathbf{w}_t; \tilde{\alpha}_t \mathbf{x}, (1 - \tilde{\alpha}_t^2)\mathbf{I}_K), \tag{17}$$

where $\tilde{\alpha}_t \in [0,1]$ is the diffusion parameter that is a monotonically decreasing function in t. The boundary conditions are $\tilde{q}_{t=0} \approx q_{\text{data}}$ and $\tilde{q}_{t=1} = \mathcal{N}(0, \mathbf{I}_K)$.

However, as shown in Sahoo et al. (2025), there exists a connection between the Gaussian and Uniform diffusion processes for the discrete setting. Specifically, we can utilize the operator, $\arg\max : \mathbb{R}^K \to \mathcal{V}$, to map a continuous vector $\mathbf{w} \in \mathbb{R}^K$ to the one-hot vector corresponding to $\arg\max(\mathbf{w}) = \arg\max_{\mathbf{z} \in \mathcal{V}} \mathbf{z}^\top \mathbf{w}$.

Then, we can define the discrete marginals to be $\mathbf{z}_t = \arg\max(\mathbf{w}_t)$ and the conditional probability mass function $p_t(\mathbf{z}_t|\mathbf{x})$ marginalized over $\mathbf{w}_t \sim \tilde{q}_t(\mathbf{w}_t|\mathbf{x}; \tilde{\alpha}_t)$ such that

$$\mathbf{z}_t \sim P_t\big(\mathbf{z}_t|\mathbf{x}; \mathcal{T}(\tilde{\alpha}_t)\big) = \text{Cat}\bigg(\mathbf{z}_t; \mathcal{T}(\tilde{\alpha}_t)\mathbf{x} + (1 - \mathcal{T}(\tilde{\alpha}_t))\frac{\mathbf{1}}{K}\bigg), \tag{18}$$

where $\mathcal{T} : [0,1] \to [0,1]$ is the Gaussian Diffusion Transformation operator. This operator is defined as

$$\alpha_t = \mathcal{T}(\tilde{\alpha}_t) = \frac{K}{K-1}\bigg[\int_{-\infty}^{\infty} \phi\bigg(z - \frac{\tilde{\alpha}_t}{\sqrt{1-\tilde{\alpha}_t^2}}\bigg)\Phi^{K-1}(z)\mathrm{d}z - \frac{1}{K}\bigg] \tag{19}$$

where $\phi(z) = \frac{\exp(-z^2)}{\sqrt{2\pi}}$ is the standard Normal distribution and $\Phi(z) = \int_{-\infty}^{z} \phi(t)\mathrm{d}t$ is the respective cumulative distribution.

Overall, there exists a fundamental connection between Uniform-state discrete and Gaussian diffusion processes, shown in Sahoo et al. (2025). Specifically, they defined this formal connection as

$$\mathbf{z}_t \sim q_t(\mathbf{z}_t|\mathbf{x}; \mathcal{T}(\tilde{\alpha}_t)) = [\arg\max]_* \, \tilde{q}_t(\mathbf{w}_t|\mathbf{x}; \tilde{\alpha}_t) \tag{20}$$

where $*$ denotes the push-forward of the K-dimensional Gaussian density $\tilde{q}_t$ under $\arg\max$ which yields a categorical distribution of K categories.

D Conditional Entropy and Curvature

In this section, inspired by Biroli et al. (2024) and D'Amico et al. (2025), we attempt to relate entropy and the curvature of the energy in the continuous setting, using local approximation, to show there exists a connection between these two ideas. Here, assume that the clean data $\mathbf{x} \in \mathbb{R}^d$ and its perturbed version $\mathbf{z}_t \in \mathbb{R}^d$ at time t.

Proof Sketch. Consider the conditional distribution $p(\mathbf{x}|\mathbf{z}_t)$ defined by an energy function $E(\mathbf{x}; \mathbf{z}_t)$:

$$p(\mathbf{x}|\mathbf{z}_t) = \frac{1}{Z(\mathbf{z}_t)} e^{-E(\mathbf{x}; \mathbf{z}_t)}, \tag{21}$$

where $Z(\mathbf{z}_t) = \int e^{-E(\mathbf{y}, \mathbf{z}_t)}\mathrm{d}\mathbf{y}$ is the partition function. We assume the distribution is peaked around a mode $\mathbf{x}^*$, representing the most likely clean data point given the noisy observation $\mathbf{z}_t$.

