PROMISES AND PITFALLS OF GENERATIVE MASKED LANGUAGE MODELING: THEORETICAL FRAMEWORK AND PRACTICAL GUIDELINES

Yuchen Li[1,2] **Alexandre Kirchmeyer**[1*] **Aashay Mehta**[1*] **Yilong Qin**[1*]

Boris Dadachev[2*] **Kishore Papineni**[2*] **Sanjiv Kumar**[2] **Andrej Risteski**[1]
[1]Carnegie Mellon University [2]Google Research
yuchenl4@cs.cmu.edu

ABSTRACT

Autoregressive language models are the currently dominant paradigm for text generation, but they have some fundamental limitations that cannot be remedied by scale—for example inherently sequential and unidirectional generation. While alternate classes of models have been explored, we have limited mathematical understanding of their fundamental power and limitations. In this paper we focus on *Generative Masked Language Models (GMLMs)*, a non-autoregressive paradigm in which we train a model to fit conditional probabilities of the data distribution via masking, which are subsequently used as inputs to a Markov Chain to draw samples from the model. These models empirically strike a promising speed-quality trade-off as each step can be typically parallelized by decoding the entire sequence in parallel. We develop a mathematical framework for analyzing and improving such models which sheds light on questions of sample complexity and inference speed and quality. Empirically, we adapt the T5 model for iteratively-refined parallel decoding, achieving 2-3x speedup in machine translation with minimal sacrifice in quality compared with autoregressive models. We run careful ablation experiments to give recommendations on key design choices, and make fine-grained observations on the common error modes in connection with our theory. Our mathematical analyses and empirical observations characterize both potentials and limitations of this approach, and can be applied to future works on improving understanding and performance of GMLMs. [1]

1 INTRODUCTION

The current dominant approach to language modeling is *autoregressive* (AR): to generate a sequence of tokens, the language model starts by predicting the leftmost token, and then proceeds from left to right, each step predicting the next token based on everything on its left (Raffel et al., 2020; Brown et al., 2020; Touvron et al., 2023). AR models are not without limitations: (1) *Lack of parallelism:* To generate a sequence of N tokens, AR language models need N sequential decoding steps. Each step consists of a forward pass of the decoder component. When N is large, N sequential decoding steps incur high latency. (2) *Quality:* When predicting each token, the model cannot access its right hand side context, and has no natural way to revise earlier predictions on the left. This intuitive limitation was more formally explored in prior theoretical works (Li & Risteski, 2021; Lin et al., 2021).

One promising alternative is based on *Generative Masked Language Models (GMLMs)*. They are trained to fit conditional probabilities for parts of the sequence (by applying a mask), conditioned on the rest. To produce samples, these conditionals are used as oracles for running Markov Chain, e.g. a Gibbs sampler (Wang & Cho, 2019; Goyal et al., 2022). Alternatively, we can think of these steps as an iterative refinement process, typically starting with pure noise (i.e. all tokens are masked or randomized). One can even fit conditional probabilities for noised versions of the input distribution, and use them as inputs to a denoiser to get certain types of discrete diffusion models (Austin et al., 2021). In GMLMs, typically one step of the Markov Chain is operationalized by a Transformer that generates the sequence in parallel (i.e. *parallel decoding* (Ghazvininejad et al., 2019; Gu & Kong, 2021; Savinov et al., 2022)). Hence, if the total number of steps is small, the latency is low. We discuss other related works in Appendix C.

*Equal theoretical / empirical contribution.

[1]Preliminary draft. Updates will be posted at cs.cmu.edu/~yuchenl4

However, none of these approaches robustly surpass autoregressive models in both speed and quality for a wider range of language generation tasks beyond machine translation. Thus, the following questions naturally arise: (Q1) GMLMs are trained to learn conditional probabilities. When does it also imply learning the *joint* probability? (Q2) What properties of the data distribution and training/inference algorithm govern the quality of the learned model and its generated samples? (Q3) What are the best practices for training GMLMs, and can we use theory to elucidate the design space of losses, training and inference procedures?

Our contributions. Towards answering the questions above, we introduce a *theoretical framework* to characterize the potentials and limitations of GMLMs, for both training and inference. Precisely,

- The **asymptotic sample complexity** for estimating the parameters of a distribution via a broad class of masked-prediction losses can be related to the mixing time of a corresponding Markov Chain that can be used to sample from the distribution (Section 2.2). Furthermore, we prove that training with larger masks always improves statistical efficiency (Theorem 1).
- We show **finite-sample bounds** that translate bounds on how closely the *conditional* distributions of the data distribution are learned, to how well the *joint* distribution is learned (Section 2.3) if we have some capacity control over the distribution class being learned (e.g. covering number bounds).
- **Transformers** for parallel decoding has certain limitations, preventing it from **efficiently sampling** even simple distributions with strong correlations between the coordinates (Section 3).

We accompany these theoretical findings with an extensive set of empirical investigations detailing important components and common error modes. Precisely:

- Our experiments (Section 4) suggest the **empirically critical components** include large masking ratio (c.f. theory in Section 2.2), custom vocabulary, distillation from AR models, and architecture improvements like positional attention. Related findings exist in prior works (Appendix C).
- GMLMs with parallel-decoding work well on **machine translation**: in fact, even *one single* forward pass can often produce reasonable translations. This aligns with our theoretical framework, as machine translation tasks typically involve unimodal, lower-entropy outputs.
- Common **error modes** ("stuttering") suggest limitations for parallel-decoding GMLMs for modeling strong dependencies (c.f. theory in Section 3), which we empirically quantify (Section 4.3).

Jointly, our theoretical and empirical findings suggest synergistically designing better Markov Chains that mix fast in the presence of strong correlations in the target, and corresponding losses that inherit good statistical behavior.

2 Theoretical framework

We develop a mathematical framework for reasoning about the core ingredients for successfully training and using GMLMs: the *statistical complexity* to learn the model, and the *speed of inference*. We show that these two are surprisingly closely related: namely, we understand both the asymptotic and finite-sample statistical complexity through *functional inequalities* (e.g. Poincaré, approximate tensorization of entropy) corresponding to the Markov Chains we would use at inference time—which in turn characterize the mixing time of these chains. This picture closely mirrors an emerging picture in the continuous case for score-based (diffusion) models Koehler et al. (2023); Qin & Risteski (2023)—though with fairly different proof techniques.

2.1 Asymptotic sample efficiency via functional inequalities

The most classical way of fitting distributions from data is maximum likelihood, which is asymptotically the most sample-efficient (Hájek, 1972) . However, optimizing maximum likelihood is computationally challenging for many families of distributions. Thus, alternate strategies and losses to fit the parameters have been developed.

For continuous distributions, a common choice is the *score matching* loss. For discrete distributions, a closely related strategy is learning the conditionals of subsets of variables. Operationalizing this as a loss gives us the *pseudolikelihood* loss (Besag, 1975). More recently, this strategy has been used in conjuction with neural models to both learn useful features in the guise of masked language modeling (MLM) (Devlin et al., 2019), which can be also used to produce a generative model (Wang & Cho, 2019). The latter is done by using the learned conditionals inside a Gibbs sampler. However, when the conditionals are not *consistent*, i.e. there is not a joint distribution that satisfies these conditionals, Gibbs sampler may amplify errors. In general, mathematical understanding about sampling from masked language models is still lagging substantially behind.

Setup: Let Ω be a finite discrete set. Let p denote a distribution over a sequence of N variables $X = X_1 \cdots X_N \in \Omega^N$. [2] We consider learning parameters θ parametrizing some distribution p_θ, for $\theta \in \Theta$. The classical way of fitting θ is to maximize the likelihood of the training data:

Definition 1 (MLE, Van der Vaart (2000))**.** *Given i.i.d. samples $x_1, \ldots, x_n \sim p_\theta$, the max likelihood estimator is $\hat{\theta}_{MLE} = \arg\max_{\theta' \in \Theta} \hat{\mathbb{E}}\left[\log p_{\theta'}(X)\right]$, where $\hat{\mathbb{E}}$ denotes the expectation over the samples. As $n \to \infty$ and under regularity conditions, we have $\sqrt{n}\left(\hat{\theta}_{MLE} - \theta\right) \to N\left(0, \Gamma_{MLE}\right)$, where $\Gamma_{MLE} := \mathcal{I}^{-1}$, $\mathcal{I}$ is the Fisher information matrix.*

A classical result due to Hájek-Le Cam (for modern exposition see Van der Vaart (2000)) is that maximum likelihood is the asymptotically most sample-efficient estimator among all "sufficiently regular" estimators (Section 8.5 in Van der Vaart (2000)) — so we will treat it as the "gold standard" against which we will compare other estimators. The class of estimators we will be focusing most is the a broad generalization of the *pseudo-likelihood estimator* (Besag, 1975).

For any $K \subset [N]$ denoting the set of masked positions in $X \in \Omega^N$, let $p(X_K|X_{-K})$ denote the conditional probability of the subsequence $(X_i \mid i \in K)$ given all other variables $(X_i \mid i \notin K)$. [3]

Definition 2 (α-weighted pseudolikelihood)**.** *Denote a collection of sets $\mathcal{K} := \{K_1, \ldots, K_{|\mathcal{K}|}\}$ such that $\cup_i K_i = [N]$, corresponding to probabilities $\alpha := \{\alpha_1, \ldots, \alpha_{|\mathcal{K}|}\}$. Given iid samples of sequences $\mathcal{S}_\mathcal{X} := \{X^{(i)} | X^{(i)} \sim p\}$, suppose each $X^{(i)}$ is assigned a sequence of $|\mathcal{S}_\mathcal{K}|$ mask configurations $\mathcal{S}_\mathcal{K}{}^{(i)} := (K_1^{(i)} \ldots K_{|\mathcal{S}_\mathcal{K}|}^{(i)})$ in which $K_j^{(i)}$ is sampled iid from $\mathcal{K}$ according to α.* [4] *Then, the* α-weighted maximum pseudolikelihood estimator *(MPLE) is $\hat{\theta}_{PL} := \arg\min_\theta L_{PL}(\theta; \mathcal{S}_\mathcal{X}, \mathcal{S}_\mathcal{K})$ where $L_{PL}(\theta; \mathcal{S}_\mathcal{X}, \mathcal{S}_\mathcal{K}) := \sum_{i=1}^{|\mathcal{S}_\mathcal{X}|} \sum_{j=1}^{|\mathcal{S}_\mathcal{K}|} l_{PL}(\theta; X^{(i)}, K_j^{(i)})$, $l_{PL}(\theta; X, K) := -\log p_\theta(X_K|X_{-K})$. The population loss is* [5] *$L_{PL}(\theta) := \mathbb{E}_{X\sim p, K\sim\alpha}\left[l_{PL}(\theta; X, K)\right]$. Let $\tilde{p}$ denote the (noisy) observed counterparts of p, we also consider $\tilde{L}_{PL}(\theta) := \mathbb{E}_{X\sim\tilde{p}, K\sim\alpha}\left[l_{PL}(\theta; X, K)\right]$*

As a special case, if $\mathcal{K}$ contains all subsets of a certain size k, with uniform weights, and $|\mathcal{S}_\mathcal{K}| = 1$, we get the classical k-pseudolikelihood estimator: In fact, for Ising models, the corresponding loss is even convex (Appendix A.1).

