CAN LLMS LEARN TO GENERALIZE BEFORE MEMORIZING HIGH-ORDER MARKOV CHAINS?

Anonymous authors
Paper under double-blind review

ABSTRACT

Early stopping is a widely-used implicit regularization technique that halts training before full optimization of training to avoid overfitting. In supervised learning it is an effective safeguard against overfitting, and it has recently been identified as a key ingredient of generalization in diffusion models. We ask whether early stopping can similarly prevent memorization when causal language models are trained on sequences generated by order-k Markov chains. Our answer is mixed. For low-order chains, early stopping leads to generalization, but for high-order chains it is not sufficient. Theoretically, we establish a regularization sufficient for learning order-1 chains, which implicitly holds early in training. Empirically, we find that for high-order chains this implicit regularization vanishes early in training, so stopping fails to prevent memorization.

1 INTRODUCTION

Early stopping is a reliable technique for avoiding overfitting in machine learning. Its critical impact on training has inspired seminal work on understanding its role in the learning mechanism, including Amari et al. (1997). Yao et al. (2007) prove that early stopping for kernel regression imposes an implicit regularization that is effective in avoiding overfitting. This effect is not only essential for supervised learning but even more important for generative models.

For diffusion models, which are trained with objectives that explicitly encourage reproduction of training data, Kamb & Ganguli (2024) proved that the optimal model trained on a finite dataset reproduces the empirical data distribution. Bonnaire et al. (2025) argue that generalization in diffusion models can emerge as a consequence of early stopping. But what is the analogous mechanism in large language models? Next-token prediction language models are likewise trained with reconstruction or prediction objectives that can promote memorization; for example, causal language modeling trains models to predict the next token following prefixes from their training sequences. This raises a fundamental question: *Does generalization in language models similarly arise from the implicit regularization induced by early stopping?*

We study the interplay between generalization and memorization in a controlled family of synthetic corpora generated by Markov chains. Each corpus consists of sequences, $\boldsymbol{x} := x_1, \ldots, x_n$ over a vocabulary $\mathcal{V}$, with transition probabilities $p(x_i \mid x_{i-1}, \ldots, x_{i-k})$. The transition matrix is sampled randomly, with each row drawn independently from $\mathrm{Dirichlet}(0.5)$. The order k serves as a control knob for corpus complexity. As k increases, the sequence distribution exhibits increasingly long-range and complex dependencies. Similar synthetic settings were used by Edelman et al. (2024) to study the emergence of in-context learning in language models with a slight difference: we use data generated from a single Markov chain, whereas they train models on sequences generated from multiple chains. This distinction reflects our different objective; our primary goal is to study memorization rather than in-context learning.

To analyze data memorization during training, we define two distinct critical times in the training dynamics: (1) a *generalization time*, at which the model logits accurately estimate the underlying Markov transition kernel, and (2) a *memorization time*, beyond the model starts to assign increasing likelihood to the training examples. These two critical times parallel analogous phenomena observed in the training of diffusion models by Bonnaire et al. (2025) and perviously observed and reported for low order Markov chains in prior study of in-context learning on language models (Gibson et al., 2026), but highlight a limitation that emerges for learning high order Markov chains: as the

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.

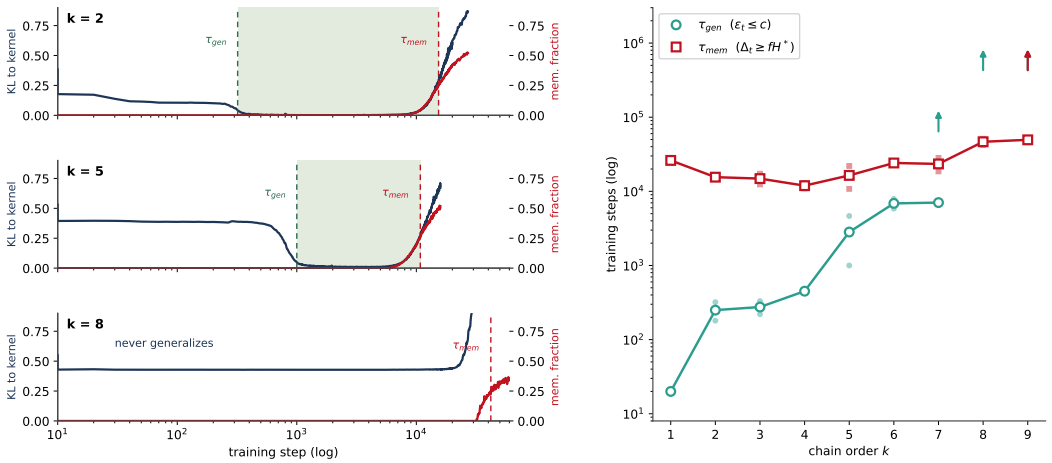


Figure 1: **The generalization–memorization transition.** Left: Training dynamics at $N = 8$K for increasing chain order k. The dark line shows the kernel-estimation error ϵ_t, the red line the memorization margin as a fraction of the reference loss, $\Delta_t/\mathcal{L}_{\text{training}}(p)$. The green area is the window $[\tau_{\text{gen}}, \tau_{\text{mem}}]$, which narrows as k grows. Right: Growth of the two characteristic times with k at the same N. τ_{gen} grows steeply with k while τ_{mem} moves by only a small factor. Arrows mark runs where a criterion is never met within the budget.

Markov order k increases, the memorization time grows more slowly than the generalization time. This is illustrated in Figure 1 from our experiments training GPT2 (Radford et al., 2019) for next-token prediction (see Appendix A for details). As the Markov chain order increases, there is no point where generalization occurs before memorization (which we observe at $k = 8$). Finally, we investigate the mechanism underlying generalization for low-order Markov chains and show that it is associated with the incremental learning of dependencies from low to high order. We then show that this training bias extends to language models trained on natural-text corpora.

Summary of contributions: Our work makes methodological, scientific, and technical contributions. *Methodologically*, we leverage controlled-corpus settings to systematically study the interplay between training dynamics, generalization, and data memorization. *Scientifically*, we demonstrate a limitation of early stopping when learning high-order Markov chains (highlighted by Figure 1). *Technically*, Theorem 2 characterizes the regularization required for generalization on first-order chains ($k = 1$); then we experimentally show that this regularization holds early in training (Observation 1 in Section 4). Section 4 experimentally shows how the fast vanishing speed of this implicit regularization leads to data memorization on high order chains.

2 Generalization and Memorization Time

We formalize the notions of generalization time and memorization time. Time is indexed by the number of optimization steps: f_t denotes the transformer's estimate of the conditional distribution after t training iterations. Because the true transition matrix is known, we can determine how well the language model matches the transition kernel p. We quantify estimation error as

$$\epsilon_t := \frac{1}{n-k}\,\mathbb{E}_{x\sim\mathcal{D}_{\text{test}}} \sum_{i=k+1}^{n} \mathrm{KL}\Big(p(\cdot \mid x_{i-1},\ldots,x_{i-k}) \,\Big\|\, f_t(\cdot \mid x_{i-1},\ldots,x_1)\Big). \tag{1}$$

where KL denotes KL divergence and $\mathcal{D}_{\text{test}}$ denotes the population density function, which we estimated with N i.i.d. samples drawn independently from training samples. We observe that ϵ_t often exhibits a U-shaped trajectory (see Figure 1): it initially decreases, reaches a minimum, and subsequently increases with continued training. This U-shaped curve is a simple analogue of the classical overfitting curve in training of neural network (Morgan & Bourlard, 1989). During the decreasing phase, the transformer progressively recovers the true transition kernel and therefore generalizes to unseen sequences generated from the same Markov process.