To analyze the local geometry, we perform a second-order Taylor expansion of the energy $E(\mathbf{x}; \mathbf{z}_t)$ with respect to $\mathbf{x}$, centered around the mode $\mathbf{x}^*(\mathbf{z}_t)$:

$$E(\mathbf{x}; \mathbf{z}_t) \approx E(\mathbf{x}^*; \mathbf{z}_t) + (\mathbf{x} - \mathbf{x}^*)^\top \nabla_{\mathbf{x}} E(\mathbf{x}^*; \mathbf{z}_t) + \frac{1}{2}(\mathbf{x} - \mathbf{x}^*)^\top \mathbf{H}(\mathbf{z}_t)(\mathbf{x} - \mathbf{x}^*). \tag{22}$$

Since $\mathbf{x}^*$ is a local minimum of the energy surface defined by $\mathbf{z}_t$, the gradient $\nabla_{\mathbf{x}} E(\mathbf{x}^*; \mathbf{z}_t)$ vanishes. The matrix $\mathbf{H}(\mathbf{z}_t) = \nabla_{\mathbf{x}}^2 E(\mathbf{x}^*; \mathbf{z}_t)$ is the Hessian of the energy, representing the local curvature or *sharpness* of the energy basin conditioned on $\mathbf{z}_t$.

Using Eq. (22) and assuming that we are at the minimum where $\nabla_{\mathbf{x}} E(\mathbf{x}^*; \mathbf{z}) = \mathbf{0}$, we can perform Laplace approximation for the partition function $Z(\mathbf{z}_t)$:

$$\begin{aligned} Z(\mathbf{z}_t) &\approx \int e^{-\left(E(\mathbf{x}^*; \mathbf{z}_t) + \frac{1}{2}(\mathbf{x} - \mathbf{x}^*)^\top \mathbf{H}(\mathbf{z}_t)(\mathbf{x} - \mathbf{x}^*)\right)} \mathrm{d}\mathbf{x} \\ &= e^{-E(\mathbf{x}^*; \mathbf{z}_t)} \int e^{-\frac{1}{2}(\mathbf{x} - \mathbf{x}^*)^\top \mathbf{H}(\mathbf{z}_t)(\mathbf{x} - \mathbf{x}^*)} \mathrm{d}\mathbf{x} \\ &= (2\pi)^{\frac{d}{2}} \, e^{-E(\mathbf{x}^*; \mathbf{z}_t)} \det\big(\mathbf{H}(\mathbf{z}_t)\big)^{-\frac{1}{2}} \end{aligned} \tag{23}$$

The conditional entropy $\mathcal{H}(\mathbf{x}|\mathbf{z}_t)$ can now be defined. Using the relationship $\log p = -E - \log Z$ and substitute it into $\mathcal{H}(\mathbf{x}|\mathbf{z}_t)$, we have

$$\begin{aligned}\mathcal{H}(\mathbf{x}|\mathbf{z}_t) &= \mathop{\mathbb{E}}_{\mathbf{x}\sim p(\mathbf{x}|\mathbf{z}_t)}\Big[E(\mathbf{x};\mathbf{z}_t) + \log Z(\mathbf{z}_t)\Big] \\ &= \mathop{\mathbb{E}}_{\mathbf{x}\sim p(\mathbf{x}|\mathbf{z}_t)}\Big[E(\mathbf{x};\mathbf{z}_t) + \frac{d}{2}\log(2\pi) - E(\mathbf{x}^*;\mathbf{z}_t) - \frac{1}{2}\log\big[\det\big(\mathbf{H}(\mathbf{z}_t)\big)\big]\Big]\end{aligned} \tag{24}$$

If we substitute Eq. (22) into the term $E(\mathbf{x};\mathbf{z}_t)$ in Eq. (24), we then have

$$\begin{aligned}\mathcal{H}(\mathbf{x}|\mathbf{z}_t) &\approx \mathop{\mathbb{E}}_{\mathbf{x}\sim p(\mathbf{x}|\mathbf{z}_t)}\Big[\cancel{E(\mathbf{x}^*;\mathbf{z}_t)} + \frac{d}{2}\log(2\pi) \cancel{- E(\mathbf{x}^*;\mathbf{z}_t)} - \frac{1}{2}\log\big[\det\big(\mathbf{H}(\mathbf{z}_t)\big)\big] + C\Big] \\ &\approx \mathop{\mathbb{E}}_{\mathbf{x}\sim p(\mathbf{x}|\mathbf{z}_t)}\Big[-\frac{1}{2}\log\big[\det\big(\mathbf{H}(\mathbf{z}_t)\big)\big] + C\Big]\end{aligned} \tag{25}$$

where C is a constant involving the omitted terms from our substitution of Eq. (22).

Discussion. Overall, this derivation highlights that the conditional entropy is inversely proportional to the log-determinant of the Hessian at the mode, and aligns well to the findings of Biroli et al. (2024) where a collapse in entropy corresponds to the system getting trapped in small-disjoint regions of the configuration space. However, in this work, we are exploring UDDMs, which are not continuous DMs. Thus, we lack formulations that attempt to link up their conditional entropy with the sharpness in the discrete setting of language or text modeling. But we suspect that the connection between Uniform and Gaussian distributions in the discrete setting, laid out by Sahoo et al. (2025), provides some clues to further extend this aspect later.