Definition 3 (k-pseudolikelihood (Huang & Ogata, 2002))**.** *Same as Definition 2 except that $\mathcal{K} := \{K \subseteq [N] \mid |K| = k\}$, $\alpha = Unif(\mathcal{K})$, and $|\mathcal{S}_\mathcal{K}| = 1$.*

Informally, we predict the variables in positions $K \in \mathcal{K}$, conditioned on the remaining variables. The benefit is that parametrizing conditionals over smaller subsets K is often computationally cheaper. For instance, if $p_\theta(x)$ is an undirected graphical model, i.e. $p_\theta(x) \propto \exp(\sum_C \phi_{C,\theta}(x_C))$, where the sum is over all maximal cliques C of the graph describing the distribution, the conditional distribution of K only depends on its Markov blanket, which can be very small for sparse graphs and small sets K. Thus, computing the partition function corresponding to $p(x_K|x_{-K})$ takes time exponential in this Markov blanket. By contrast, computing the likelihood requires calculating the partition function of $p_\theta(x)$, which takes time exponential in the dimension of X. In fact, for Ising models, the corresponding loss is even *convex* (Appendix A.1). A similar tradeoff exists for masked language models: fitting the conditionals for larger masks would likely require a larger model, thus would be computationally more expensive.

2.2 Sample complexity via mixing time bounds

We provide a framework for bounding the asymptotic sample complexity of learning the parameters θ of a discrete probability distribution with a broad family of "masked prediction" objectives. We will measure the quality of an estimator in terms of *parameter recovery*. To make this formal, we first recall that under mild technical conditions, the estimator will be asymptotically normal:

[2] In language models, Ω is the set of tokens in the vocabulary.

[3] $p(X_K|X_{-K})$ is motivated by the masked language modeling objective in Bert (Devlin et al., 2019).

[4] The set of mask configurations $\mathcal{S}_\mathcal{K}{}^{(i)}$ can be different for different $X^{(i)}$, but technically $\mathcal{S}_\mathcal{K}{}^{(i)}$ does not depend on the *content* of $X^{(i)}$. It just depends on the *index* i of $X^{(i)}$ in $\mathcal{S}_\mathcal{X}$. For simplicity, assume that each sample $X^{(i)}$ observes the same number of mask configurations, i.e. the size $|\mathcal{S}_\mathcal{K}|$ is the same constant for all $X^{(i)}$.

[5] This is equivalent to minimizing the KL divergence of the groundtruth conditional distribution $p(X_K|X_{-K})$ from the predicted conditional distribution $p_\theta(X_K|X_{-K})$: $\mathbb{E}_{X\sim p}\left[\mathbb{E}_{K\sim\alpha}\left[D_{\mathrm{KL}}\left(p(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right)\right]\right]$

Lemma 1 (Asymptotic normality (Van der Vaart, 2000))**.** *For the α-weighted MPLE in Definition 2, fix $|\mathcal{S}_{\mathcal{K}}| = 1$, and define $\theta^* \in \arg\min_\theta L_{PL}(\theta)$. Under mild regularity conditions (Lemma 3 in Appendix A.3), as $|\mathcal{S}_{\mathcal{X}}| \to \infty$, $\sqrt{|\mathcal{S}_{\mathcal{X}}|}(\hat{\theta}_{PL} - \theta^*) \xrightarrow{d} \mathcal{N}(0, (\nabla_\theta^2 L_{PL}(\theta^*))^{-1}\mathrm{Cov}(\nabla_\theta l_{PL}(\theta^*))(\nabla_\theta^2 L_{PL}(\theta^*))^{-1})$*

If we know $\sqrt{|\mathcal{S}_{\mathcal{X}}|}(\hat{\theta}_{PL} - \theta^*) \xrightarrow{d} \mathcal{N}(0, \Gamma_{PL})$, we can extract bounds on the expected ℓ_2^2 distance between $\hat{\theta}_n$ and θ^*. Namely, from Markov's inequality, (see e.g., Remark 4 in Koehler et al. (2023)), for sufficiently large $|\mathcal{S}_{\mathcal{X}}|$, with probability at least 0.99 it holds that $\|\hat{\theta}_{PL} - \theta^*\|_2^2 \leq \frac{\mathrm{Tr}(\Gamma_{PL})}{|\mathcal{S}_{\mathcal{X}}|}$.

2.2.1 Masking more is (statistically) better

We prove that increasing the number of variables k we predict in k-pseudolikelihood (Definition 3) strictly improves the statistical efficiency of the resulting estimator. Note, for larger k, we expect the computational cost to optimize the corresponding loss to be larger, and when $k = N$ we just get max likelihood. Thus, this naturally formalizes a computational/statistical tradeoff in choosing k.

Assumption 1. *$\forall \theta \in \Theta, x \in \Omega^N, K \subset [N]$, the norms of the gradient $\|\nabla_\theta \log p_\theta(x_K|x_{-K})\|_2$ and the Hessian $\|\nabla_\theta^2 \log p_\theta(x_K|x_{-K})\|_F$ exist and are finite .*

Theorem 1 (Masking more is (statistically) better)**.** *Under Assumption 1, for every $k \in [N-1]$, let Γ_{PL}^k denote the asymptotic variance of the k-MPLE estimator (Definition 3). We have:*[6] $\Gamma_{PL}^{k+1} \preceq \Gamma_{PL}^k$

Remark 1. *By monotonicity of trace, Thm 1 implies $\mathrm{Tr}(\Gamma_{PL}^{k+1}) \leq \mathrm{Tr}(\Gamma_{PL}^k)$. Thm 1 also implies larger k gives stronger asymptotic l_2 bound for learning θ since $\mathbb{E}_{x_{1:n}, s_{1:n}}\left[\|\hat{\theta}_{PL}^k - \theta\|_2^2\right] \to \frac{\mathrm{Tr}(\Gamma_{PL}^k)}{|\mathcal{S}_{\mathcal{X}}|}$.*

The main lemma for Theorem 1 is that the two matrices in the asymptotic covariance of MPLE, $\nabla_\theta^2 L_{PL}(\theta)$ and $\mathrm{Cov}(\nabla_\theta l_{PL}(\theta))$ are actually equal. For MLE (namely, when $k = N$) this is well-known and called the *information matrix equality*. Proofs of Lemma 2 and Theorem 1 are in Appendix A.2 and Appendix A.4, respectively.

Lemma 2 (Generalized information matrix equality)**.** *Under Assumption 1, the α-weighted pseudolikelihood loss (Definition 2) verifies: $\nabla_\theta^2 L_{PL}(\theta) = \mathrm{Cov}(\nabla_\theta l_{PL}(\theta))$*

As a consequence, the pseudolikelihood estimator $\hat{\theta}_n$ is asymptotically normal with the following asymptotic covariance matrix: $\sqrt{|\mathcal{S}_{\mathcal{X}}|}(\hat{\theta}_{PL} - \theta^*) \to \mathcal{N}(0, \left(\nabla_\theta^2 L_{PL}(\theta^*)\right)^{-1})$

2.2.2 Statistical efficiency bounds via mixing time bounds

We could in general conceive of masking strategies where certain subsets of variables get masked with different probabilities. For instance, in language, nearby words will tend to be more correlated; grammatical constraints will dictate the parts-of-speech that can occur in different positions. We would then like to have theoretical guidance on what choices of masking distributions are better. Remarkably, it turns out that we can relate the statistical efficiency — in the sense of $\mathbb{E}\|\hat{\theta} - \theta^*\|^2$ for the resulting estimator $\hat{\theta}$ — and the mixing time of an appropriately chosen Markov Chain. In fact, this is the Markov Chain that would be typically chosen at inference time. Towards making this formal, we will need several preliminary concepts and results for Markov chains. Recall, a Markov chain on a state space Ω is described by a (row-stochastic) transition matrix P. Moreover, we can assign a natural bilinear form called the Dirichlet form:

Definition 4 (Dirichlet form)**.** *Let M be an ergodic, reversible Markov chain with transition matrix P on state space Ω. Let μ be its unique stationary distribution. $\forall f, g : \Omega \to \mathbb{R}$ the associated Dirichlet form is $\mathcal{E}_P(f, g) = \langle f, (I - P)g\rangle_\mu = \frac{1}{2}\Sigma_{x,y\in\Omega}\mu(x)P(x,y)(f(x) - f(y))(g(x) - g(y))$*

Mixing time of the Markov chain can be bounded in χ^2 sense by the gap between the 1st and 2nd eigenvalue of the Laplacian matrix $I - P$, expressed as Poincaré inequality:

Definition 5 (Poincaré inequality)**.** *We say that a Markov chain satisfies a Poincaré inequality with constant C if for all $f : \Omega \to \mathbb{R}$, we have $\mathcal{E}_P(f, f) \geq \frac{1}{C}\mathrm{Var}_\mu(f)$.*

[6] The notation $A \preceq B$ means $B - A$ is positive semidefinite.

The Poincaré inequality implies exponential ergodicity of the Markov chain in χ^2-divergence, precisely $\chi^2(p_t, \mu) \leq e^{-2t/C}\chi^2(p_0, \mu)$, where μ is the stationary distribution of the chain and p_t is the distribution after running the Markov process for time t, starting at p_0. We will be particularly interested in several generalizations of Gibbs sampling:

Definition 6 (α-weighted block dynamics (Caputo & Parisi (2021))). *Let $\mathcal{K} := \{K_1, \ldots, K_{|\mathcal{K}|}\}$ be a collection of sets (or blocks) such that $\cup_i K_i = [N]$. A block dynamics with blocks $\mathcal{K}$ is a Markov chain that picks a block K in each step according to some distribution* [7] *α, and then updates the configuration in K according to the conditional measure given the configuration in $-K := [N] \setminus K$. The Dirichlet form corresponding to this chain is: $\mathcal{E}_{P_\alpha}(f, g) = \sum_{K \in \mathcal{K}} \alpha(K) \mathbb{E}_{X_{-K}} \left[Cov_{X_K | X_{-K}}(f, g) \right]$*

The crucial result we show is that the statistical efficiency of the α-weighted MPLE (Definition 2) as captured by the asymptotic variance can be related to the Poincaré constant of the corresponding α-weighted Block dynamics (Definition 6). Proof of Theorem 2 is in Appendix A.5.

Theorem 2 (Asymptotic variance under a Poincaré Inequality). *Suppose the distribution p_{θ^*} satisfies a Poincaré inequality with constant C with respect to the α-weighted Block dynamics. Then the asymptotic variance of the α-weighted MPLE can be bounded as: $\Gamma_{PL} \preceq C\mathcal{I}^{-1}$ where $\mathcal{I}$ is the Fisher Information matrix (Definition 1).*

2.3 Finite sample bounds and distributional distance

The framework in Section 2.2 was asymptotic in nature, and used parameter closeness as a notion of "quality" of the estimator. In this section, we remove both requirements, at the cost of the bounds depending on a notion of "complexity" of the parametric class we are fitting. It turns out that we can prove very similar results, with the notion of "mixing" — as captured by the Poincaré constant — being replaced by a different constant called the "approximate tensorization constant". We first introduce several preliminary concepts.