We call first time step where the model provides a good estimate of the true kernel the *generalization time*. We define the *memorization time* as the time step at which the model overestimates the density of training sequences relative to the true underlying density. We define these times formally below, and then study how the Markov order k affects both generalization and memorization time.

Definition 1 (Generalization time)**.** We define *generalization time*, denoted by τ_{gen}, as the first time step where ϵ_t is below a threshold:

$$\tau_{\text{gen}} := \min \{t : \epsilon_t \leq c\} \tag{2}$$

where $c \in \mathbb{R}_+$ is an absolute constant.

Marking the generalization time is particularly important, since the model's estimate of the true kernel changes with overtraining; see Figure 1. We argue that this shift in the kernel estimate occurs when the model starts to memorize. We use the difference between training and the true density (denoted by p) loss as an indictor for memorization

$$\Delta_t := \mathcal{L}_{\text{training}}(p) - \mathcal{L}_{\text{training}}(f_t) \tag{3}$$

The next section gives an explicit expression for the training objective. A positive Δ_t therefore means f_t assigns the training sequences more probability than p does, which is possible only by fitting them.

Definition 2 (Memorization time)**.** We define *memorization time*, denoted by τ_{mem}, as the first time step when the difference between training loss and the true density loss is greater than the prescribed threshold $c' \in \mathbb{R}_+$:

$$\tau_{\text{mem}} := \min \{t : \Delta_t \geq c' \cdot \mathcal{L}_{\text{training}}(p)\} . \tag{4}$$

where $c' \in \mathbb{R}_+$ is an absolute constant.

We use $c = 0.05$ nats and $c' = 0.02$ in all experiments, and require each condition to hold on three consecutive evaluations. The error (Equation 1) of the best predictor that ignores context is 0.14–0.45 nats across our chains, well above c, and τ_{mem} moves by at most one evaluation interval.

Transition from Generalization to Memorization. For low-order Markov chains, we observe that the model moves from estimating the true kernel to memorizing the training data (Figure 2). This transition is marked by the generalization and memorization times. In particular, we observe that τ_{mem} coincides with the point at which ϵ_t begins to increase, suggesting that the misestimation of the true kernel under overtraining is driven by data memorization. In the next section, we explain why generalization via early stopping occurs only for low-order Markov chains, and investigate why the same transition does not occur for high-order Markov chains, as shown in Figure 1.

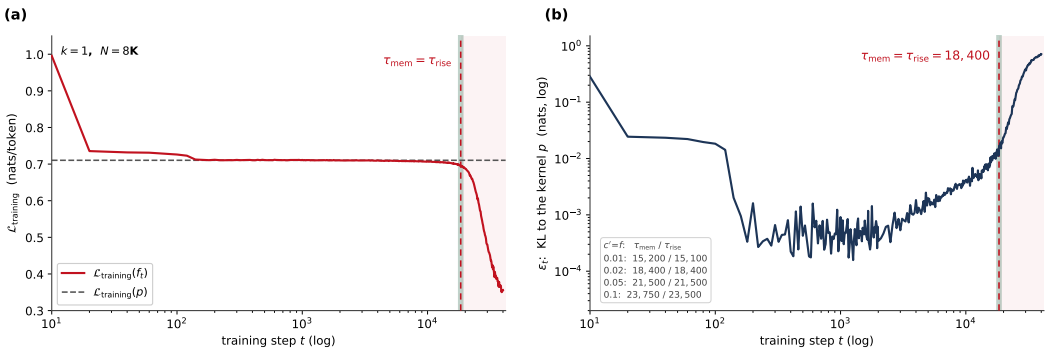


Figure 2: **Memorization begins where kernel estimation stops improving.** Order-1 chain, $N = 8$K. Left: training loss $\mathcal{L}_{\text{training}}(f_t)$ against the true kernel's loss on the same sequences. The loss reaches the kernel's value early and stays on it until τ_{mem}, after which it falls below it. Right: kernel-estimation error ϵ_t. It descends to a minimum then turns sharply upward at τ_{rise}, the first sustained step at which ϵ_t has recovered a fraction c' of its eventual rise. We observe that $\tau_{\text{mem}} = \tau_{\text{rise}}$. Shading marks the memorization phase.

3 LAZY SENSITIVITY AS IMPLICIT REGULARIZATION

The transition described in Section 2 leaves a mechanistic question open. This is surprising since the model are trained to memorize. In particular, the training loss is defined as

$$\mathcal{L}_{\text{training}}(f) := \frac{1}{n}\mathbb{E}_{x\sim\mathcal{D}_{\text{training}}}\left[-\log f(x)\right]. \tag{5}$$

where $\mathcal{D}_{\text{training}}$ denotes the empirical distribution over M i.i.d. samples. The objective is trained over a parametric set of of density functions such as transformers which we denote by $\mathcal{P}$, obeying the chain rule of probability as

$$\forall f \in \mathcal{P} : f(x) := f(x_n, \ldots, x_1) = f(x_1)\prod_{i=2}^{n} f(x_i \mid x_{<i}). \tag{6}$$

Since language models are trained only to estimate conditional probabilities, we can assume that $f(x_i) = p(x_i)$ for $f \in \mathcal{P}$. Note that the minimizer of $\mathcal{L}_{\text{training}}$ is not the true underlying transition kernel p in general cases as we will discuss in the next section. Yet training passes through an extended phase in which the model instead recovers the true kernel. Something other than the objective must be responsible for that phase.

We focus on explaining why this learning occurs for $k = 1$. Recall f_t is obtained after t optimization steps. Our hypothesis is that early in training the transformer is insensitive to all but the most recent token,

$$f_t(x_i \mid x_{i-1}, \ldots, x_1) \;\approx\; f_t(x_i \mid x_{i-1}) \qquad \text{for small } t, \tag{7}$$

a property we call *lazy sensitivity*. We show that if Equation 7 holds exactly, then minimizing the training objective cannot memorize: it returns a consistent estimator of the transition kernel, with an explicit rate. It is the constraint, not the objective, that encourages generalization.

3.1 THE LAZY-SENSITIVITY CLASS

Recall from Section 2 that $\mathcal{P}$ denotes the conditional distributions over $\mathcal{V}$ given a full history, with $p_\theta \in \mathcal{P}$ for every θ. We define the subclass that depends only on the previous token as

$$\mathcal{P}_1 \;:=\; \left\{f \in \mathcal{P} \;:\; f(\cdot \mid x_{i-1}, \ldots, x_1) = f(\cdot \mid x_{i-1})\right\} \;\subset\; \mathcal{P}, \tag{8}$$

where lazy sensitivity is the statement $p_{\theta_t} \in \mathcal{P}_1$. Restricting attention to the class actually visited along the optimization path, rather than to the full class $\mathcal{P}$ over which the objective is defined is the same strategy used by Pesme & Flammarion (2023) to characterize implicit bias through a constrained minimizer. Here the constraint is on the model's context-dependence rather than on a saddle of the loss landscape.

Throughout this section the chain has order $k = 1$, with kernel $p(b \mid a)$ on $\mathcal{V}$, stationary distribution π, and minimum transition probability $p_{\min} := \min_{a,b\in\mathcal{V}} p(b \mid a) > 0$. The corpus supplies $M := N(n-1)$ transitions. We write $\#(a, b)$ for the number of times b follows a in it the training corpus, so that $\#(a) := \sum_b \#(a, b)$, and

$$\hat{p}(b \mid a) \;:=\; \frac{\#(a, b)}{\#(a)} \tag{9}$$

for the empirical kernel. Finally, we extend the notation Equation 5 and write $\mathcal{L}_{\text{training}}(f)$ for the per-token training loss evaluated at an arbitrary $f \in \mathcal{P}$, so that $\mathcal{L}_{\text{training}}(t) = \mathcal{L}_{\text{training}}(p_{\theta_t})$.