E ADDITIONAL RESULTS

E.1 CONDITIONAL ENTROPY HISTOGRAMS

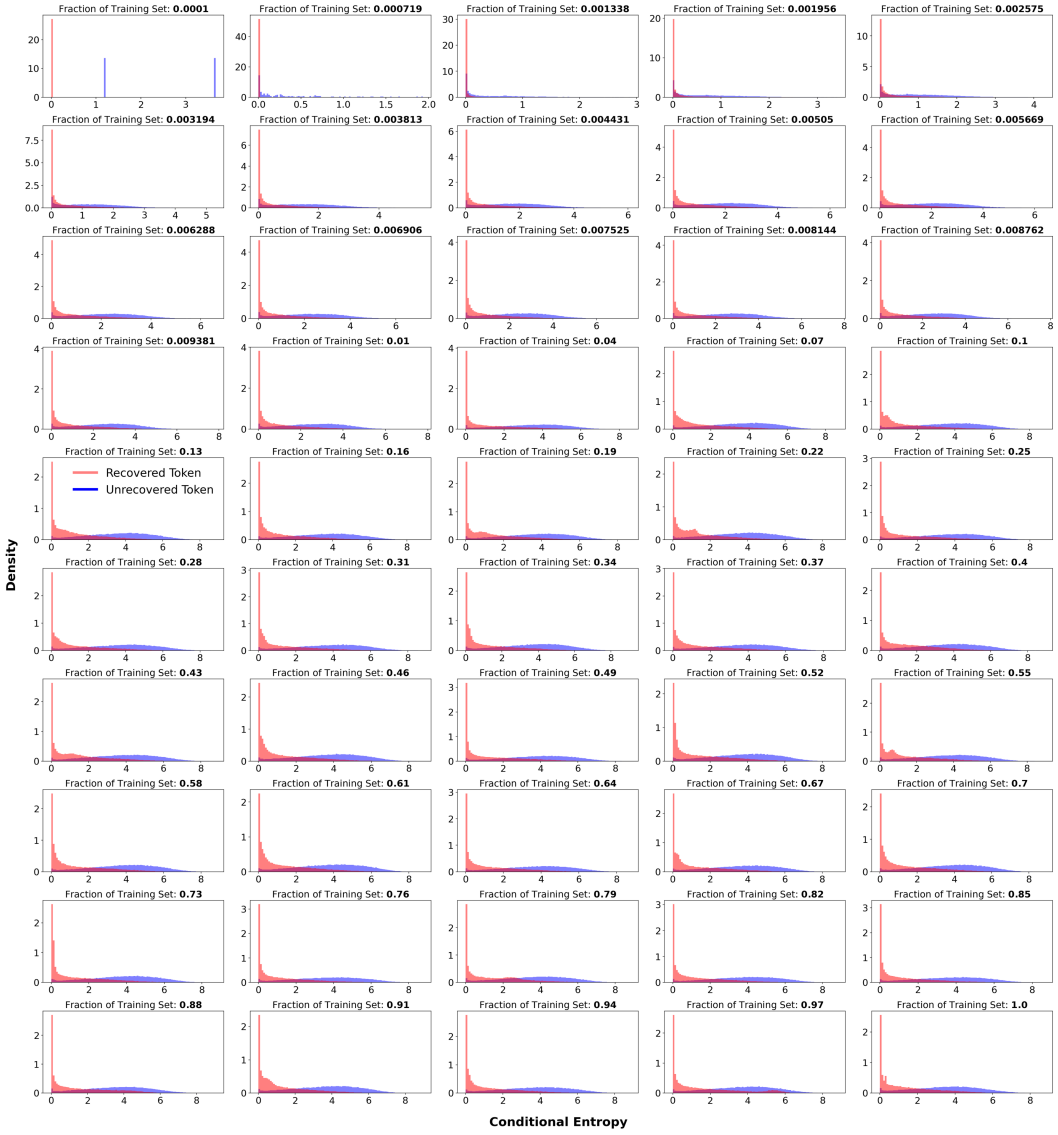


Figure 3: An illustration of the density of conditional entropy for two categories of tokens, *recovered* and *unrecovered*, computed at $t = 0.25$ for the *Tiny* model. The subplots are ordered by the fraction of training dataset, ranging from 0.0001 (top-left) to 1.0 (bottom-right). As the fraction of training data increases, recovered tokens concentrate near zero entropy (high confidence), while unrecovered tokens exhibit a broad distribution at higher entropy.