Definition 7 (Approximate tensorization of entropy (Marton, 2013; 2015; Caputo et al., 2015)). *We say the distribution q on Ω^N satisfies* approximate tensorization of entropy *with constant $C_{AT}(q)$ if for any distribution r on Ω^N, it holds that: $D_{\mathrm{KL}}(r, q) \leq C_{AT}(q) \mathbb{E}_{X \sim r} \left[\sum_{i=1}^N D_{\mathrm{KL}} \left(r(\cdot \mid X_{-\{i\}}), q(\cdot \mid X_{-\{i\}}) \right) \right]$*

This inequality is closely related to the mixing time of Glauber dynamics. Namely, the inequality is sandwiched between two discrete versions of the log-Sobolev inequality (Proposition 1.1 of Caputo et al. (2015)): it is weaker than the standard discrete version of the log-Sobolev inequality (Diaconis & Saloff-Coste, 1996) and stronger than the Modified Log-Sobolev Inequality (Bobkov & Tetali, 2006) which implies exponential ergodicity of Glauber dynamics in KL divergence, namely[8] $\mathrm{KL}(p_t, \mu) \leq e^{-2t/C_{AT}(q)} \mathrm{KL}(p_0, \mu)$. This notion can be generalized to conditional distribution of more than one variables:

Definition 8 (Block-generalized approximate tensorization of entropy (Caputo & Parisi, 2021)). *The distribution q over Ω^N and the distribution α over binary masks $\mathcal{K}$ satisfies the* block-generalized *approximate tensorization of entropy with constant $\bar{C}_{AT}(q, \alpha)$ if for any distribution r over Ω^N, $D_{\mathrm{KL}}(r, q) \leq \bar{C}_{AT}(q, \alpha) \cdot \mathbb{E}_{X \sim r} \left[\mathbb{E}_{K \sim \alpha} \left[D_{\mathrm{KL}} \left(r(\cdot \mid X_{-K}), q(\cdot \mid X_{-K}) \right) \right] \right]$*

Similarly as the standard approximate tensoriation constant (Definition 7), it implies exponential ergodicity of the block Gibbs dynamics (Definition 6). Our results will also require two mild assumptions on the distribution we are fitting. First, we assume that when the ground-truth conditional probability is nonzero, the learned conditional probability is uniformly lower-bounded by a constant:

Assumption 2 (Support margin). *There exists constant $\beta \in (0, 1)$ such that $\forall X \sim \tilde{p}, \forall K \subset [N]$ such that $|K| = k$, if $\tilde{p}(X_K | X_{-K}) > 0$, then $p_\theta(X_K | X_{-K}) \geq \beta, \forall \theta \in \Theta$.*

We also assume that the parameter space can be discretized into a finite grid such that: (1) within the same grid cell, the parameters correspond to distributions with similar losses; (2) the cardinality of the grid is small. This is a relatively weak assumption because the population loss $\tilde{L}_{PL}(\theta)$ and the sample loss $L_{PL}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}})$ are bounded within $[0, \ln \frac{1}{\beta}]$ (by Proposition 7 in Appendix A.6).

[7] This is analogous to the training objective setting in Definition 2.

[8] This in turns, also implies a Poincaré inequality and exponential ergodicity in χ^2 divergence.

Assumption 3 (Covering bound and Lipschitzness)**.** *$\forall \epsilon > 0$, there exists a finite partition $Par_\epsilon(\Theta) = \{\Theta_1, \cdots, \Theta_{|Par(\Theta)|}\}$ of Θ, $\forall i, \forall \theta_1, \theta_2 \in \Theta_i$, $\left|\tilde{L}_{PL}(\theta_1) - \tilde{L}_{PL}(\theta_2)\right| \leq \frac{\epsilon}{2}$, $|L_{PL}(\theta_1; \mathcal{S}_\mathcal{X}, \mathcal{S}_\mathcal{K}) - L_{PL}(\theta_2; \mathcal{S}_\mathcal{X}, \mathcal{S}_\mathcal{K})| \leq \frac{\epsilon}{2}$ Moreover, $C_\epsilon(\Theta)$ denote the smallest possible cardinality among such partitions $Par_\epsilon(\Theta)$.*

With this setup, we can prove the following finite-sample bound on the closeness of the learned distribution, provided the α-weighted pseudolikelihood (Definition 2) is small:

Theorem 3 (Generalization bound for learning the joint distribution)**.** *Let $\hat{\theta} \coloneqq \hat{\theta}_{PL}$. Under Assumption 2 and Assumption 3 (in Appendix A.6), $\forall \epsilon > 0$, $\forall \delta \in (0, 1)$, with probability at least $1 - \delta$ over the randomness of $\mathcal{S}_\mathcal{X}$ and $\mathcal{S}_\mathcal{K}$, we have $D_{\mathrm{TV}}\left(p_{\hat{\theta}}, p\right) < \sqrt{\frac{1}{2}\bar{C}_{AT}(p_{\hat{\theta}})\left(A + B \cdot \ln \frac{1}{\beta} + \epsilon\right)} + C$ where $A = L_{PL}(\hat{\theta}; \mathcal{S}_\mathcal{X}, \mathcal{S}_\mathcal{K})$, $B = \sqrt{\frac{2^{3N} C_\epsilon(\Theta)}{|\mathcal{S}_\mathcal{K}| \cdot \delta}} + \sqrt{\frac{\ln \frac{8 C_\epsilon(\Theta)}{\delta}}{2|\mathcal{S}_\mathcal{X}|}}$, and $C = \sqrt{\frac{|\Omega|^{3N}}{8\delta|\mathcal{S}_\mathcal{X}|}}$.*

Proof of Theorem 3 is in Appendix A.6. We can compare the statement to Theorem 2: (1) On the LHS, rather than parameter distance, we have total variation distance between the learned distribution and p. (2) On the RHS, rather than a Poincaré inequality, we have the $\bar{C}_{AT}(p_{\hat{\theta}})$ constant. (3) On the RHS, instead of the Fisher information matrix, we have quantities capturing the generalization error, through a notion of complexity of the class ($C_\epsilon(\Theta)$).

3 Sampling efficiency via Gibbs-like algorithms

In this section, we focus on inference, and the quality and limitations of different sampling procedures. In particular, we focus on Gibbs-like algorithms, implemented by Transformer-based architectures, and derive fine-grained differences between several natural variants we consider. We consider the following variants of per-step update rules:

1. **k-Gibbs sampler.** Definition 6 when $\mathcal{K} \coloneqq \{K \subseteq [N] \mid |K| = k\}$, and $\alpha = \mathrm{Unif}(\mathcal{K})$.

$$X_K^{(t+1)} \sim p(\cdot \mid X_{-K}^{(t)}),\ X_j^{(t+1)} = X_j^{(t)}\ \forall j \notin K \tag{1}$$

2. **Independent parallel.** Perform *coordinate-wise* for all i in parallel, to speed up the process. [9]

$$\forall i \in [N],\ X_i^{(t+1)} \sim p(\cdot \mid X_{-\{i\}}^{(t)}) \tag{2}$$

Among existing language generation approaches via iterative refinement, Wang & Cho (2019) uses 1-Gibbs sampler, Ghazvininejad et al. (2019) is similar to performing k-Gibbs sampler for a predicted subset of indices K. [10] Savinov et al. (2022) and our experiments are similar to running N-Gibbs sampler (see Remark 6 in Appendix A.11 for more details). However, all of these methods rely on the learned parameterized conditional distributions $\hat{p}$, which is different from the groundtruth distribution p, due to limitations in model expressivity and optimization process. Moreover, empirically, $\hat{p}$ may not admit a consistent joint distribution (Young & You, 2022; Torroba Hennigen & Kim, 2023). To formally reason about iterative refinement, we will relax some of these limitations to focus on several underlying theoretical obstacles that these methods face.

3.1 Can Transformers implement Markov chains via parallel decoding?

In this section, we characterize the power and restrictions of Transformers at inference time, and in particular when they are restricted to decoding the tokens of the sequence in parallel. The inference algorithms for a model that has access to approximate conditional probabilities typically look like steps of a Gibbs sampler (e.g. Definition 6). More generally, we can consider inference algorithms that perform several steps of a Markov Chain of our choosing. Note that while there are well-known prior results about the expressive power of Transformers as sequence-to-sequence modelers (Yun

[9] The stationary distribution of this chain is unclear: in fact, it is not even clear the chain is ergodic.

[10] Subject to implementation details: for example, if attention masks are added to prevent any masked position from receiving attention.

et al., 2020), representing steps of a Markov Chain with parallel decoding is more subtle, due to the fact that a step of a Markov Chain requires randomness. First, we state a result characterizing the power of Transformers to approximate "deterministic" Markov Chains: that is, Markov Chains whose transition distributions are delta functions. Unsurprisingly, standard universal approximation results apply to understand such Markov Chains. We show:

Proposition 1 (informal)**.** *Transformers (with sufficient depth and width) can implement any number of transitions of any deterministic Markov Chain over sequences in* Ω^N.

On the other hand, Transformers using parallel decoding cannot implement general Markov chains over Ω^N. In fact, they can only implement Markov Chains for which the transition probabilities are product distributions:

Proposition 2 (informal)**.** *The class of Markov chains over sequences in* Ω^N *implementable by (sufficiently wide and deep) Transformers is those whose next-state transition probability distributions are product distributions over the positions, conditioned on the current state.*

For readability, we defer background information on the Transformer architecture as well as further explanations of Proposition 1 and Proposition 2 to Appendix A.11. Note that this does *not* mean one can only simulate Markov Chains whose *stationary* distribution is a product distribution. In fact, the standard 1-Gibbs sampler, by virtue of the fact that it only updates one coordinate at a time, encodes a product distribution for each transition. On the other hand, under fairly mild conditions on a joint p, the 1-Gibbs sampler corresponding to p is ergodic and has p as a stationary distribution. On the other hand, a step of a k-Gibbs sampler for $k > 1$ is in general not a product distribution, and will not be implementable by a Transformer with parallel decoding.

3.2 Accurately approximating conditionals can be (much) better

Next, we show that being able to take Markov Chain steps that depend on conditionals of (large) sets of coordinates can result in Markov Chains that reach the mode of the distribution much faster. Intuitively, in cases where there is a strong dependence between subsets of variables, jointly updating them will bring us much faster to their modes.