3.2 RESTRICTING THE HYPOTHESIS CLASS RECOVERS THE KERNEL

Consider minimizing the same training loss over each of the two classes $\mathcal{P}$ and $\mathcal{P}_1$,

$$f^\star_{\mathcal{P}} \;:=\; \arg\min_{f\in\mathcal{P}} \mathcal{L}_{\text{training}}(f), \qquad f^\star \;:=\; \arg\min_{f\in\mathcal{P}_1} \mathcal{L}_{\text{training}}(f). \tag{10}$$

The two behave in opposite ways. Over the unrestricted class the minimizer memorizes: with $|\mathcal{V}| = 3$ and $M \approx 2.6 \times 10^4$, essentially every history longer than about $\log_{|\mathcal{V}|} M \approx 9$ tokens occurs

exactly once in the corpus, so $f^\star_{\mathcal{P}}$ places all of its mass on the single continuation that was observed there. This drives $\mathcal{L}_{\text{training}}$ to zero and the memorization margin Δ_t to its maximum. Over $\mathcal{P}_1$ the same objective instead returns an estimate of the kernel, and the rest of this section makes that statement precise.

For any predictor, the expected loss splits into a part no model can change and a distance to the true kernel p. Restricting to $\mathcal{P}_1$ turns that distance into one indexed by the $|\mathcal{V}|$ states, which is the form the finite-sample analysis needs. To state this we define

$$\bar{\mu}(a) \;:=\; \frac{1}{n-1}\sum_{i=1}^{n-1}\Pr[x_i = a] \tag{11}$$

as the average occupancy of state a over the conditioning positions (positive for every a, since $x_1 \sim \text{Unif}(\mathcal{V})$ gives $\bar{\mu}(a) \geq \frac{1}{(n-1)|\mathcal{V}|}$) and $H^\star := \mathbb{E}_{a\sim\bar{\mu}}\, H\big(p(\cdot \mid a)\big)$ for the corresponding average conditional entropy, the irreducible entropy of the chain. Note that $\bar{\mu}$ does not need to be the stationary distribution π. Expectations below are over a sequence $x_{1:n}$ drawn from the chain, so $\mathbb{E}[\mathcal{L}_{\text{training}}(f)]$ is the population loss of a fixed f. In every statement f is assumed to satisfy $f(b \mid x_{i-1}, \ldots, x_1) > 0$ whenever $p(b \mid x_{i-1}) > 0$, so that all divergences are finite.

Theorem 1. *For every $f \in \mathcal{P}_1$,*

$$\mathbb{E}\left[\mathcal{L}_{\textit{training}}(f)\right] \;=\; H^\star \;+\; \mathbb{E}_{a\sim\bar{\mu}}\,\mathrm{KL}\big(p(\cdot \mid a)\,\big\|\, f(\cdot \mid a)\big). \tag{12}$$

The first term does not depend on f, so minimizing the loss over $\mathcal{P}_1$ means minimizing the divergence to p. Consequently $f = p$ is the unique minimizer over $\mathcal{P}_1$, since $\bar{\mu}(a) > 0$ for every a.

The theorem is proven in Appendix B. In words: the population loss is the entropy floor plus an average KL to the true conditional, and inside $\mathcal{P}_1$ that average runs over the $|\mathcal{V}|$ states only, so it is a $\bar{\mu}$-weighted distance between two kernels. The first step of the proof, which holds for any $f \in \mathcal{P}$, evaluates the same KL once per history.

Theorem 1 is a population statement: it describes the loss under infinite data, where minimizing over $\mathcal{P}$ and over $\mathcal{P}_1$ both return p (with $p_{\min} > 0$ every history has positive probability). Training instead minimizes the loss on N sequences, and there the two classes differ. The empirical loss constrains $f \in \mathcal{P}$ only on histories present in the corpus, and Section 3.2 showed that most histories longer than $\log_{|\mathcal{V}|} M$ occur at most once. In contrast, $f \in \mathcal{P}_1$ is constrained on every state a, each seen $\#(a) \approx M\bar{\mu}(a)$ times. The minimizer over $\mathcal{P}_1$ that training can reach is therefore not p but the kernel obtained by counting, which Lemma 1 (proven in Appendix B) makes precise:

Lemma 1. *The minimizer of the training objective over $\mathcal{P}_1$ is $f^\star = \hat{p}$, uniquely on every state a with $\#(a) > 0$. Consequently $f^\star = p$ holds only if the empirical transition frequencies coincide exactly with p, an event of probability zero for finite N.*

The standard assumption that the corpus contains every pair $x_i, x_{i-1} \in \mathcal{V}$ asserts only $\#(a, b) \geq 1$. That is what keeps $\mathrm{KL}(p \,\|\, f^\star)$ finite, but it does not make $f^\star$ exact: $\hat{p}$ still deviates from p by $\Theta(\#(a)^{-1/2})$. The relevant question is therefore how fast $f^\star$ approaches p.

Theorem 2. *Fix $\delta \in (0, 1)$. On the event that $\#(a) \geq \frac{8}{p_{\min}} \log \frac{|\mathcal{V}|}{\delta}$ for every a, which holds with probability at least $1 - \delta$ once $M \gtrsim \frac{8}{p_{\min}\,\min_a \bar{\mu}(a)} \log \frac{|\mathcal{V}|}{\delta}$,*

$$\mathbb{E}\Big[\mathbb{E}_{a\sim\bar{\mu}}\,\mathrm{KL}\big(p(\cdot \mid a)\,\big\|\, f^\star(\cdot \mid a)\big)\Big] \;\leq\; \frac{2\,|\mathcal{V}|\big(|\mathcal{V}| - 1\big)}{p_{\min}\, M}. \tag{13}$$

By Theorem 1, the left side of equation 13 is the excess population loss of $f^\star$ over $H^\star$. So $f^\star \to p$ at rate $1/M$, with a $1/p_{\min}$ factor absent from the population identity: $\mathrm{KL}(p\|f)$ penalizes near-zero f most heavily, and those values concentrate on the rarest transitions. This is a byproduct of our proof (bounding KL by χ^2, Lemma 2 in Appendix B), not of the problem: the sharp minimax rate for Markov kernel estimation under KL is $\Theta\big(\frac{|\mathcal{V}|(|\mathcal{V}|-1)}{M}\big)$ with no $p_{\min}$ dependence (Hao et al., 2018). Further remarks on Theorem 2, including why the $p_{\min}$ factor is nonetheless expected under this reference direction and a caution about the experiments, are given alongside its proof in Appendix B.

4 MEMORIZATION WITH RAPID VANISHING OF IMPLICIT REGULARIZATION

Section 3 shows that for $k = 1$ the constraint $f_t \in \mathcal{P}_1$ turns the training objective into a consistent estimator of p. Figure 1 shows that this phenomenon is absent for large k. In this section we measure the constraint directly, as the distance from f_t to $\mathcal{P}_j$, and make three observations. First, for $k = 1$, f_t remains in $\mathcal{P}_1$ for the whole generalizing phase. Second, for $k \geq 2$, f_t leaves $\mathcal{P}_1$ once the loss can decrease outside it. Whether it then generalizes or memorizes depends on k. We define the *horizon* $k^*(N)$ as the largest k for which the generalizing window is nonempty, i.e., $\tau_{\text{gen}} < \tau_{\text{mem}}$ (Figure 1). Third, the same holds on natural text. Each observation is followed by a note relating it to Theorem 1, Theorem 2 and prior work.