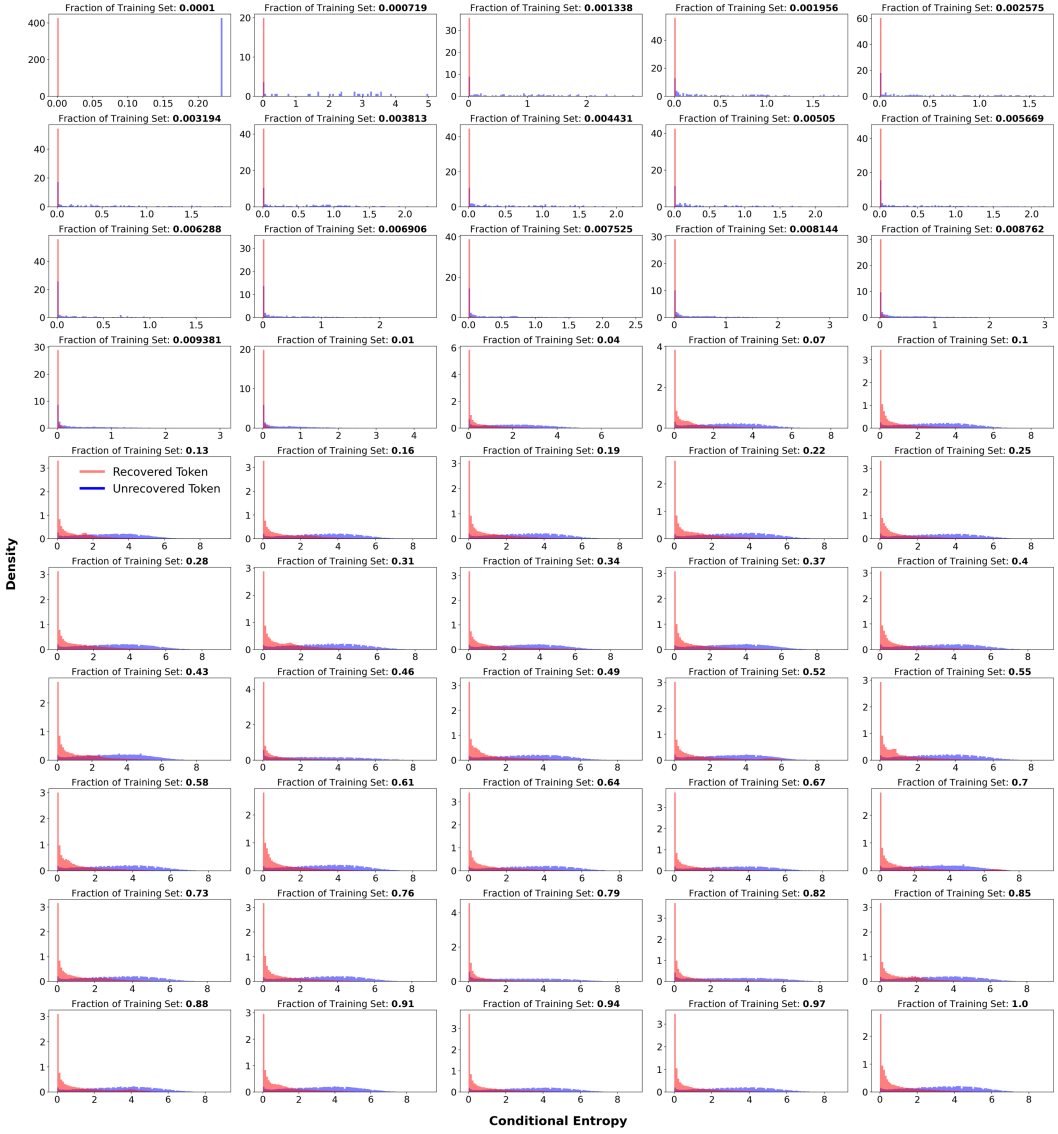


Figure 4: An illustration of the density of conditional entropy for two categories of tokens, *recovered* and *unrecovered*, computed at $t = 0.25$ for the *Small* model. The subplots are ordered by the fraction of training dataset, ranging from 0.0001 (top-left) to 1.0 (bottom-right). As the fraction of training data increases, recovered tokens concentrate near zero entropy (high confidence), while unrecovered tokens exhibit a broad distribution at higher entropy.

Figure 5: An illustration of the density of conditional entropy for two categories of tokens, *recovered* and *unrecovered*, computed at $t = 0.25$ for the *Medium* model. The subplots are ordered by the fraction of training dataset, ranging from 0.0001 (top-left) to 1.0 (bottom-right). As the fraction of training data increases, recovered tokens concentrate near zero entropy (high confidence), while unrecovered tokens exhibit a broad distribution at higher entropy.