The toy probabilistic family to elicit this phenomenon will be Ising models. Specifically, we consider an undirected graphical model G that can be represented by the union of a clique C_G (in which $|C_G| \geq 2$, and the dependency among variables is strong) and a set of $N - |C_G|$ independent vertices. More formally, we consider: $p_G : \{\pm 1\}^N \to \mathbb{R}^+$,

$$p_G(\mathbf{x}) = \frac{1}{Z_G} \exp\left(\sum_{i \in [N]} \boldsymbol{h}_i \mathbf{x}_i + \sum_{i \neq j \in C_G} J \mathbf{x}_i \mathbf{x}_j \right) \tag{3}$$

in which Z_G is the partition function, and $\boldsymbol{h}_i \in \mathbb{R}$ s.t. $\sum_{i \in C_G} \boldsymbol{h}_i > 0$ and $J > 0$ are scalar constants. This is a *ferromagnetic* Ising model (i.e. the pairwise interactions prefer the variables to have the same value), and when $J \gg \|\boldsymbol{h}\|_1$, the two "modes" of the distribution p_G are such that all variables have the same value:

$$\mathcal{R}_1 \coloneqq \{\boldsymbol{X} \in \{-1, 1\}^N | X_i = 1 \, \forall i \in C_G\} \tag{4}$$

$$\mathcal{R}_{-1} \coloneqq \{\boldsymbol{X} \in \{-1, 1\}^N | X_i = -1 \, \forall i \in C_G\} \tag{5}$$

The above distribution can be seen as a simple prototype of language tasks in which grammatical rules or semantic constraints create "clusters" of positions in which changing isolated words leads to very unlikely sentences. Next, we formalize the concentration around the "modes":

Assumption 4 (Strongly ferromagnetic Ising model)**.** *There exist constants* $h_G > 0, J_0 > 0$ *such that* $h_G \coloneqq \sum_{i \in C_G} \boldsymbol{h}_i > \sum_{i \notin C_G} |\boldsymbol{h}_i|$, $J - \|\boldsymbol{h}\|_1 \geq J_0$.

Informally, under Assumption 4, sequences in $\mathcal{R}_1$ are much more likely under the groundtruth distribution than those in $\mathcal{R}_{-1}$, which are further much more likely than all other sequences. The formal statement and proof are in Appendix A.7. As a result, we can think of sampling from $\mathcal{R}_1$ as analogous to sampling a high-quality sentence, and moreover, not reaching $\mathcal{R}_1$ implies the Markov chain sampling process has not mixed to the groundtruth distribution yet. In the analogy to

language tasks, in tasks like machine translation, for each source sentence, sampling one high-quality target sentence is potentially good enough. In some other tasks like creative writing, producing well-calibrated samples might be desirable—so mixing would be needed.

We show that running k**-Gibbs sampler** requires a small number of steps to reach $\mathcal{R}_1$. This implies that if a model can efficiently approximate one step of k**-Gibbs sampler**, then it is fast to sample a high-probability sequence by iteratively applying the model. Proof is in Appendix A.8.

Proposition 3 (k-Gibbs sampler sampling can reach the mode fast)**.** *On Ising model G in Equation* (3) *under Assumption 4, with any initial $\boldsymbol{X}^{(0)}$, $\forall \delta \in (0,1)$, with probability at least $1-\delta$, after $T \coloneqq \left\lceil \log_{c_{\mathcal{R}_1}} \delta \right\rceil$ steps of k**-Gibbs sampler** (Gibbs sampler 1) with $k \geq |C_G|$, we have $\{\boldsymbol{X}^{(t)} | t \in [T]\} \cap \mathcal{R}_1 \neq \emptyset$ in which the constant $c_{\mathcal{R}_1} \in (0,1)$, $c_{\mathcal{R}_1} \coloneqq 1 - \frac{\binom{N-|C_G|}{k-|C_G|}}{\binom{N}{k}} \frac{e^{2(J_0+h_G)}}{e^{2(J_0+h_G)}+e^{2J_0}+2^{|C_G|}-2}$*

By contrast, we show that for nontrivial probability over the randomness in the initial sequence, running **independent parallel** requires a large number of steps to reach the largest mode $\mathcal{R}_1$ of the distribution. This implies that the sampling process may not reach a high-probability sequence in less than exponentially large number of iterations.

Proposition 4 (Independent parallel sampling stuck in bad samples)**.** *On Ising model G in Equation* (3) *under Assumption 4, if the initial $\boldsymbol{X}^{(0)}$ is such that $\sum_{i \in C_G} \boldsymbol{X}_i^{(0)} \leq -2$, $\forall \delta \in (0,1)$, with probability at least $1-\delta$, after $T \coloneqq \left\lfloor \frac{\delta}{2} \exp\left(c_{stuck}\right) \right\rfloor$ steps of **independent parallel** (Gibbs sampler 2), we have $\forall t \in [T], \sum_{i \in C_G} \boldsymbol{X}_i^{(t)} \leq -2$, in which $c_{stuck} \coloneqq \frac{2\left(-1+\frac{1-\exp(-2J_0)}{\exp(-2J_0)+1}\frac{|C_G|}{2}\right)^2}{|C_G|}$*

The proof is in Appendix A.9. Combining Proposition 3 and Proposition 4 leads to a separation result between k**-Gibbs sampler** and **independent parallel**, in particular when the clique size in G is large and dependency is strong within the clique: with high probability, while the former reaches $\mathcal{R}_1$ in 1 step, the latter cannot do so in arbitrarily large number of steps. Proof is in Appendix A.10.

4 Experiments

4.1 Parallel Decoding by Iterative Refinement (PaDIR)

We consider an encoder-decoder architecture, in which the decoder is modified to be *non-autoregressive*: instead of iteratively predicting the next token, each of our decoder forward pass predicts an update to *all* target positions *in parallel*. The encoder extracts features from the source sequence, and based on these features, each decoder forward pass refines its current hypothesis of the target sequence. The initial decoder hypothesis is a purely random sequence, and more decoder forward passes correspond to more steps of refinement. Note that we are *not* the first in the literature to propose this language modeling paradigm. Our focus in this paper is to provide theoretical and empirical analyses to characterize its potentials, limitations and document useful training practices.

One-stage training Given source sequence X^{source} and target sequence X^{target} in the supervised training data $\mathcal{D}_{\text{train}}$, we use a preprocessing rule to create the initial hypothesis target sequence $X^{(0)}$. [11] The training objective is $L^{(1)} = \sum_{X^{\text{source}}, X^{\text{target}} \in \mathcal{D}_{\text{train}}} l(f^{\text{dec}}_{\theta_d}(X^{(0)}, f^{\text{enc}}_{\theta_e}(X^{\text{source}}))$ where l is the cross-entropy loss applied to each position.

Multi-stage training One limitation of the one-stage training is that the inference situation is *out-of-distribution*: when decoder step $t > 1$, the model needs to refine its own predictions in step $t-1$, which is not reflected in the training objective. Therefore, we use the multi-stage training objective (Ghazvininejad et al., 2020; Savinov et al., 2022): $L^{(S)} = \frac{1}{S}\sum_{s \in [S]} L^{(s)}$ where S is the number of training stages, and $L^{(s)} = \sum_{X^{\text{source}}, X^{\text{target}} \in \mathcal{D}_{\text{train}}} l(f^{\text{dec}}_{\theta_d}(X^{(s-1)}, f^{\text{enc}}_{\theta_e}(X^{\text{source}}))$

4.2 Training recipe and evaluation

We train models on machine translation datasets, provide practical recommendations based on our empirical observations, and discuss their connections to our theory.

[11] Each position in $X^{(0)}$ may contain a `[MASK]` token, a random token, or the correct token in X^{source}, dependning on the preprocessing rule.

Model training We use Transformer encoder-decoder with size similar to Transformer-Base (Vaswani et al., 2017) and T5-Small-1.0 (Raffel et al., 2020): 6 encoder and decoder layers, 8 attention heads, 512 embedding dimensions and 2048 FFN hidden dim. We add a positional attention mechanism (Gu et al., 2018; Kreutzer et al., 2020) in each Transformer layer and use learnt positional embeddings. The total number of parameters is 67M. We initialize model parameters randomly and train using a batch size of 2048 for 500k iterations, with a 10% dropout rate, 15% unmasking rate and 2 training stages. The optimizer is AdaFactor (Shazeer & Stern, 2018), with default T5X hyperparameters (Roberts et al., 2022). The learning rate peaks at 0.003 with a linear rampup for 10k steps followed by cosine decay, from and to a minimum value of $1e-5$. Unlike most prior work, we do not use a remasking schedule; [12] we simply remask token-level stutter (i.e., consecutive repeated tokens) across iterations and drop repeated tokens after the final iteration. As commonly done, we distill our models by training on the output of an autoregressive model. For simplicity, we use a Translation API [13] to generate this distillation data.

Datasets We evaluate our models on machine translation benchmarks commonly used in the non-autoregressive modeling literature. We conduct experiments on both directions of three WMT datasets: WMT14 DE↔EN ($4.5M$ examples) (Bojar et al., 2014), WMT16 RO↔EN ($610k$ examples) (Bojar et al., 2016) and WMT17 ZH↔EN ($20M$ examples) (Bojar et al., 2017). We load the data from the `tensorflow_datasets` library and do not apply any preprocessing other than sentence piece tokenization (Kudo & Richardson (2018)). Bilingual vocabularies of 32k tokens are created using the training sets of each language pair.

Benchmarking PaDIR models and AR models reach similar BLEU and BLEURT scores. Quantitative experimental results and common baselines are shown in Table 1 and Table 2 in Appendix B.3. We discuss several considerations for evaluation metrics in Appendix B.2.

Speed The average target length in all datasets ranges between 28 and 33 tokens, including the `EOS` token. As such a non-autogressive model using 4 decoding steps does 7 to 8 times fewer decoder passes. In practice we see an end-to-end speedup greater than >2x for the median and >5x for the 99th percentile latency on the same hardware [14] (with 4 decoding steps and batch size 1). The gap between expected and observed speedup is due to fixed costs (input tokenization, encoding, etc.) as well as a better optimization of AR decoding (e.g. through caching of intermediate results). For longer sequences, the constant number of decoding passes in GMLM is advantageous. For completeness, it is worth noting that the number of decoder passes necessary to achieve good quality (and thus model speed) is application dependent, with some tasks like non-autoregressive text in-painting remaining slower than their autoregressive counterparts, as shown in Savinov et al. (2022).

4.3 Connecting to theory: quantifying dependency via attention scores

Our theory suggests that stronger dependency between target positions leads to worse generalization guarantee and sampling efficiency. However, it is unclear how to measure such dependency for Transformer-based language models trained on natural language data. In this section, we empirically investigate: *how to predict what target positions have strong dependency which may be challenging for Transformers?* We test the following two hypotheses: (1) Strongly dependent target positions have larger **decoder self-attention** between each other. (2) Strongly dependent target positions have similar **cross-attention** distribution to source tokens.

For a pair of target positions, to measure how well their dependency is modeled in the generated output, we focus on adjacent repetitive tokens, a.k.a. *stutter*. Stuttering is a common error mode among parallel decoding models, and we use it as one reasonable proxy for measuring failures in modeling target-side dependency. We show Hypothesis 1 is unlikely to hold: even on average, stuttering positions do not have larger **decoder self-attention** between each other, compared with non-stuttering adjacent positions. [15] By contrary, Hypothesis 2 is potentially promising: with various of distribution distance measures, stuttering positions in the generated output have more similar **cross-attention** distributions to source tokens, compared with non-stuttering adjacent positions. Details are in Table 4 and Table 5 in Appendix B.