Distance to $\mathcal{P}_j$. Extending equation 8, let $\mathcal{P}_j := \{f \in \mathcal{P} : f(\cdot \mid x_{i-1}, \ldots, x_1) = f(\cdot \mid x_{i-1}, \ldots, x_{i-j})\}$ be the class of predictors that depend on at most the j previous tokens, so that $\mathcal{P}_0 \subset \mathcal{P}_1 \subset \cdots \subset \mathcal{P}$. We measure how far a predictor is from the class by

$$d(f, \mathcal{P}_j) := \min_{g \in \mathcal{P}_j} \frac{1}{n-k} \mathbb{E}_{x \sim \mathcal{D}_{\text{test}}} \sum_{i=k+1}^{n} \mathrm{KL}\Big(f(\cdot \mid x_{i-1}, \ldots, x_1) \,\Big\|\, g(\cdot \mid x_{i-1}, \ldots, x_{i-j}) \Big), \quad (14)$$

the KL divergence from f to the nearest member of $\mathcal{P}_j$. The minimizer is the conditional average of f given the last j tokens, so $d(f, \mathcal{P}_j)$ is computable from forward passes alone (Section B.2). We have $d(f, \mathcal{P}_j) = 0$ if and only if $f \in \mathcal{P}_j$ [1], and lazy sensitivity equation 7 is the statement $d(f_t, \mathcal{P}_1) \approx 0$. We say that f_t has left $\mathcal{P}_j$ at the first step at which $d(f_t, \mathcal{P}_j) \geq 10^{-2}$ nats on three consecutive evaluations. The same distance evaluated at the kernel, $d(p, \mathcal{P}_j)$, is the cost of truncating p to j tokens of context. By Theorem 1, $H^\star + d(p, \mathcal{P}_j)$ is the smallest test loss attainable in $\mathcal{P}_j$, and $d(p, \mathcal{P}_k) = 0$. A model that has recovered an order-k kernel therefore satisfies $d(f_t, \mathcal{P}_1) = d(p, \mathcal{P}_1)$, and a value above it is dependence on context that p does not have, which on a finite corpus can only be fit to the training sequences.

4.1 EXPERIMENTS ON SYNTHETIC MARKOV CHAINS

We train one model per order $k = 1, \ldots, 9$. For each k, we draw one chain with $|\mathcal{V}| = 3$, sample $N = 8$K sequences from it, and train for 60K steps with the model and optimizer of Section 2. We then measure ϵ_t and $d(f_t, \mathcal{P}_j)$ for all $j \leq \min(k, 5)$. Figure 3a shows $d(f_t, \mathcal{P}_1)$ for each run, with $d(p, \mathcal{P}_1)$ marked and a circle at the minimum of ϵ_t. Observations 1 and 2 that follow are based on these experiments.

Observation 1: for $k = 1$, f_t remains in $\mathcal{P}_1$. For $k = 1$, $d(f_t, \mathcal{P}_1)$ stays more than an order of magnitude below the exit threshold from initialization through the minimum of ϵ_t, and exceeds it only shortly before τ_{mem} (Figure 3a). Thus, f_t matches p within $\mathcal{P}_1$, and it falls outside $\mathcal{P}_1$ only as memorization begins.

Note: Theorem 1 and Theorem 2 assume $f_t \in \mathcal{P}_1$. The experiment is consistent with the theorem, and Observation 1 verifies the assumption. Learning in stages of increasing order is well documented on Markov data (Edelman et al., 2024), on natural text (Rende et al., 2024), and as near-stationary points of the population loss (Varre et al., 2025); the Boolean staircase (Abbe et al., 2022; 2023) is the analogous statement for sparse functions. What is new here is that the stage is exact membership in $\mathcal{P}_1$, which is the premise the theorems require.

Observation 2: for $k \geq 2$, f_t leaves $\mathcal{P}_1$ when there is loss to be gained. The exit time from $\mathcal{P}_1$ is not monotone in k (Figure 3a). For $2 \leq k \leq 7$, it grows by roughly a factor of 2 per order, whereas a counting estimate of the kernel predicts a factor of $|\mathcal{V}| = 3$. The exits from all intermediate classes happen together: for every k, the exit times from $\mathcal{P}_1$ through $\mathcal{P}_{\min(k,5)}$ fall within a few hundred steps of each other, so f_t does not remain in any class between $\mathcal{P}_1$ and $\mathcal{P}_k$ on its way to the kernel. For $k = 1$ and $k = 8, 9$, f_t stays in $\mathcal{P}_1$ an order of magnitude longer, until τ_{mem}. For $k = 1$, exiting $\mathcal{P}_1$ cannot lower the loss, since $d(p, \mathcal{P}_1) = 0$. For $k \geq 8$, $d(p, \mathcal{P}_1) > 0$ in the population, but this reduction cannot be obtained from $N = 8$K sequences. Figure 3b shows that the reduction

[1] Assuming the expectation over x is taken over population level density.

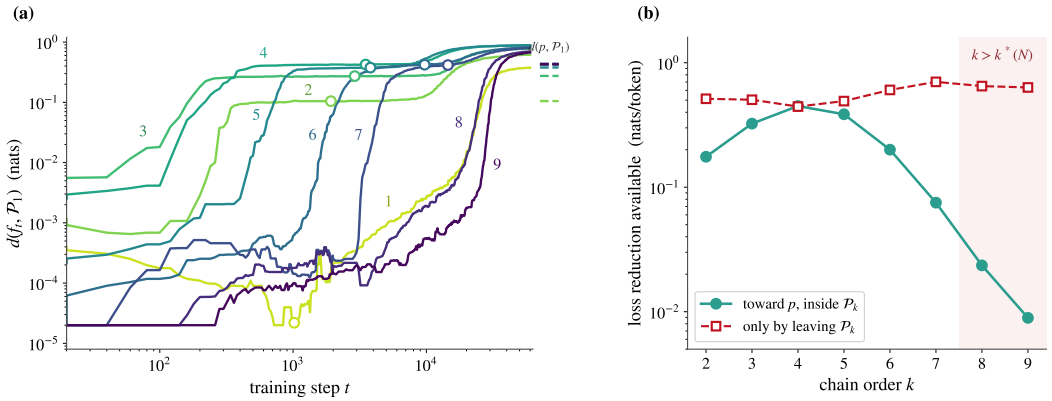


Figure 3: **Distance from f_t to $\mathcal{P}_1$, by chain order.** $N = 8$K, $|\mathcal{V}| = 3$; log–log axes. **(a)** $d(f_t, \mathcal{P}_1)$ Equation 14 over training, labeled by order k. Dashes at the right edge mark $d(p, \mathcal{P}_1)$ (zero for $k = 1$) and circles mark the minimum of ϵ_t. **(b)** Filled: training-loss reduction available inside $\mathcal{P}_k$ (best loss in $\mathcal{P}_0$ minus best loss in $\mathcal{P}_k$). The whitespace shows further reduction from fitting the training sample, down to the final training loss. For $k > 5$, $\mathcal{P}_5$ stands in for $\mathcal{P}_k$. The shaded area marks $k > k^*(N)$, where the filled reduction is two orders of magnitude below the open one.

in $\mathcal{L}_{\text{training}}$ available inside $\mathcal{P}_k$ by moving toward p drops by more than an order of magnitude from $k \leq 5$ to $k \geq 8$, while the reduction available only by fitting the training sequences barely changes with k. Lazy sensitivity is therefore not an obstacle that the data must overcome, but a property induced by order-1 data. It vanishes once the context carries information that can be extracted from the sample.