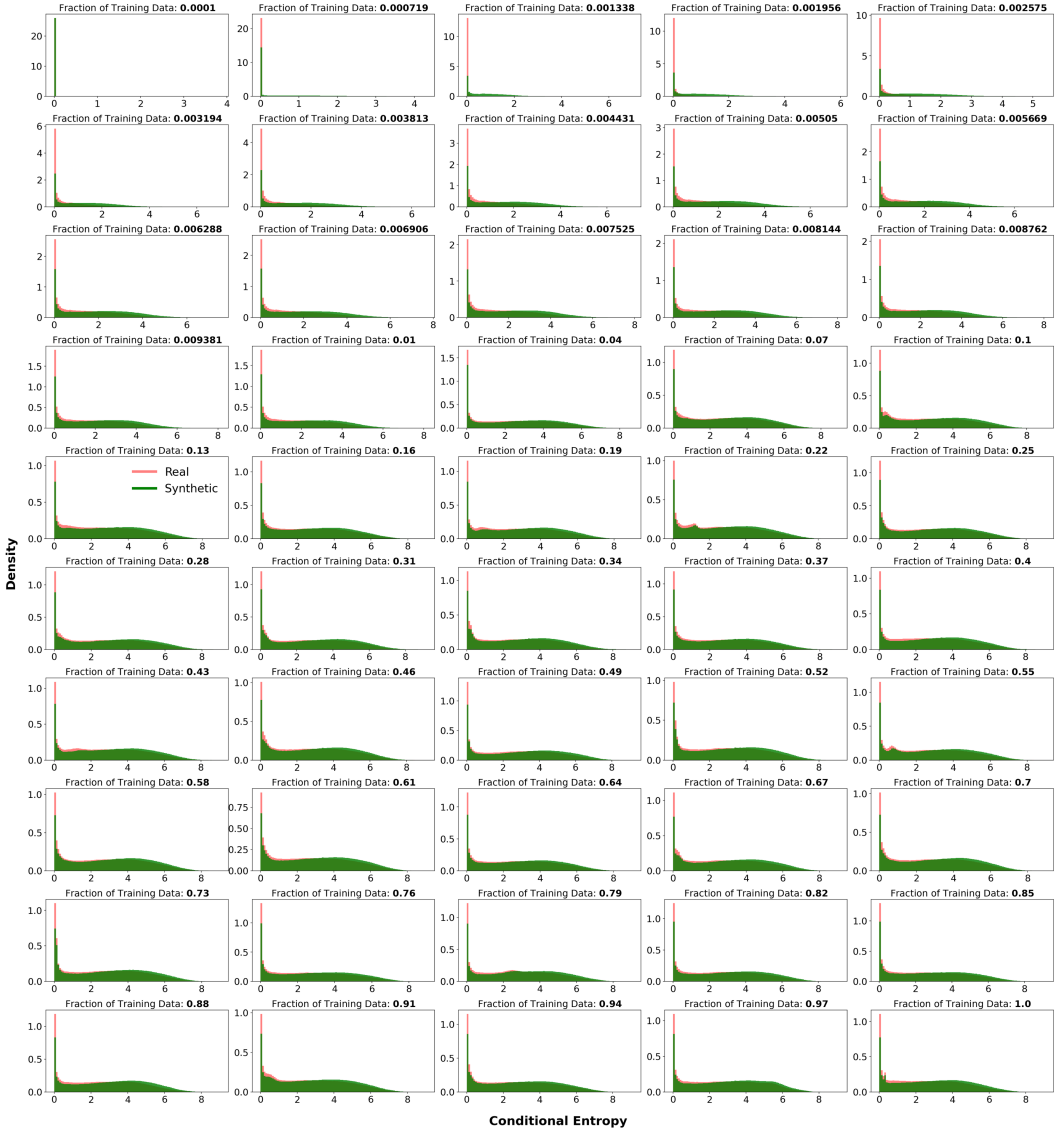


Figure 6: An illustration of the evolution of the density of the conditional entropy for the probabilities of training and synthetic tokens, computed at $t = 10^{-5}$ using the *Tiny* models, as the training dataset size grows. In the memorization phase, when the fraction of the training dataset is small, most tokens have very low conditional entropy. In contrast, during the generalization phase, many tokens have high conditional entropy, but low-entropy tokens still remain.