[12] We experimented with various remasking schedules but the results were not visibly affected.

[13] The AR baseline model is trained on the output of the same API, by distillation. During anonymous review, we remove the service name to avoid unnecessary associations.

[14] Full hardware detail will be provided for camera-ready paper.

[15] Since all stuttering positions are by definition adjacent, we think a fair comparison should only consider adjacent positions for non-stuttering position pairs.

REFERENCES

Nima Anari, Yizhi Huang, Tianyu Liu, Thuy-Duong Vuong, Brian Xu, and Katherine Yu. Parallel discrete sampling via continuous walks. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023, pp. 103–116, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450399135. doi: 10.1145/3564246.3585207. URL https://doi.org/10.1145/3564246.3585207.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=h7-XixPCAL.

Julian Besag. Statistical analysis of non-lattice data. *Journal of the Royal Statistical Society Series D: The Statistician*, 24(3):179–195, 1975.

Sergey G Bobkov and Prasad Tetali. Modified logarithmic sobolev inequalities in discrete settings. *Journal of Theoretical Probability*, 19(2):289–336, 2006.

Ond rej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pp. 131–198, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W/W16/W16-2301.

Ondrej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Ale s Tamchyna. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pp. 12–58, Baltimore, Maryland, USA, June 2014. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W/W14/W14-3302.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. Findings of the 2017 conference on machine translation (WMT17). In Ondřej Bojar, Christian Buck, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, and Julia Kreutzer (eds.), *Proceedings of the Second Conference on Machine Translation*, pp. 169–214, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4717. URL https://aclanthology.org/W17-4717.

Tom Bosc and Pascal Vincent. Do sequence-to-sequence VAEs learn global features of sentences? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4296–4318, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.350. URL https://aclanthology.org/2020.emnlp-main.350.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pp. 10–21, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/K16-1002. URL https://aclanthology.org/K16-1002.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33,

pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D. Lee, Deming Chen, and Tri Dao. Medusa: Simple llm inference acceleration framework with multiple decoding heads, 2024.

Pietro Caputo and Daniel Parisi. Block factorization of the relative entropy via spatial mixing. *Communications in Mathematical Physics*, 388(2):793–818, 2021.

Pietro Caputo, Georg Menz, and Prasad Tetali. Approximate tensorization of entropy at high temperature. In *Annales de la Faculté des sciences de Toulouse: Mathématiques*, volume 24, pp. 691–716, 2015.

William Chan, Chitwan Saharia, Geoffrey Hinton, Mohammad Norouzi, and Navdeep Jaitly. Imputer: Sequence modelling via imputation and dynamic programming. In *International Conference on Machine Learning*, pp. 1403–1413. PMLR, 2020.

Tong Che, Yanran Li, Ruixiang Zhang, R Devon Hjelm, Wenjie Li, Yangqiu Song, and Yoshua Bengio. Maximum-likelihood augmented discrete generative adversarial networks, 2017.

Yuntian Deng, Anton Bakhtin, Myle Ott, Arthur Szlam, and Marc'Aurelio Ranzato. Residual energy-based models for text generation. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=B1l4SgHKDH.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Persi Diaconis and Laurent Saloff-Coste. Logarithmic sobolev inequalities for finite markov chains. *The Annals of Applied Probability*, 6(3):695–750, 1996.

Benjamin L Edelman, Surbhi Goel, Sham Kakade, and Cyril Zhang. Inductive biases and variable creation in self-attention mechanisms. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 5793–5831. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/edelman22a.html.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. Mask-predict: Parallel decoding of conditional masked language models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 6112–6121, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1633. URL https://aclanthology.org/D19-1633.

Marjan Ghazvininejad, Omer Levy, and Luke Zettlemoyer. Semi-autoregressive training improves mask-predict decoding, 2020.

Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and Lingpeng Kong. Diffuseq: Sequence to sequence text generation with diffusion models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=jQj-_rLVXsj.

Kartik Goyal, Chris Dyer, and Taylor Berg-Kirkpatrick. Exposing the implicit energy networks behind masked language models via metropolis–hastings. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=6PvWo1kEvlT.

Jiatao Gu and Xiang Kong. Fully non-autoregressive neural machine translation: Tricks of the trade. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 120–133, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.11. URL https://aclanthology.org/2021.findings-acl.11.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor O.K. Li, and Richard Socher. Non-autoregressive neural machine translation. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=B1l8BtlCb.

Jiaxian Guo, Sidi Lu, Han Cai, Weinan Zhang, Yong Yu, and Jun Wang. Long text generation via adversarial training with leaked information. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. ISBN 978-1-57735-800-8.

Jaroslav Hájek. Local asymptotic minimax and admissibility in estimation. In *Proceedings of the sixth Berkeley symposium on mathematical statistics and probability*, volume 1, pp. 175–194, 1972.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *The collected works of Wassily Hoeffding*, pp. 409–426, 1994.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rygGQyrFvH.

Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=6nbpPqUCIi7.

Fuchun Huang and Yosihiko Ogata. Generalized pseudo-likelihood estimates for markov random fields on lattice. *Annals of the Institute of Statistical Mathematics*, 54:1–18, 2002.

Samy Jelassi, Michael Eli Sander, and Yuanzhi Li. Vision transformers provably learn spatial structure. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=eMW9AkXaREI.

Jungo Kasai, James Cross, Marjan Ghazvininejad, and Jiatao Gu. Non-autoregressive machine translation with disentangled context transformer. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In Jian Su, Kevin Duh, and Xavier Carreras (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1317–1327, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1139. URL https://aclanthology.org/D16-1139.

Frederic Koehler, Alexander Heckett, and Andrej Risteski. Statistical efficiency of score matching: The view from isoperimetry. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=TD7AnQjNzR6.

Xiang Kong, Zhisong Zhang, and Eduard Hovy. Incorporating a local translation mechanism into non-autoregressive translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1067–1073, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.79. URL https://aclanthology.org/2020.emnlp-main.79.

Julia Kreutzer, George Foster, and Colin Cherry. Inference strategies for machine translation with conditional masking. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5774–5782, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.465. URL https://aclanthology.org/2020.emnlp-main.465.

Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In Eduardo Blanco and Wei Lu (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System*

Demonstrations, pp. 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-2012. URL https://aclanthology.org/D18-2012.

Holden Lee. Parallelising glauber dynamics, 2023.

Jason Lee, Elman Mansimov, and Kyunghyun Cho. Deterministic non-autoregressive neural sequence modeling by iterative refinement. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 1173–1182, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1149. URL https://aclanthology.org/D18-1149.

Jason Lee, Raphael Shu, and Kyunghyun Cho. Iterative refinement in the continuous space for non-autoregressive neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1006–1015, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.73. URL https://aclanthology.org/2020.emnlp-main.73.

Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori Hashimoto. Diffusion-LM improves controllable text generation. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=3s9IrEsjLyk.

Yuchen Li and Andrej Risteski. The limitations of limited context for constituency parsing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 2675–2687, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.208. URL https://aclanthology.org/2021.acl-long.208.

Yuchen Li, Yuanzhi Li, and Andrej Risteski. How do transformers learn topic structure: Towards a mechanistic understanding. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19689–19729. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/li23p.html.

Chu-Cheng Lin, Aaron Jaech, Xin Li, Matthew R. Gormley, and Jason Eisner. Limitations of autoregressive models and their alternatives. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5147–5173, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.405. URL https://aclanthology.org/2021.naacl-main.405.

Kevin Lin, Dianqi Li, Xiaodong He, Zhengyou Zhang, and Ming-Ting Sun. Adversarial ranking for language generation. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 3158–3168, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Bingbin Liu, Daniel Hsu, Pradeep Kumar Ravikumar, and Andrej Risteski. Masked prediction: A parameter identifiability view. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=Hbvlb4D1aFC.

Bingbin Liu, Jordan T. Ash, Surbhi Goel, Akshay Krishnamurthy, and Cyril Zhang. Transformers learn shortcuts to automata. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=De4FYqjFueZ.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion language modeling by estimating the ratios of the data distribution, 2023.

Haoye Lu, Yongyi Mao, and Amiya Nayak. On the dynamics of training attention models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=1OCTOShAmqB.

Xuezhe Ma, Chunting Zhou, Xian Li, Graham Neubig, and Eduard Hovy. FlowSeq: Non-autoregressive conditional sequence generation with generative flow. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4282–4292, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1437. URL https://aclanthology.org/D19-1437.

Katalin Marton. An inequality for relative entropy and logarithmic sobolev inequalities in euclidean spaces. *Journal of Functional Analysis*, 264(1):34–61, 2013.

Katalin Marton. Logarithmic sobolev inequalities in discrete product spaces: a proof by a transportation cost distance. *arXiv preprint arXiv:1507.02803*, 2015.

Yu Meng, Jitin Krishnan, Sinong Wang, Qifan Wang, Yuning Mao, Han Fang, Marjan Ghazvininejad, Jiawei Han, and Luke Zettlemoyer. Representation deficiency in masked language modeling. *arXiv preprint arXiv:2302.02060*, 2023.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In Pierre Isabelle, Eugene Charniak, and Dekang Lin (eds.), *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040.

Matt Post. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pp. 186–191, Belgium, Brussels, October 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W18-6319.

Amy Pu, Hyung Won Chung, Ankur P. Parikh, Sebastian Gehrmann, and Thibault Sellam. Learning compact metrics for MT. *CoRR*, abs/2110.06341, 2021. URL https://arxiv.org/abs/2110.06341.

Lianhui Qin, Sean Welleck, Daniel Khashabi, and Yejin Choi. COLD decoding: Energy-based constrained text generation with langevin dynamics. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=TiZYrQ-mPup.

Yilong Qin and Andrej Risteski. Fit like you sample: Sample-efficient generalized score matching from fast mixing diffusions, 2023.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

Pradeep Ravikumar, Martin J Wainwright, and John D Lafferty. High-dimensional ising model selection using l1-regularized logistic regression. *Ann. Statist. 38(3): 1287-1319 (June 2010). DOI: 10.1214/09-AOS691*, 2010.

Machel Reid, Vincent Josua Hellendoorn, and Graham Neubig. Diffuser: Diffusion via edit-based reconstruction. In *The Eleventh International Conference on Learning Representations*, 2022.

Adam Roberts, Hyung Won Chung, Anselm Levskaya, Gaurav Mishra, James Bradbury, Daniel Andor, Sharan Narang, Brian Lester, Colin Gaffney, Afroz Mohiuddin, Curtis Hawthorne, Aitor Lewkowycz, Alex Salcianu, Marc van Zee, Jacob Austin, Sebastian Goodman, Livio Baldini Soares, Haitang Hu, Sasha Tsvyashchenko, Aakanksha Chowdhery, Jasmijn Bastings, Jannis Bulian, Xavier Garcia, Jianmo Ni, Andrew Chen, Kathleen Kenealy, Jonathan H. Clark, Stephan Lee, Dan Garrette, James Lee-Thorp, Colin Raffel, Noam Shazeer, Marvin Ritter, Maarten Bosma, Alexandre Passos, Jeremy Maitin-Shepard, Noah Fiedel, Mark Omernick, Brennan Saeta, Ryan Sepassi, Alexander Spiridonov, Joshua Newlan, and Andrea Gesmundo. Scaling up models and data with t5x and seqio. *arXiv preprint arXiv:2203.17189*, 2022. URL https://arxiv.org/abs/2203.17189.