Below the horizon, f_t moves out of $\mathcal{P}_1$ exactly as far as p requires, and no further. At the minimum of ϵ_t, $d(f_t, \mathcal{P}_1)$ matches $d(p, \mathcal{P}_1)$ and $d(f_t, \mathcal{P}_k)$ is below the exit threshold: the constraint of Section 3 has shifted from $\mathcal{P}_1$ to $\mathcal{P}_k$. The generalizing phase is the interval during which this constraint holds. It ends at τ_{mem}, when $d(f_t, \mathcal{P}_k)$ rises above zero for good, and $d(f_t, \mathcal{P}_1)$ exceeds $d(p, \mathcal{P}_1)$ at about the same time. Above the horizon, there is no plateau: $d(f_t, \mathcal{P}_1)$ rises from zero directly to its memorization value, and ϵ_t never drops below $d(p, \mathcal{P}_0)$. In short, below the horizon the loss decreases by moving toward p inside $\mathcal{P}_k$, and f_t generalizes while it stays there; above it, the only way to decrease the loss is to memorize.

Note: The plateau at $d(p, \mathcal{P}_1)$ is Theorem 1 applied to $\mathcal{P}_k$. While $f_t \in \mathcal{P}_k$, minimizing the objective recovers p. The horizon $k^*(N)$ is where the loss reduction available inside $\mathcal{P}_k$ becomes negligible compared with the reduction from fitting the training sample (Figure 3b). In this sense, the implicit regularization of Section 3 weakens as k grows. f_t does not exit $\mathcal{P}_1$ any earlier. Rather, the loss reduction available inside $\mathcal{P}_k$ shrinks with k, and by $k = 8$ it is negligible. Up to $k = 7$, the in-class reduction is a substantial fraction of the total. At $k = 8$, it drops to a few percent, and generalization ends. Early stopping ends there too, since it requires a phase in which the loss can decrease only by learning structure. Plateaus of predicted height are known for a single ambiguity (Makkuva et al., 2025; Sahasrabudhe, 2026). Here each plateau is the projection of a known higher-order kernel, one per order, which Makkuva et al. (2025) list as an open problem. Epoch-wise double descent (Nakkiran et al., 2020) has the same falling-then-rising shape, but it does not tie the rising branch to a change in what the model conditions on; $d(f_t, \mathcal{P}_1)$ does.

4.2 Experiments on Natural Language

We repeat the analysis on two text sources. The first is a chain distilled from natural text. The first is a Markovify chain of state size 2 over a Shakespearean corpus, for which p and its nearest member in every $\mathcal{P}_j$ are known exactly. The second is Wikipedia, for which no p exists. In place of these we use the corpus n-gram predictors of order $n = 1, \ldots, 6$ as reference points.

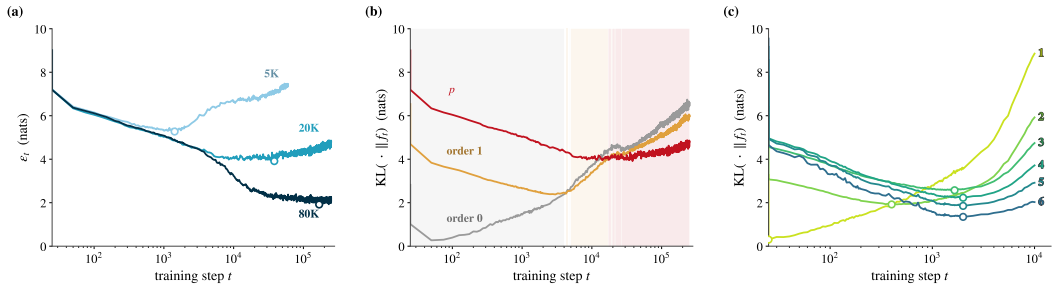


Figure 4: **The transition on text.** **(a)** ϵ_t on the Shakespeare chain for three corpus sizes. Circles mark minima. **(b)** Test KL to f_t from p and its order-0 and order-1 projections, at the middle N. Shading marks the nearest one. **(c)** KL from corpus n-gram predictors to f_t on Wikipedia. Circles mark minima. Colors match Figure 3a, which shows distance to a class rather than to a fixed predictor.

Observation 3: on text, f_t passes through the classes and settles where a kernel exists. Figure 4a plots ϵ_t on the Shakespeare chain for three corpus sizes. The descent follows one curve for every N, and each run leaves it at a τ_{mem} that grows with N. The largest run has not left its minimum within our training budget. Figure 4b plots the test KL to f_t from p and from the members of $\mathcal{P}_0$ and $\mathcal{P}_1$ nearest to p, at the intermediate N. The nearest reference is first the order-0 one, then the order-1 one, then p, and the crossover to p coincides with the minimum of ϵ_t. This is Observation 2 on real words: f_t passes from $\mathcal{P}_0$ to $\mathcal{P}_1$ to $\mathcal{P}_2$, and leaves $\mathcal{P}_2$ for the sample at a time set by N. What N changes is the length of the plateau at p, which is the area in which early stopping is useful.

Figure 4c plots the KL from each n-gram predictor to f_t on Wikipedia. The model comes closest to the n-gram predictors in order of increasing n, each roughly an order of magnitude in steps after the last one. This is again the passage of Observation 2, through the reference predictors in order of increasing depth, but with no kernel there is no class for the constraint to settle in: no plateau forms, and the sequence ends in memorization.

Note: On the Shakespeare text, the distilled chain is the closed-form counterpart of the order-limited corpus clones of Rende et al. (2024). It gives the projections of p and the distances $d(p, \mathcal{P}_j)$ exactly, at the cost of an order-2 upper bound. On the Wikipedia text, with no p, the decomposition of Figure 3b is unavailable and the horizon cannot be located.

5 Related Work

Prior work bearing directly on each of our observations is discussed in the notes following them in Section 4. Here we place the paper against the Markov-chain testbed and the measurement of memorization.

Transformers on Markov data. Sequences from a Markov chain of known order have become a standard probe of what transformers learn. Edelman et al. (2024) train on a prior over first-order chains to study the emergence of in-context learning; Makkuva et al. (2025) characterize the loss landscape for first-order chains and show that single-layer models can be trapped at the unigram solution; Rajaraman et al. (2024) show that constant depth suffices to represent order-k conditionals; and Gibson et al. (2026) sweep the number of training chains and find phases distinguished by whether the model generalizes or memorizes. We train on a single chain rather than a prior over chains, since our object is memorization of a fixed corpus rather than in-context inference, and we use the chain's order as the control on how much context the data reward.

Memorization in language models. Tirumala et al. (2022) measure verbatim reproduction of training examples over training and find that larger models memorize faster and forget less; verbatim reproduction requires no model of the data source and so applies to real corpora, but says nothing about which part of the context is used. Morris et al. (2025) separate the information a model holds about a dataset from the information it holds about the data-generating process, measure the former

as a capacity of 3.5 to 3.6 bits per parameter, and place the onset of double descent where the data exceed that capacity. Their ordering runs from memorization to generalization as data is added at fixed model size; ours is a trajectory at fixed data, in which the model reaches the projections of a known kernel and then departs from them, so the deciding quantity is when the departure happens rather than how many bits are available.

Memorization as a late-time phenomenon. For diffusion models the optimal score on a finite sample reproduces the training data, so memorization is the endpoint of optimization. Bonnaire et al. (2025) show that what prevents it in practice is dynamical: samples become good at an early τ_{gen} and memorization sets in at a later τ_{mem} that grows linearly with the training set size, so the window widens with data. Their analysis is confined to U-Net diffusion models. Our two times are the autoregressive counterpart, and Section 4 identifies why the window closes with the order of the data rather than only how it widens with its size.

Position embeddings. Our models carry no positional encoding. Kazemnejad et al. (2023) show that decoder-only transformers without one remain competitive and often generalize better across lengths, the causal mask alone letting the first layer recover absolute position; removing the embedding weakens a shortcut without disabling the model.