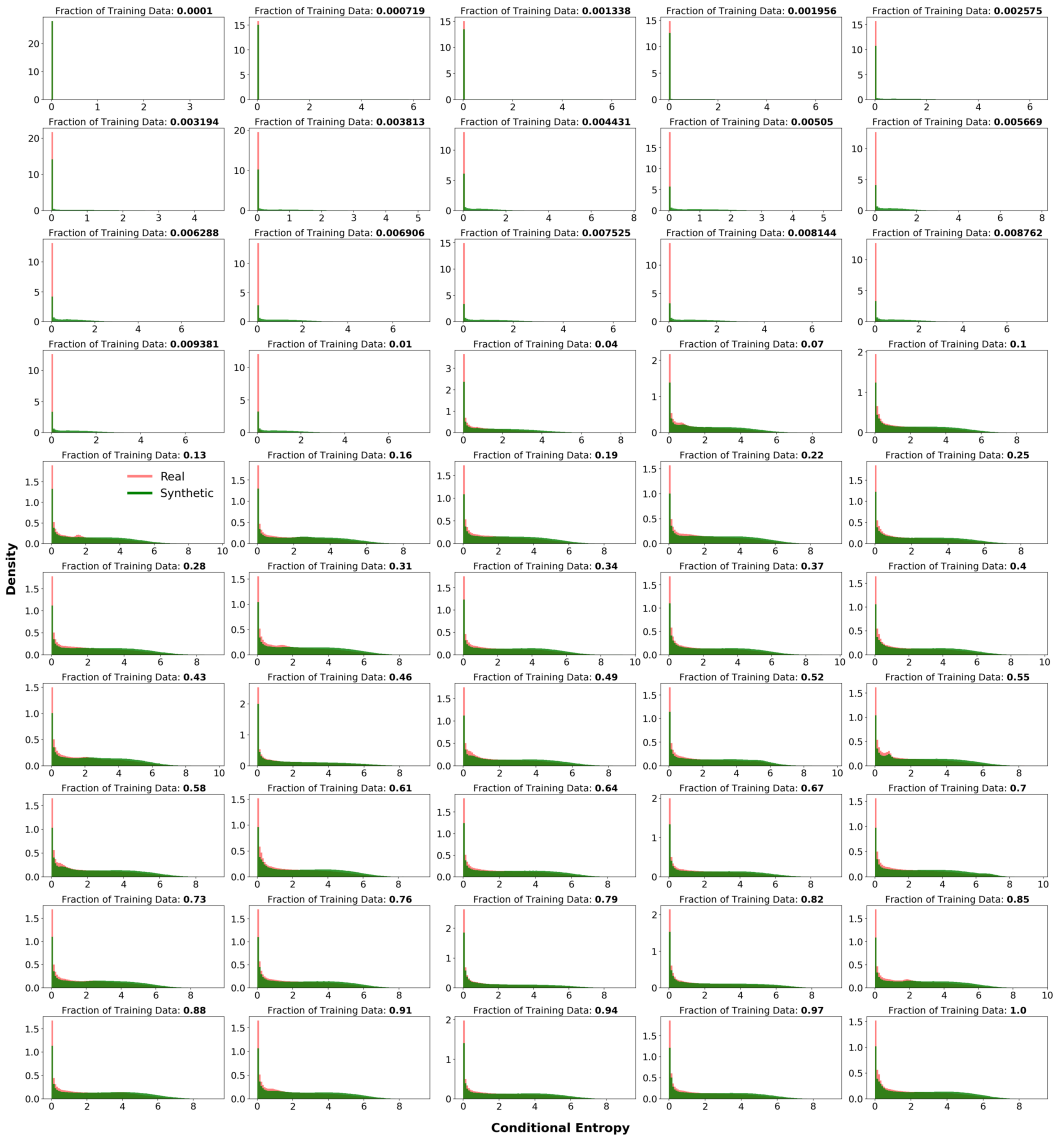


Figure 7: An illustration of the evolution of the density of the conditional entropy for the probabilities of training and synthetic tokens, computed at $t = 10^{-5}$ using the *Small* models, as the training dataset size grows. In the memorization phase, when the fraction of the training dataset is small, most tokens have very low conditional entropy, nearing zero value. In contrast, during the generalization phase, many tokens have high conditional entropy, but low-entropy tokens still remain.