Chitwan Saharia, William Chan, Saurabh Saxena, and Mohammad Norouzi. Non-autoregressive machine translation with latent alignments. *arXiv preprint arXiv:2004.07437*, 2020.

Nikolay Savinov, Junyoung Chung, Mikolaj Binkowski, Erich Elsen, and Aaron van den Oord. Step-unrolled denoising autoencoders for text generation. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=T0GpzBQ1Fg6.

Robin M. Schmidt, Telmo Pires, Stephan Peitz, and Jonas Lööf. Non-autoregressive neural machine translation: A call for clarity, 2022.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. BLEURT: learning robust metrics for text generation. *CoRR*, abs/2004.04696, 2020. URL https://arxiv.org/abs/2004.04696.

Noam Shazeer and Mitchell Stern. Adafactor: Adaptive learning rates with sublinear memory cost. *CoRR*, abs/1804.04235, 2018. URL http://arxiv.org/abs/1804.04235.

Mitchell Stern, William Chan, Jamie Kiros, and Jakob Uszkoreit. Insertion transformer: Flexible sequence generation via insertion operations. In *International Conference on Machine Learning*, pp. 5976–5985. PMLR, 2019.

Lucas Torroba Hennigen and Yoon Kim. Deriving language models from masked language models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 1149–1159, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-short.99. URL https://aclanthology.org/2023.acl-short.99.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Aad W Van der Vaart. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Marc Vuffray, Sidhant Misra, Andrey Lokhov, and Michael Chertkov. Interaction screening: Efficient and sample-optimal learning of ising models. *Advances in neural information processing systems*, 29, 2016.

Alex Wang and Kyunghyun Cho. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In Antoine Bosselut, Asli Celikyilmaz, Marjan Ghazvininejad, Srinivasan Iyer, Urvashi Khandelwal, Hannah Rashkin, and Thomas Wolf (eds.), *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pp. 30–36, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-2304. URL https://aclanthology.org/W19-2304.

Colin Wei, Yining Chen, and Tengyu Ma. Statistically meaningful approximation: a case study on approximating turing machines with transformers, 2021. URL https://arxiv.org/abs/2107.13163.

Kaiyue Wen, Yuchen Li, Bingbin Liu, and Andrej Risteski. Transformers are uninterpretable with myopic methods: a case study with bounded dyck grammars. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=OitmaxSAUu.

Shunyu Yao, Binghui Peng, Christos Papadimitriou, and Karthik Narasimhan. Self-attention networks can process bounded hierarchical languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 3770–3785, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.292. URL https://aclanthology.org/2021.acl-long.292.

Tom Young and Yang You. On the inconsistencies of conditionals learned by masked language models. *arXiv preprint arXiv:2301.00068*, 2022.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pp. 2852–2858. AAAI Press, 2017.

Chulhee Yun, Srinadh Bhojanapalli, Ankit Singh Rawat, Sashank Reddi, and Sanjiv Kumar. Are transformers universal approximators of sequence-to-sequence functions? In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ByxRM0Ntvr.

Haoyu Zhao, Abhishek Panigrahi, Rong Ge, and Sanjeev Arora. Do transformers parse while predicting the masked word? In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 16513–16542, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.1029. URL https://aclanthology.org/2023.emnlp-main.1029.

Lin Zheng, Jianbo Yuan, Lei Yu, and Lingpeng Kong. A reparameterized discrete diffusion model for text generation, 2023.

Chunting Zhou, Jiatao Gu, and Graham Neubig. Understanding knowledge distillation in non-autoregressive machine translation. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=BygFVAEKDH.

Zachary Ziegler and Alexander Rush. Latent normalizing flows for discrete sequences. In *International Conference on Machine Learning*, pp. 7673–7682. PMLR, 2019.

Supplementary Material

CONTENTS

1 Introduction **1**

2 Theoretical framework **2**

2.1 Asymptotic sample efficiency via functional inequalities 2

2.2 Sample complexity via mixing time bounds 3

2.2.1 Masking more is (statistically) better 4

2.2.2 Statistical efficiency bounds via mixing time bounds 4

2.3 Finite sample bounds and distributional distance 5

3 Sampling efficiency via Gibbs-like algorithms **6**

3.1 Can Transformers implement Markov chains via parallel decoding? 6

3.2 Accurately approximating conditionals can be (much) better 7

4 Experiments **8**

4.1 Parallel Decoding by Iterative Refinement (PaDIR) 8

4.2 Training recipe and evaluation 8

4.3 Connecting to theory: quantifying dependency via attention scores 9

A Proofs and theoretical backgrounds **19**

A.1 Optimization landscape for fitting the conditional distributions 19

A.2 Proof of Lemma 2: Generalized information matrix equality 20

A.3 Lemma 3: Regularity conditions for asymptotic behavior of parameter estimation . 22

A.4 Proof of Theorem 1: Masking more is (statistically) better 23

A.5 Proof of Theorem 2: Asymptotic variance under a Poincaré Inequality 24

A.6 Proof of Theorem 3: Generalization bound for learning the joint distribution 25

A.7 Proof of Proposition 8: Modes of the strongly ferromagnetic Ising model 32

A.8 Proof of Proposition 3: k-Gibbs sampler can reach the mode fast 34

A.9 Proof of Proposition 4 independent parallel sampling stuck in bad samples 35

A.10 Proof of Corollary 3: Separation between N-Gibbs sampler and independent parallel sampling 37

A.11 Background and proofs of Proposition 1 and Proposition 2: on the expressive power of Transformers for implementing sequence-to-sequence Markov chains in parallel 39

B Additional experimental details **41**

B.1 Inference and training setting 41
B.1.1 Inference 41
B.2 Discussion on modeling and metrics 42
B.3 Quantitative experimental results 43

C Additional related works **45**

A PROOFS AND THEORETICAL BACKGROUNDS

A.1 OPTIMIZATION LANDSCAPE FOR FITTING THE CONDITIONAL DISTRIBUTIONS

We explain a comment in Section 2.

Is the pseudo-likelihood training objective $L_{\text{MLPE}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}})$ (Definition 2) intrinsically harder to optimize? We show that it is not the case: training a classic parameteric model for distributions (namely, Ising models) on $L_{\text{MLPE}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}})$ is in fact *convex*: [16]

Ising models. For random variables $\boldsymbol{X} = \{X_i \in \{-1, 1\} : i \in [N]\}$, an Ising model with parameters $\boldsymbol{J} \in \mathbb{R}^{N \times N}$ and $\boldsymbol{h} \in \mathbb{R}^N$ has joint distribution

$$p(\mathbf{X} = \mathbf{x}) = \frac{1}{Z} \exp\big(\sum_{i \in [N]} \boldsymbol{h}_i \mathbf{x}_i + \sum_{i \neq j \in [N]} \boldsymbol{J}_{ij} \mathbf{x}_i \mathbf{x}_j \big), \tag{A.6}$$

in which Z is the partition function.

Proposition 5 (Fitting an Ising model over the conditional distributions is convex)**.** *When p_θ is an Ising model (Equation* (A.6)*), i.e. $\theta = (\boldsymbol{J}, \boldsymbol{h})$. The objective $L_{MLPE}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}})$ is convex in θ.*

Remark 2. *When the parameterization of p_θ admits a benign loss landscape and is sufficiently expressive, Proposition 5 suggests that there exists efficient algorithms for finding $\hat{\theta}$ such that $L_{MLPE}(\hat{\theta}; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}})$ is small. In Section 2.3 we will show that this also implies generalization guarantee on the learned* joint *distribution $p_{\hat{\theta}}$.*

Proof. Recall that

$$L_{\text{MLPE}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) = \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{j, X \in \mathcal{S}_{\mathcal{X}}} \frac{1}{|\mathcal{S}_{\mathcal{K}}(j)|} \sum_{K \in \mathcal{S}_{\mathcal{K}}(j)} D_{\text{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right)$$

Hence it suffices to prove that $D_{\text{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right)$ is convex in θ. Note that

$$\begin{aligned} D_{\text{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right) &= \sum_{X_K \in \Omega^{|K|}} \tilde{p}(X_K|X_{-K}) \ln \frac{\tilde{p}(X_K|X_{-K})}{p_\theta(X_K|X_{-K})} \\ &= \sum_{X_K \in \Omega^{|K|}} \tilde{p}(X_K|X_{-K}) \left[\ln \tilde{p}(X_K|X_{-K}) - \ln p_\theta(X_K|X_{-K})\right] \end{aligned}$$

Hence it suffices to prove that $-\ln p_\theta(X_K = \mathbf{x}_K | X_{-K} = \mathbf{x}_{-K})$ is convex in θ.

When p_θ is an Ising model (Equation (A.6)),

$$\begin{aligned} -\ln p_\theta(\mathbf{x}_K|\mathbf{x}_{-K}) &= -\ln \frac{\exp\left(\sum_{i \in [N]} \boldsymbol{h}_i \mathbf{x}_i + \sum_{i \neq j \in [N]} \boldsymbol{J}_{ij} \mathbf{x}_i \mathbf{x}_j\right)}{Z(\mathbf{x}_{-K})} \\ &= -\big(\sum_{i \in [N]} \boldsymbol{h}_i \mathbf{x}_i + \sum_{i \neq j \in [N]} \boldsymbol{J}_{ij} \mathbf{x}_i \mathbf{x}_j \big) + \ln Z(\mathbf{x}_{-K}) \end{aligned}$$

in which the denominator

$$\begin{aligned} Z(\mathbf{x}_{-K}) = \sum_{X_K \in \Omega^{|K|}} \exp\Bigg(&\sum_{i \in K} \boldsymbol{h}_i X_K + \sum_{i \in [N] \setminus K} \boldsymbol{h}_i \mathbf{x}_i \\ &+ \sum_{i \neq j \in [K]} \boldsymbol{J}_{ij} X_i X_j + \sum_{i \in K, j \in [N] \setminus K} \boldsymbol{J}_{ij} X_i \mathbf{x}_j + \sum_{i \neq j \in [N] \setminus K} \boldsymbol{J}_{ij} \mathbf{x}_i \mathbf{x}_j \Bigg) \end{aligned}$$

Note that $-(\sum_{i \in [N]} \boldsymbol{h}_i \mathbf{x}_i + \sum_{i \neq j \in [N]} \boldsymbol{J}_{ij} \mathbf{x}_i \mathbf{x}_j)$ is linear in (h, J) and $\ln Z(\mathbf{x}_{-K})$ is convex in (h, J), so $-\ln p_\theta(X_K = \mathbf{x}_K | X_{-K} = \mathbf{x}_{-K})$ is convex in (h, J), which completes the last piece of the proof.

□

[16] A known fact which has been used to design (provably) efficient algorithms for learning bounded-degree Ising models (Ravikumar et al., 2010; Vuffray et al., 2016).