6 Discussion

We began by asking whether early stopping can prevent memorization in language models, as it does in diffusion models. On Markov chains early stopping works as long as there is structure left to learn, and the order k determines how long that is. Our main finding is that early stopping has a limit set by the data. Because τ_{gen} grows steeply with k while τ_{mem} barely changes, there is a horizon, $k^*(N)$, beyond which the model memorizes before it learns the kernel (Section 2). We trace the generalizing phase to an implicit constraint in the model. Early in training, the model's estimate f_t conditions on a single token, and within this class the training objective is a consistent estimator of the true kernel p (Theorem 1, Theorem 2). Measuring this constraint directly shows that it breaks down exactly at the horizon (Section 4). More broadly, generalization and memorization are two ways for the same objective to decrease. Training generalizes as long as the loss can still decrease within f_t's current class, and memorizes once it cannot. Since this can be computed from the corpus before training, the horizon is predictable in advance.

We acknowledge three limitations in our work. First, Theorem 1 and Theorem 2 cover only first-order chains ($k = 1$); for $k \geq 2$ we use them by analogy (Section 4.1), and we have not proved that the argument extends to $\mathcal{P}_k$. Second, both theorems are conditional on $f_t \in \mathcal{P}_1$, and we show this only experimentally (Observation 1), for the chains and models we tried and under a 10^{-2}-nat threshold; we have no proof that training stays in the class early on. Third, we do not know why this implicit regularization vanishes so quickly for high orders, which we identified as the main driver of memorization. The loss reduction available inside $\mathcal{P}_k$ shrinks with k while the reduction from fitting the sample does not (Figure 3b), but this is a correlate, not a mechanism, and it does not predict the horizon $k^*(N)$. A natural hypothesis is that coverage of order-k contexts, which falls exponentially in k, decides whether f_t leaves $\mathcal{P}_1$ toward p or toward the sample; we have not tested it by varying k at fixed coverage.

References

Emmanuel Abbe, Enric Boix-Adsera, and Theodor Misiakiewicz. The merged-staircase property: a necessary and nearly sufficient condition for SGD learning of sparse functions on two-layer neural networks. In *Conference on Learning Theory*, 2022.

Emmanuel Abbe, Enric Boix-Adsera, and Theodor Misiakiewicz. SGD learning on neural networks: leap complexity and saddle-to-saddle dynamics. In *Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, 2023.

Shun-ichi Amari, Noboru Murata, K-R Muller, Michael Finke, and Howard Hua Yang. Asymptotic statistical theory of overtraining and cross-validation. *IEEE transactions on neural networks*, 8 (5):985–996, 1997.

Tony Bonnaire, Raphaël Urfin, Giulio Biroli, and Marc Mézard. Why diffusion models don't memorize: The role of implicit dynamical regularization in training. In *Advances in Neural Information Processing Systems*, 2025. Oral. arXiv:2505.17638.

Benjamin L. Edelman, Ezra Edelman, Surbhi Goel, Eran Malach, and Nikolaos Tsilivis. The evolution of statistical induction heads: In-context learning markov chains. In *Advances in Neural Information Processing Systems*, 2024. arXiv:2402.11004.

Cole Gibson, Wenping Cui, and Gautam Reddy. Distinct mechanisms underlying in-context learning in transformers. *arXiv preprint arXiv:2604.12151*, 2026.

Yi Hao, Alon Orlitsky, and Venkatadheeraj Pichapati. On learning Markov chains. In *Advances in Neural Information Processing Systems*, volume 31, pp. 648–657, 2018. arXiv:1810.11754.

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

Andrej Karpathy. nanoGPT. `https://github.com/karpathy/nanoGPT`, 2022. MIT License.

Amirhossein Kazemnejad, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Payel Das, and Siva Reddy. The impact of positional encoding on length generalization in transformers. In *Advances in Neural Information Processing Systems*, 2023. arXiv:2305.19466.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Ashok Vardhan Makkuva, Marco Bondaschi, Adway Girish, Alliot Nagle, Martin Jaggi, Hyeji Kim, and Michael Gastpar. Attention with markov: A framework for principled analysis of transformers via markov chains. In *International Conference on Learning Representations*, 2025. arXiv:2402.04161.

Nelson Morgan and Hervé Bourlard. Generalization and parameter estimation in feedforward nets: Some experiments. *Advances in neural information processing systems*, 2, 1989.

John X. Morris, Chawin Sitawarin, Chuan Guo, Narine Kokhlikyan, G. Edward Suh, Alexander M. Rush, Kamalika Chaudhuri, and Saeed Mahloujifar. How much do language models memorize? *arXiv preprint arXiv:2505.24832*, 2025.

Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. In *International Conference on Learning Representations*, 2020. arXiv:1912.02292.

Scott Pesme and Nicolas Flammarion. Saddle-to-saddle dynamics in diagonal linear networks. In *Advances in Neural Information Processing Systems*, volume 36, 2023. arXiv:2304.00488.

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

Nived Rajaraman, Marco Bondaschi, Kannan Ramchandran, Michael Gastpar, and Ashok Vardhan Makkuva. Transformers on markov data: Constant depth suffices. *arXiv preprint arXiv:2407.17686*, 2024.

Riccardo Rende, Federica Gerace, Alessandro Laio, and Sebastian Goldt. A distributional simplicity bias in the learning dynamics of transformers. In *Advances in Neural Information Processing Systems*, 2024. arXiv:2410.19637.

Mihir Sahasrabudhe. Marginals before conditionals: Staged disambiguation in gradient-trained transformers. *arXiv preprint arXiv:2603.10074*, 2026.

Kushal Tirumala, Aram H. Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. Memorization without overfitting: Analyzing the training dynamics of large language models. In *Advances in Neural Information Processing Systems*, 2022. arXiv:2205.10770.

Aditya Varre, Gizem Yüce, and Nicolas Flammarion. Learning in-context n-grams with transformers: Sub-n-grams are near-stationary points. *arXiv preprint arXiv:2508.12837*, 2025.

Geoffrey Wolfer and Aryeh Kontorovich. Minimax learning of ergodic Markov chains. In *Proceedings of the 30th International Conference on Algorithmic Learning Theory*, volume 98 of *Proceedings of Machine Learning Research*, pp. 904–930. PMLR, 2019. arXiv:1809.05014.

Yuan Yao, Lorenzo Rosasco, and Andrea Caponnetto. On early stopping in gradient descent learning. *Constructive approximation*, 26(2):289–315, 2007.

A Training Dynamics

We train a standard GPT-2 model (Radford et al., 2019), implemented following nanoGPT (Karpathy, 2022), using the AdamW optimizer. The models are trained for next-token prediction on an ensemble of N independently generated sequences, each of length $n = 129$ tokens, processed in a context window of 128 so that each sequence contributes 128 next-token prediction targets. All models are trained with AdamW (Loshchilov & Hutter, 2019) (learning rate 10^{-3}, $\beta_1 = 0.9$, $\beta_2 = 0.99$, weight decay 0.1, gradient clipping at norm 1.0), batch size 64 sequences, no warmup. For the timescale grid the learning rate is held constant over the full run rather than decayed, so that onset times are comparable across cells and not confounded with a schedule. Weight decay is applied to the weight matrices only, following standard practice.