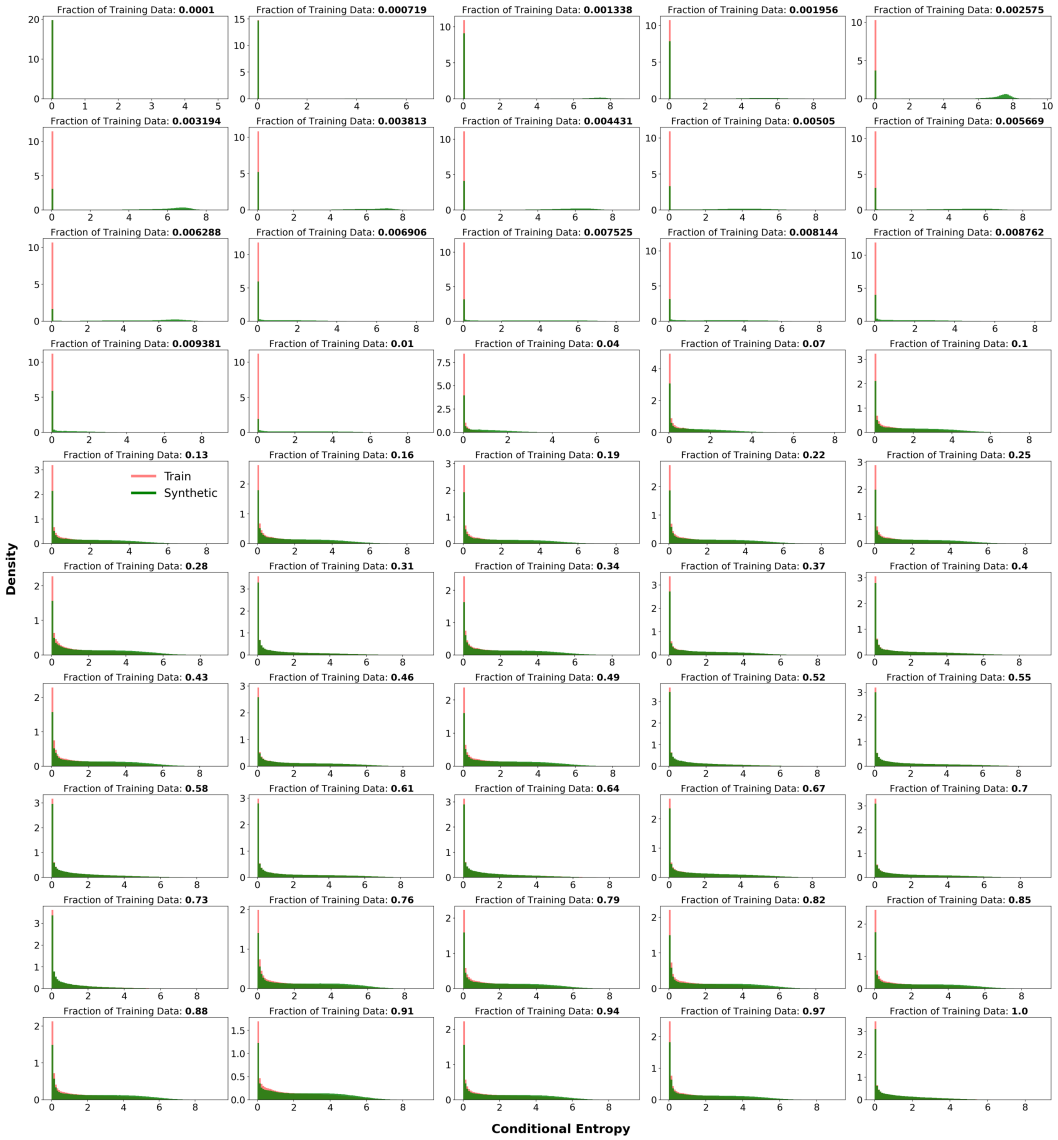


Figure 8: An illustration of the evolution of the density of the conditional entropy for the probabilities of training and synthetic tokens, computed at $t = 10^{-5}$ using the *Medium* models, as the training dataset size grows. In the memorization phase, when the fraction of the training dataset is small, most tokens have very low conditional entropy, nearing zero value. In contrast, during the generalization phase, many tokens have high conditional entropy, but low-entropy tokens still remain.