A.2 Proof of Lemma 2: Generalized information matrix equality

Lemma 2 (Generalized information matrix equality)**.** *Under Assumption 1, the α-weighted pseudo-likelihood loss (Definition 2) verifies:* $\nabla_\theta^2 L_{PL}(\theta) = \mathrm{Cov}(\nabla_\theta l_{PL}(\theta))$

As a consequence, the pseudolikelihood estimator $\hat{\theta}_n$ is asymptotically normal with the following asymptotic covariance matrix: $\sqrt{|\mathcal{S}_{\mathcal{X}}|}(\hat{\theta}_{PL} - \theta^*) \to \mathcal{N}(0, \left(\nabla_\theta^2 L_{PL}(\theta^*)\right)^{-1})$

Proof. **Step 1: Assumption 1 allows us to change the order of expectation and derivatives**

First, since Ω, $[N]$, and $K \subset [N]$ are both discrete finite, the conditions for the Dominated Convergence Theorem holds under Assumption 1: there exists function $f : \Theta \times \Omega \times \mathcal{K} \mapsto \mathbb{R}$ such that $\forall \theta \in \Theta$, $\mathbb{E}_{X,K}\left[f(\theta, X, K)\right] < \infty$, $\|\nabla_\theta \log p_\theta(x_K|x_{-K})\|_2 \leq f(\theta, X, K)$, and $\|\nabla_\theta^2 \log p_\theta(x_K|x_{-K})\|_F \leq f(\theta, X, K)$.

Therefore,

$$\begin{aligned}\frac{\partial}{\partial \theta_j}\mathbb{E}_{S,x_S,x_{-S}}\left[\log p_\theta(x_S|x_{-S})\right] &= \lim_{h\to 0}\frac{1}{h}\left(\mathbb{E}_{S,x_S,x_{-S}}\left[\log p_{\theta+\boldsymbol{e}_j h}(x_S|x_{-S})\right] - \mathbb{E}_{S,x_S,x_{-S}}\left[\log p_\theta(x_S|x_{-S})\right]\right)\\ &= \lim_{h\to 0}\mathbb{E}_{S,x_S,x_{-S}}\left[\frac{\log p_{\theta+\boldsymbol{e}_j h}(x_S|x_{-S}) - \log p_\theta(x_S|x_{-S})}{h}\right]\end{aligned}$$

By Mean Value Theorem, there exists $\xi(h) \in (0, h)$ such that

$$\frac{\log p_{\theta+\boldsymbol{e}_j h}(x_S|x_{-S}) - \log p_\theta(x_S|x_{-S})}{h} = \frac{\partial}{\partial_{\theta_j}}\log p_{\theta+\boldsymbol{e}_j \xi(h)}(x_S|x_{-S})$$

So

$$\begin{aligned}&\frac{\partial}{\partial \theta_j}\mathbb{E}_{S,x_S,x_{-S}}\left[\log p_\theta(x_S|x_{-S})\right]\\ &= \lim_{h\to 0}\left(\mathbb{E}_{S,x_S,x_{-S}}\left[\frac{\partial}{\partial_{\theta_j}}\log p_{\theta+\boldsymbol{e}_j\xi(h)}(x_S|x_{-S})\right]\right)\\ &= \mathbb{E}_{S,x_S,x_{-S}}\left[\lim_{h\to 0}\left(\frac{\partial}{\partial_{\theta_j}}\log p_{\theta+\boldsymbol{e}_j\xi(h)}(x_S|x_{-S})\right)\right] \quad \text{(Dominated Convergence Thm and Assumption 1)}\\ &= \mathbb{E}_{S,x_S,x_{-S}}\left[\frac{\partial}{\partial_{\theta_j}}\log p_\theta(x_S|x_{-S})\right]\end{aligned}$$

So

$$\nabla_\theta \mathbb{E}_{S,x_S,x_{-S}}\log p_\theta(x_S|x_{-S}) = \mathbb{E}_{S,x_S,x_{-S}}\nabla_\theta \log p_\theta(x_S|x_{-S})$$

Likewise, by Mean Value Theorem, Dominated Convergence Thm and Assumption 1,

$$\frac{\partial^2}{\partial\theta_i\partial\theta_j}\mathbb{E}_{S,x_S,x_{-S}}\left[\log p_\theta(x_S|x_{-S})\right] = \mathbb{E}_{S,x_S,x_{-S}}\left[\frac{\partial^2}{\partial\theta_i\partial\theta_j}\log p_\theta(x_S|x_{-S})\right]$$

and so

$$\nabla_\theta^2 \mathbb{E}_{S,x_S,x_{-S}}\log p_\theta(x_S|x_{-S}) = \mathbb{E}_{S,x_S,x_{-S}}\nabla_\theta^2 \log p_\theta(x_S|x_{-S})$$

Step 2: rewrite $\nabla_\theta^2 L_{PL}(\theta)$

$$\begin{aligned}\nabla_\theta^2 L_{PL}(\theta) &= -\nabla_\theta^2 \mathbb{E}_{S,x_S,x_{-S}}\log p_\theta(x_S|x_{-S})\\ &\overset{①}{=} -\mathbb{E}_{S,x_S,x_{-S}}\nabla_\theta^2 \log p_\theta(x_S|x_{-S})\\ &\overset{②}{=} \mathbb{E}_{S,x_S,x_{-S}}\nabla_\theta \log p_\theta(x_S|x_{-S})\nabla_\theta \log p_\theta(x_S|x_{-S})^\top - \frac{\nabla_\theta^2 p_\theta(x_S|x_{-S})}{p_\theta(x_S|x_{-S})}\\ &\overset{③}{=} \mathbb{E}_{S,x_S,x_{-S}}\nabla_\theta \log p_\theta(x_S|x_{-S})\nabla_\theta \log p_\theta(x_S|x_{-S})^\top\end{aligned}$$

where ① follows by exchanging the order of expectation and Hessian ($S \in \mathcal{S}_k$ and $x \in \Omega$ are finite), and this is valid by **Step 1** above , ② by an application of chain rule. The last equality ③ follows by a similar calculation as the proof of the classical information matrix equality:

$$\begin{aligned}
\mathbb{E}_{S,x_S,x_{-S}} \frac{\nabla_\theta^2 p_\theta(x_S|x_{-S})}{p_\theta(x_S|x_{-S})} &= \mathbb{E}_S \mathbb{E}_{x_{-S}} \mathbb{E}_{x_S|x_{-S}} \frac{\nabla_\theta^2 p_\theta(x_S|x_{-S})}{p_\theta(x_S|x_{-S})} \\
&= \mathbb{E}_S \mathbb{E}_{x_{-S}} \int \nabla_\theta^2 p_\theta(x_S|x_{-S}) dx_S \\
&= \mathbb{E}_S \mathbb{E}_{x_{-S}} \nabla_\theta^2 \int p_\theta(x_S|x_{-S}) dx_S \\
&= 0
\end{aligned}$$

where the last equality follows since $\int p_\theta(x_S|x_{-S}) dx_S = 1$ (so doesn't depend on θ).

Similarly, we have

$$\begin{aligned}
\mathbb{E}_{S,x_S,x_{-S}} \nabla_\theta \log p_\theta(x_S|x_{-S}) &= \mathbb{E}_S \mathbb{E}_{x_{-S}} \mathbb{E}_{x_S|x_{-S}} \frac{\nabla_\theta p_\theta(x_S|x_{-S})}{p_\theta(x_S|x_{-S})} \\
&= \mathbb{E}_S \mathbb{E}_{x_{-S}} \int \nabla_\theta p_\theta(x_S|x_{-S}) dx_S \\
&= \mathbb{E}_S \mathbb{E}_{x_{-S}} \nabla_\theta \int p_\theta(x_S|x_{-S}) dx_S \\
&= 0
\end{aligned}$$

where the last equality follows since $\int p_\theta(x_S|x_{-S}) dx_S = 1$ (so doesn't depend on θ). Plugging this into the definition of covariance, we have:

$$\begin{aligned}
\mathrm{Cov}(\nabla_\theta l_{PL}(\theta)) &= \mathbb{E}_{S,x_S,x_{-S}} \nabla_\theta \log p_\theta(x_S|x_{-S}) \nabla_\theta \log p_\theta(x_S|x_{-S})^\top \\
&- \mathbb{E}_{S,x_S,x_{-S}} \nabla_\theta \log p_\theta(x_S|x_{-S}) \mathbb{E}_{S,x_S,x_{-S}} \nabla_\theta \log p_\theta(x_S|x_{-S})^\top \\
&= \mathbb{E}_{S,x_S,x_{-S}} \nabla_\theta \log p_\theta(x_S|x_{-S}) \nabla_\theta \log p_\theta(x_S|x_{-S})^\top
\end{aligned}$$

The proof of the lemma thus follows. □

A.3 LEMMA 3: REGULARITY CONDITIONS FOR ASYMPTOTIC BEHAVIOR OF PARAMETER ESTIMATION

With infinite samples, estimators like max likelihood or max pseudolikelihood converge in distribution to a normal distribution, under mild regularity conditions:

Lemma 3 (Van der Vaart (2000), Theorem 5.23; adopted precise statement in Qin & Risteski (2023))**.** *Consider a loss $L : \Theta \mapsto \mathbb{R}$, such that $L(\theta) = \mathbb{E}_p[\ell_\theta(x)]$ for $l_\theta : \mathcal{X} \mapsto \mathbb{R}$. Let Θ^* be the set of global minima of L, that is*

$$\Theta^* = \{\theta^* : L(\theta^*) = \min_{\theta \in \Theta} L(\theta)\}$$

Suppose the following conditions are met:

- *(Gradient bounds on l_θ) The map $\theta \mapsto l_\theta(x)$ is measurable and differentiable at every $\theta^* \in \Theta^*$ for p-almost every x. Furthermore, there exists a function $B(x)$, s.t. $\mathbb{E}\left[B(x)^2\right] < \infty$ and for every θ_1, θ_2 near θ^*, we have:*

$$|l_{\theta_1}(x) - l_{\theta_2}(x)| < B(x)\|\theta_1 - \theta_2\|_2$$

- *(Twice-differentiability of L) $L(\theta)$ is twice-differentiable at every $\theta^* \in \Theta^*$ with Hessian $\nabla^2_\theta L(\theta^*)$, and furthermore $\nabla^2_\theta L(\theta^*) \succ 0$.*

- *(Uniform law of large numbers) The loss L satisfies a uniform law of large numbers, that is*

$$\sup_{\theta \in \Theta} \left| \hat{\mathbb{E}}[l_\theta(x)] - L(\theta) \right| \xrightarrow{p} 0$$

Then, for every $\theta^* \in \Theta^*$, and every sufficiently small neighborhood S of θ^*, there exists a sufficiently large n, such that there is a unique minimizer $\hat{\theta}_n$ of $\hat{\mathbb{E}}[l_\theta(x)]$ in S. Furthermore, $\hat{\theta}_n$ satisfies:

$$\sqrt{n}(\hat{\theta}_n - \theta^*) \xrightarrow{d} \mathcal{N}\left(0, (\nabla^2_\theta L(\theta^*))^{-1} Cov(\nabla_\theta \ell(\theta^*; x))(\nabla^2_\theta L(\theta^*))^{-1}\right)$$