B Proofs for Section 3

Proof of Lemma 1. Written in terms of the counts, the training objective is

$$\mathcal{L}_{\text{training}}(f) \;=\; -\frac{1}{M}\sum_{a}\sum_{b} \#(a,b)\,\log f(b \mid a), \tag{15}$$

which decouples across states a. Dividing the a-th block by $\#(a)$ turns it into the cross-entropy between $\hat{p}(\cdot \mid a)$ and $f(\cdot \mid a)$,

$$\frac{1}{\#(a)}\sum_{b}\#(a,b)\big(-\log f(b \mid a)\big) \;=\; H\big(\hat{p}(\cdot \mid a)\big) + \mathrm{KL}\big(\hat{p}(\cdot \mid a)\,\big\|\, f(\cdot \mid a)\big), \tag{16}$$

whose unique minimizer over the simplex is $f(\cdot \mid a) = \hat{p}(\cdot \mid a)$ by Gibbs' inequality. Minimizing each block independently is legitimate because $f \in \mathcal{P}_1$ imposes no coupling between distinct states. □

Proof of Theorem 1. We first show, for an arbitrary $f \in \mathcal{P}$ satisfying the support condition, the per-history identity

$$\mathbb{E}\left[\mathcal{L}_{\text{training}}(f)\right] = H^\star + \frac{1}{n-1}\sum_{i=2}^{n}\mathbb{E}_{x_{1:i-1}}\mathrm{KL}\big(p(\cdot \mid x_{i-1})\,\big\|\, f(\cdot \mid x_{i-1},\dots,x_1)\big). \tag{17}$$

Fix a position i and condition on the full history $x_{1:i-1}$. By the Markov property and time-homogeneity, $\mathrm{Law}(x_i \mid x_{1:i-1}) = p(\cdot \mid x_{i-1})$ irrespective of history beyond x_{i-1} and of how the chain was started; this uses the Markov property of the data-generating process p and no property of f. By the tower property and the pointwise cross-entropy identity $\mathbb{E}_{b\sim q}[-\log g(b)] = H(q)+\mathrm{KL}(q\|g)$ for $\mathrm{supp}(q) \subseteq \mathrm{supp}(g)$, applied with $q = p(\cdot \mid x_{i-1})$ and $g = f(\cdot \mid x_{i-1},\dots,x_1)$,

$$\begin{aligned}\mathbb{E}_{x\sim p}\big[-\log f(x_i \mid x_{i-1},\dots,x_1)\big] \;=\; \mathbb{E}_{x_{1:i-1}}\Big[&H\big(p(\cdot \mid x_{i-1})\big)\\ &+\mathrm{KL}\big(p(\cdot \mid x_{i-1})\,\big\|\, f(\cdot \mid x_{i-1},\dots,x_1)\big)\Big].\end{aligned} \tag{18}$$

The entropy term depends on $x_{1:i-1}$ only through x_{i-1}, so its expectation is $\sum_a \mu_{i-1}(a)H(p(\cdot \mid a))$ with $\mu_{i-1}(a) := \Pr[x_{i-1} = a]$; averaging over $i = 2,\dots,n$ replaces μ_{i-1} by the occupancy measure $\bar{\mu}(a) := \frac{1}{n-1}\sum_{i=1}^{n-1}\Pr[x_i = a]$ and gives $H^\star$. Averaging the KL term over i in the same way gives equation 17; no property of f beyond the support condition has been used.

Now let $f \in \mathcal{P}_1$, so $f(\cdot \mid x_{i-1},\dots,x_1) = f(\cdot \mid x_{i-1})$. The KL summand in equation 17 is then a function of x_{i-1} alone, so

$$\mathbb{E}_{x_{1:i-1}}\big[\mathrm{KL}\big(p(\cdot \mid x_{i-1})\,\big\|\, f(\cdot \mid x_{i-1})\big)\big] \;=\; \sum_{a}\mu_{i-1}(a)\,\mathrm{KL}\big(p(\cdot \mid a)\,\big\|\, f(\cdot \mid a)\big), \tag{19}$$

and averaging over $i = 2,\dots,n$ again replaces μ_{i-1} by $\bar{\mu}$, giving equation 12. This is the only place the restriction $f \in \mathcal{P}_1$ is used: it is what collapses the history-dependent divergence in (i) onto a single state, re-weighted by the occupancy measure. (When the chain is started from its invariant measure, $\mu_{i-1} = \bar{\mu} = \pi$ at every position, but this is not assumed.)

The first term is independent of f. For the second, Gibbs' inequality gives $\mathrm{KL} \geq 0$ with equality if and only if the two conditionals agree: by Jensen, $-\sum_b p \log(f/p) \geq -\log \sum_b p\,(f/p) = -\log \sum_b f = 0$, and strict concavity of $\log$ forces f/p to be constant, hence equal to 1 since both sum to one. Because the conditionals $f(\cdot \mid a)$ for distinct a are unconstrained relative to one another, every divergence term is minimized simultaneously at $f(\cdot \mid a) = p(\cdot \mid a)$, and this holds for any nonnegative weights: the weights fix the value of the constant, not the location of the minimum, and matter only in that a state of zero occupancy leaves $f(\cdot \mid a)$ undetermined. □

Lemma 2 (reverse Pinsker)**.** *For distributions P, Q on $\mathcal{V}$ with Q everywhere positive,*

$$\mathrm{KL}(P \,\|\, Q) \;\leq\; \chi^2(P \,\|\, Q) \;=\; \sum_b \frac{(P(b) - Q(b))^2}{Q(b)} \;\leq\; \frac{\|P - Q\|_1^2}{\min_b Q(b)}. \tag{20}$$

Proof. From $\log x \leq x - 1$, $\mathrm{KL}(P\|Q) \leq \sum_b P\big(\frac{P}{Q} - 1\big) = \sum_b \frac{P^2}{Q} - 1 = \chi^2(P\|Q)$. For the second inequality, bound $1/Q(b)$ by $1/\min_b Q(b)$ and use $\sum_b v_b^2 \leq \big(\sum_b |v_b|\big)^2$. □

Proof of Theorem 2. Conditionally on $\#(a)$, the counts $\#(a, \cdot)$ are multinomial with parameter $p(\cdot \mid a)$; this uses the Markov property alone and so does not depend on how the chain was started. A multiplicative Chernoff bound gives $\Pr[\hat{p}(b \mid a) < p(b \mid a)/2] \leq \exp(-\#(a)\, p(b \mid a)/8)$; a union bound over $b \in \mathcal{V}$, together with the assumed lower bound on $\#(a)$, places us with probability at least $1 - \delta$ on the event $\min_b \hat{p}(b \mid a) \geq p_{\min}/2$. On that event Lemma 2 gives

$$\mathrm{KL}\big(p(\cdot \mid a) \,\big\|\, \hat{p}(\cdot \mid a)\big) \;\leq\; \frac{2}{p_{\min}} \big\|\hat{p}(\cdot \mid a) - p(\cdot \mid a)\big\|_1^2. \tag{21}$$

By Cauchy–Schwarz and the multinomial variance,

$$\mathbb{E}\big\|\hat{p}(\cdot \mid a) - p(\cdot \mid a)\big\|_1^2 \;\leq\; |\mathcal{V}|\, \mathbb{E}\big\|\hat{p}(\cdot \mid a) - p(\cdot \mid a)\big\|_2^2 \;=\; \frac{|\mathcal{V}|\big(1 - \sum_b p(b \mid a)^2\big)}{\#(a)} \;\leq\; \frac{|\mathcal{V}| - 1}{\#(a)}. \tag{22}$$

The expected number of positions at which a is the conditioning token is $\mathbb{E}[\#(a)] = N \sum_{i=1}^{n-1} \Pr[x_i = a] = M\, \bar{\mu}(a)$, by the definition of $\bar{\mu}$ in equation 11 and for any initial distribution of the chain. Averaging over a with $\#(a) \approx M\bar{\mu}(a)$,

$$\sum_a \bar{\mu}(a)\, \frac{2\big(|\mathcal{V}| - 1\big)}{p_{\min}\, M \bar{\mu}(a)} \;=\; \frac{2\,|\mathcal{V}|\big(|\mathcal{V}| - 1\big)}{p_{\min}\, M}. \qquad \square$$

Remark 1 (The $p_{\min}$ factor is a direction-of-divergence artifact). Measured the other way, Wilks' theorem gives $\mathbb{E}_\pi \mathrm{KL}(f^\star \,\|\, p) \approx \frac{|\mathcal{V}|(|\mathcal{V}|-1)}{2M}$ with no dependence on $p_{\min}$; it enters Theorem 2 only because the true kernel is the reference, the convention also used for ϵ_t in equation 1. A $1/p_{\min}$-type factor ($1/\pi_{\min}$ there) also appears in the sharp minimax rate for Markov chain estimation under total-variation and operator-norm losses (Wolfer & Kontorovich, 2019): the dependence is a feature of estimating rare transitions under a reference-asymmetric metric, not an artefact of this proof.