E.2 VISUALIZATIONS OF TEXT RECOVERY EXAMPLES

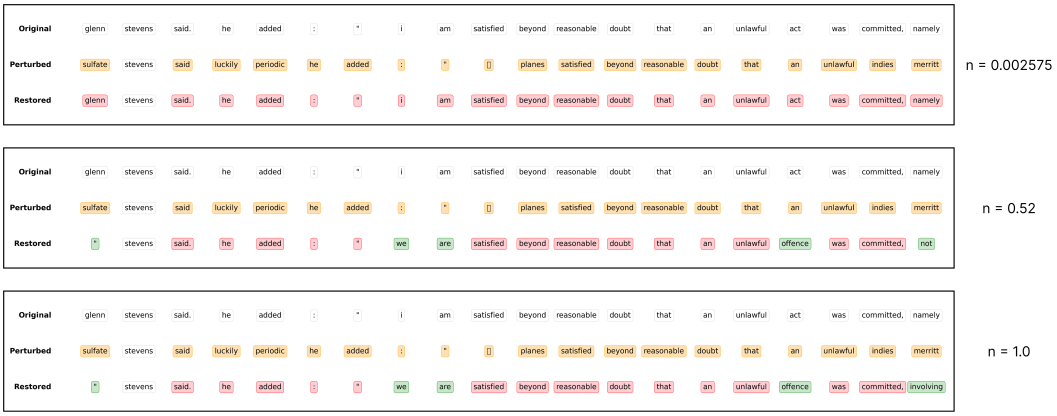


Figure 9: An illustration of the model's ability to recover tokens from perturbed sequences on **training examples** at different fractions of the training dataset, denoted as n. Perturbation is computed at $t = 0.25$ and the reverse process is performed afterwards. The restored row displays the model's output with color-coded tokens: orange denotes perturbed tokens, red indicates successfully recovered tokens, green indicates failed recoveries, and purple represents tokens that are changed but were not originally perturbed. As n increases, the model's ability to recover perturbed tokens becomes worse and in contrast, its generative ability improves.

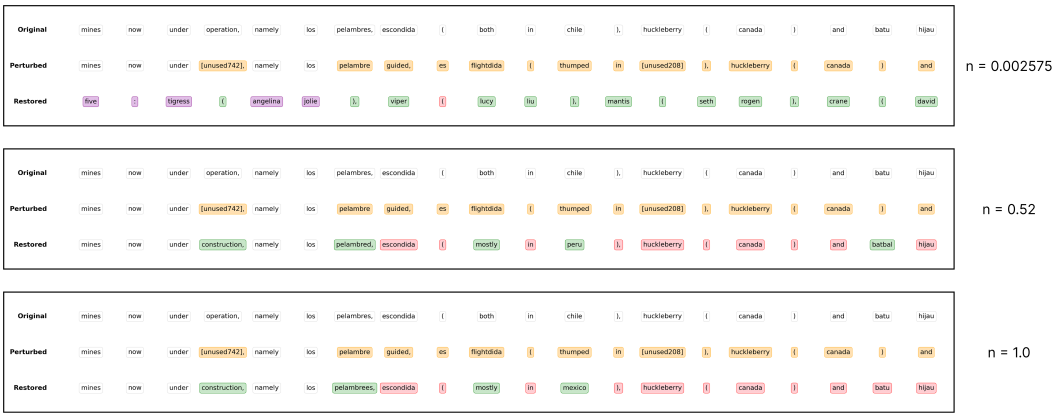


Figure 10: An illustration of the model's ability to recover tokens from perturbed sequences on **test examples** at different fractions of the training dataset, denoted as n. Perturbation is computed at $t = 0.25$ and the reverse process is performed afterwards. The restored row displays the model's output with color-coded tokens: orange denotes perturbed tokens, red indicates successfully recovered tokens, green indicates failed recoveries, and purple represents tokens that are changed but were not originally perturbed. As n increases, the model shows signs of generality, where it can recover perturbed tokens given unseen examples.

E.3 Perplexity as Training Dataset Size Increases

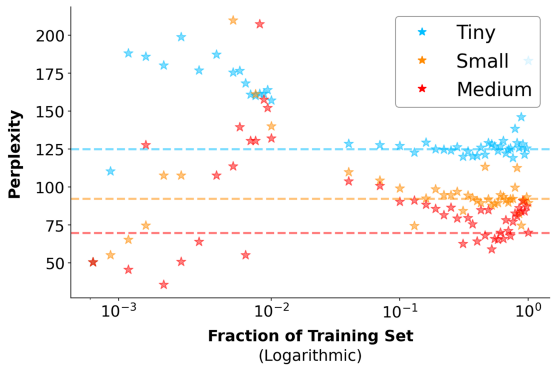


Figure 11: Perplexity across varying fractions of the training dataset for the *Tiny*, *Small*, and *Medium* models. The plot illustrates an initial noisy trend at small training dataset fractions, which mirrors the conditional entropy behavior observed earlier for the *Medium* models in Fig. (2). The horizontal **dashed lines** indicate the stabilized perplexity values achieved by each model size as the training dataset grows, which suggests that increasing the model's size leads to better perplexity.

To further characterize the learning dynamics of UDDMs, we evaluated the perplexity of the *Tiny*, *Small*, and *Medium* models across varying fractions of the training dataset. As illustrated in Fig. (2), the perplexity, computed using `GPT-2 Large` model (Radford et al., 2019) as the reference model, exhibits an initial noisy trend at lower training dataset fractions. This instability is particularly pronounced in the *Medium* model, mirroring the high-variance conditional entropy behavior observed during the early stages of the memorization phase.

However, as the size of the training dataset increases and the models transition toward generalization, the perplexity scores steadily decrease and stabilize. Notably, the models converge to distinct, model's size-dependent perplexity baselines, with the Medium model achieving the lowest perplexity at the full dataset size. This confirms that the transition to the generalization regime aligns with an overall improvement in the model's predictive certainty and scaling law (Kaplan et al., 2020).