Remark 2 (TV and KL are not interchangeable). A fixed threshold does not carry the same meaning under the two metrics, and since $p_{\min}$ varies with the kernel draw, not across chains either — which matters once τ_{gen} is compared across orders. The factor is not small: for rows drawn from $\mathrm{Dirichlet}(0.5)$ with $|\mathcal{V}| = 3$, the median of $1/p_{\min}$ is roughly 190, estimated over 2×10^5 random kernels. We report KL throughout, the metric already fixed by equation 12.

Remark 3 (A caution for the experiments). Roughly 12% of $\mathrm{Dirichlet}(0.5)$ kernels at $M = 25{,}600$ admit a pair (a, b) with expected count below 1. If unobserved, $\hat{p}(b \mid a) = 0$ while $p(b \mid a) > 0$ and $\mathrm{KL}(p \,\|\, f^\star) = \infty$: the bound of Theorem 2 then fails outright rather than degrading. Either $f^\star$ should be Laplace-smoothed, or a larger Dirichlet parameter used for the theoretical statement.

B.1 Details of the validation experiment

Setup: The run of Figure 2 trains the model of Section 3 on an order-1 chain.

Top panel: $C_t = \min_{f \in \mathcal{P}_1} \mathbb{E}\,\mathrm{KL}(p_{\theta_t}(\cdot \mid x_{i-1}, \ldots, x_1) \,\|\, f(\cdot \mid x_{i-1}))$ is estimated from 64 sequences. The inner minimizer is attained in closed form at the conditional average of the model's predictions over contexts sharing the last token, so C_t requires forward passes only. Measured on training and held-out contexts separately, the two curves are indistinguishable throughout ($C_0 \approx 6 \times 10^{-3}$, in-class floor 10^{-4}, 0.35 at the end of training).

Middle panel: The dashed line is the in-class minimum $\hat{H} + \sum_a \hat{\pi}(a)\,\mathrm{KL}(\hat{p}(\cdot \mid a) \,\|\, f(\cdot \mid a))$ evaluated at the model's projection onto $\mathcal{P}_1$, with $\hat{p}$, $\hat{\pi}$, $\hat{H} = 0.7119$ the empirical kernel, its context marginal, and the empirical conditional entropy, all computed once from the corpus. Training loss matches it to within 7×10^{-3} nats at initialization and 2×10^{-3} nats through the generalization phase, and ends 0.36 nats below it.

Bottom panel: The finite-sample bound of Theorem 2 evaluates to $2\,|\mathcal{V}|(|\mathcal{V}| - 1)/(p_{\min} M) = 2.5 \times 10^{-3}$ nats for this chain ($p_{\min} = 0.0048$). The measured estimation floor is $\mathrm{KL}(p \,\|\, \hat{p}) = 5.7 \times 10^{-6}$ nats. The model's held-out divergence to the kernel reaches 1.6×10^{-4} nats at its minimum, below the theorem's guarantee and a factor ~ 30 above the floor.

B.2 Computing the distance to the low-order classes

Let P_j denote the class of order-j predictors: maps $f\colon \mathcal{V}^j \to \Delta(\mathcal{V})$ whose prediction depends on the context only through its last j tokens. Although P_j is a function class, it is finite-dimensional—a stochastic table with V^j rows and $V^j(V-1)$ free parameters—and the distance from the model to it admits a closed form; no iterative projection is required.

Fix a distribution μ over probe contexts (in practice the empirical distribution over a fixed probe set, described below), write $p_x = p_\theta(\cdot \mid x)$ for the model's next-token distribution in context x, and let $s_j(x)$ be the last-j suffix of x. Define

$$C_j \;=\; \min_{f \in P_j} \; \mathbb{E}_{x \sim \mu}\, \mathrm{KL}\big(p_x \,\big\|\, f(\cdot \mid s_j(x))\big). \tag{23}$$

Closed form: The objective in equation 23 decomposes over suffix values: for each $s \in \mathcal{V}^j$, the row $f(\cdot \mid s)$ appears only in the terms with $s_j(x) = s$, and $\mathbb{E}\big[\mathrm{KL}(p_x \,\|\, q) \mid s_j(x) = s\big]$ is minimized over q by the conditional average $\bar{p}_s = \mathbb{E}\big[p_x \mid s_j(x) = s\big]$ (the cross-entropy term is linear in q, so the minimizer of $-\mathbb{E}[\sum_y p_x(y) \log q(y)]$ on the simplex is the mean). Substituting the minimizer gives

$$C_j \;=\; \mathbb{E}_x\, \mathrm{KL}\big(p_x \,\big\|\, \bar{p}_{s_j(x)}\big) \;=\; \mathbb{E}\, H\big(\bar{p}_{s_j(x)}\big) \;-\; \mathbb{E}\, H(p_x), \tag{24}$$

the mean entropy of the suffix-group averages minus the mean entropy of the model's predictions.

Distance equals sensitivity: Let $X \sim \mu$ and let $Y \sim p_\theta(\cdot \mid X)$ be a token sampled from the model. Since $s_j(X)$ is a function of X, equation equation 24 is precisely the conditional mutual information

$$C_j \;=\; I\big(Y;\, X \,\big|\, s_j(X)\big)\,: \tag{25}$$

the predictive information the model extracts from the context beyond its last j tokens. "Distance to P_j" and "sensitivity to tokens deeper than j" are therefore the same quantity. The chain rule refines this to a per-position statement: the increment $C_j - C_{j+1} = I\big(Y;\, X_{-(j+1)} \,\big|\, s_j(X)\big)$ is the information carried by the $(j{+}1)$-th token from the end, given the j tokens after it. Because the classes are nested ($P_0 \subset P_1 \subset \cdots$), C_j is nonincreasing in j, and $C_j = 0$ if and only if the model is (μ-a.s.) an order-j predictor. Note that C_j references only the model's own predictions—not the true kernel—so it is computable on any data source, including the natural-text corpora, and is distinct from the information floors Δ_j, which measure the distance from the *true kernel* to its order-j projection.

Estimator: We evaluate C_j on a fixed probe set: B sequences with predictions taken at positions $i = k, \ldots, T-1$, giving $M = B\,(T-k)$ context–prediction pairs (we use training contexts; Section 3 shows held-out probes give matching values). One forward pass yields all p_x; grouping the pairs by last-j suffix and applying equation 24 under the empirical measure gives the plug-in

estimate for every j simultaneously. Suffix groups containing a single probe context contribute exactly zero to the sum—the group mean *is* the prediction—but are retained in the normalization M; dropping them would rescale the objective and could violate the monotonicity $C_{j+1} \leq C_j$. The practical limit of the instrument is resolution, not computation: as j grows, the M probe contexts spread over V^j suffix groups, group sizes shrink toward one, and the plug-in estimate is biased toward zero. We therefore report C_j only for depths at which the probe set still populates the suffix groups (the "deepest estimable class"), and scale the probe set with the depths of interest.

C GENERATIVE AI STATEMENT

We used LLM-based tools to edit the manuscript, to assist in proving Theorem 2, and to identify related areas of work that help contextualize our contributions. The authors verified all technical content, including the proofs, and developed the experimental design and analysis. The authors take full responsibility for the paper's contents.