A.4 Proof of Theorem 1: Masking more is (statistically) better

Theorem 1 (Masking more is (statistically) better)**.** *Under Assumption 1, for every $k \in [N-1]$, let Γ_{PL}^k denote the asymptotic variance of the k-MPLE estimator (Definition 3). We have:*[17] $\Gamma_{PL}^{k+1} \preceq \Gamma_{PL}^k$

Proof. By Lemma 2, we have:

$$\nabla_\theta^2 L_{PL}^k = \mathbb{E}_{S\sim\alpha}\mathbb{E}_{x_S,x_{-S}}\nabla_\theta \log p_\theta(x_S|x_{-S})\nabla_\theta \log p_\theta(x_S|x_{-S})^\top$$

Let $\mathcal{S}_k$ denote the set

$$\{K \subset [N] \mid |K| = k\}$$

Moreover, for every $T \in \mathcal{S}_{k+1}$ and $a \in T$

$$\begin{aligned}\log p(x_T|x_{-T}) &= \log p(x_S, x_a|x_{-S\setminus\{a\}}) \text{ where } S = T\setminus\{a\}\\ &= \log p(x_a|x_{-S\setminus\{a\}}) + \log p(x_S|x_{-S})\end{aligned}$$

Using this, we can write:

$$\begin{aligned}\nabla_\theta^2 L_{PL}^{k+1} &= \mathbb{E}_{T\sim S_{k+1}}\mathbb{E}_{x_T,x_{-T}}\nabla_\theta \log p_\theta(x_T|x_{-T})\nabla_\theta \log p_\theta(x_T|x_{-T})^\top\\ &= \mathbb{E}_{S\sim S_k}\mathbb{E}_{a\notin S}\mathbb{E}_{x_S,x_a,x_{-S\setminus\{a\}}}\nabla_\theta \log p_\theta(x_T|x_{-T})\nabla_\theta \log p_\theta(x_T|x_{-T})^\top\\ &= \mathbb{E}_{S\sim S_k}\mathbb{E}_{a\notin S}\mathbb{E}_{x_S,x_a,x_{-S\setminus\{a\}}}[\nabla \log p(x_a|x_{-S\setminus\{a\}}) + \nabla \log p(x_S|x_{-S})][\nabla \log p(x_a|x_{-S\setminus\{a\}})\\ &\quad + \nabla \log p(x_S|x_{-S})]^\top \end{aligned} \tag{A.7}$$

Let us denote: $\begin{cases} A = \mathbb{E}_{S\sim S_k}\mathbb{E}_{a\notin S}\mathbb{E}_x \nabla \log p(x_S|x_{-S})\nabla \log p(x_S|x_{-S})^\top\\ B = \mathbb{E}_{S\sim S_k}\mathbb{E}_{a\notin S}\mathbb{E}_x \nabla \log p(x_a|x_{-S\setminus\{a\}})\nabla \log p(x_S|x_{-S})^\top\\ C = \mathbb{E}_{S\sim S_k}\mathbb{E}_{a\notin S}\mathbb{E}_x \nabla \log p(x_a|x_{-S\setminus\{a\}})\nabla \log p(x_a|x_{-S\setminus\{a\}})^\top\end{cases}$

By expanding the previous expression, we have $\nabla_\theta^2 L_{PL}^{k+1} = A + B + B^\top + C$.

Consider A first. Note that for a fixed $S \in S_k$, $\mathbb{E}_x \nabla \log p(x_S|x_{-S})\nabla \log p(x_S|x_{-S})^\top$ is independent of $a \notin S$ and therefore:

$$\begin{aligned}A &= \mathbb{E}_{S\sim S_k}\mathbb{E}_{a\notin S}\mathbb{E}_x \nabla \log p(x_S|x_{-S})\nabla \log p(x_S|x_{-S})^\top\\ &= \mathbb{E}_{S\sim S_k}\mathbb{E}_x \nabla \log p(x_S|x_{-S})\nabla \log p(x_S|x_{-S})^\top\\ &= \nabla_\theta^2 L_{PL}^k\end{aligned}$$

Proceeding to B, for a given $S \in S_k, x_{-S}$, we have $\mathbb{E}_{x_S|x_{-S}}[\nabla_\theta \log p(x_S|x_{-S})] = 0$ therefore:

$$\begin{aligned}B &= \mathbb{E}_{S\sim S_k}\mathbb{E}_{a\notin S}\mathbb{E}_{x_S,x_a,x_{-S\setminus\{a\}}}[\nabla \log p(x_a|x_{-S\setminus\{a\}})\nabla \log p(x_S|x_{-S})^\top]\\ &= \mathbb{E}_{S\sim S_k}\mathbb{E}_{a\notin S}\mathbb{E}_{x_a,x_{-S\setminus\{a\}}}[\mathbb{E}_{x_S|x_{-S}}(\nabla \log p(x_a|x_{-S\setminus\{a\}})\nabla \log p(x_S|x_{-S})^\top)]\\ &= \mathbb{E}_{S\sim S_k}\mathbb{E}_{a\notin S}\mathbb{E}_{x_a,x_{-S\setminus\{a\}}}[\nabla \log p(x_a|x_{-S\setminus\{a\}})\mathbb{E}_{x_S|x_{-S}}\nabla \log p(x_S|x_{-S})^\top]\\ &= 0\end{aligned}$$

Finally, each term $\nabla \log p(x_S|x_{-S})\nabla \log p(x_S|x_{-S})^\top \succeq 0$ therefore $C \succeq 0$.

Plugging this back in (A.7), we have:

$$\nabla_\theta^2 L_{PL}^{k+1} = \nabla_\theta^2 L_{PL}^k + C \succeq \nabla_\theta^2 L_{PL}^k$$

Consequently, by monotonicity of the matrix inverse, we have

$$\Gamma_{PL}^{k+1} = \left(\nabla_\theta^2 L_{PL}^{k+1}\right)^{-1} \preceq \left(\nabla_\theta^2 L_{PL}^k\right)^{-1} = \Gamma_{PL}^k$$

as we need. □

[17] The notation $A \preceq B$ means $B - A$ is positive semidefinite.

A.5 Proof of Theorem 2: Asymptotic variance under a Poincaré Inequality

Theorem 2 (Asymptotic variance under a Poincaré Inequality)**.** *Suppose the distribution p_{θ^*} satisfies a Poincaré inequality with constant C with respect to the α-weighted Block dynamics. Then the asymptotic variance of the α-weighted MPLE can be bounded as: $\Gamma_{PL} \preceq C\mathcal{I}^{-1}$ where $\mathcal{I}$ is the Fisher Information matrix (Definition 1).*

Proof. Let $\hat{\theta}_n \in \arg\min_\theta L_{PL}(\theta; \mathcal{S}_\mathcal{X}, \mathcal{S}_\mathcal{K})$ (Definition 2). Let d_Θ denote its dimensionality, i.e. $\hat{\theta}_n \in \mathbb{R}^{d_\Theta}$.

As a consequence of $\nabla^2_\theta L_{PL}(\theta) = \mathrm{Cov}(\nabla_\theta l_{PL}(\theta))$ (Lemma 2), we have:

$$\sqrt{|\mathcal{S}_\mathcal{X}|}(\hat{\theta}_n - \theta^*) \to \mathcal{N}(0, (\mathrm{Cov}(\nabla_\theta l_{PL}(\theta)))^{-1}) \tag{A.8}$$

Now we relate $\mathrm{Cov}(\nabla_\theta l_{PL}(\theta))$ to $\mathcal{I} = \mathrm{Cov}(\nabla_\theta l_{MLE}(\theta))$. Consider a test vector $v \in \mathbb{R}^{d_\Theta}$,

$$\begin{aligned}
&v^\top \mathrm{Cov}(\nabla_\theta l_{PL}(\theta)) v \\
&= v^\top \mathbb{E}_{S, x_S, x_{-S}} \nabla_\theta \log p_\theta(x_S|x_{-S}) \nabla_\theta \log p_\theta(x_S|x_{-S})^\top v \\
&= \mathbb{E}_{S, x_S, x_{-S}} (\nabla_\theta \log p_\theta(x_S|x_{-S})^\top v)^2 \\
&= \mathbb{E}_S \mathbb{E}_{x_{-S}} [\mathrm{Var}_{x_S|x_{-S}} (\nabla_\theta \log p_\theta(x_S|x_{-S})^\top v) + (\mathbb{E}_{x_S|x_{-S}} \nabla_\theta \log p_\theta(x_S|x_{-S})^\top v)^2] \\
&\geq \mathbb{E}_S \mathbb{E}_{x_{-S}} \mathrm{Var}_{x_S|x_{-S}} (\nabla_\theta \log p_\theta(x_S|x_{-S})^\top v) + 0 \\
&\geq \frac{1}{C} \mathrm{Var}_x (\nabla_\theta \log p_\theta(x)^\top v) \quad \text{(by Definition 6)} \\
&= \frac{1}{C} v^\top \mathcal{I} v
\end{aligned}$$

Therefore, we have

$$\mathrm{Cov}(\nabla_\theta l_{PL}(\theta)) \succeq \frac{1}{C} \mathcal{I}$$

Plugging into Equation (A.8), we obtain an upper bound on the asymptotic variance of our estimator:

$$\Gamma_{PL} \preceq C\mathcal{I}^{-1}$$

□

A.6 Proof of Theorem 3: Generalization bound for learning the joint distribution

We first state our overall structure of the proof of Theorem 3, and then state and prove the key lemmas mentioned therein.

Theorem 3 (Generalization bound for learning the joint distribution)**.** *Let $\hat{\theta} \coloneqq \hat{\theta}_{PL}$. Under Assumption 2 and Assumption 3 (in Appendix A.6), $\forall \epsilon > 0$, $\forall \delta \in (0,1)$, with probability at least $1-\delta$ over the randomness of $\mathcal{S}_{\mathcal{X}}$ and $\mathcal{S}_{\mathcal{K}}$, we have $D_{\mathrm{TV}}\left(p_{\hat{\theta}}, p\right) < \sqrt{\frac{1}{2}\bar{C}_{AT}(p_{\hat{\theta}})\left(A + B \cdot \ln\frac{1}{\beta} + \epsilon\right) + C}$ where $A = L_{PL}(\hat{\theta}; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}})$, $B = \sqrt{\frac{2^{3N}C_{\epsilon}(\Theta)}{|\mathcal{S}_{\mathcal{K}}|\cdot\delta}} + \sqrt{\frac{\ln\frac{8C_{\epsilon}(\Theta)}{\delta}}{2|\mathcal{S}_{\mathcal{X}}|}}$, and $C = \sqrt{\frac{|\Omega|^{3N}}{8\delta|\mathcal{S}_{\mathcal{X}}|}}$.*

Remark 3. *Building on Remark 2 in Appendix A.1, if additionally suppose the sample size $|\mathcal{S}_{\mathcal{X}}|$ and the number of mask configurations $|\mathcal{S}_{\mathcal{K}}|$ trained per sequence are large, then both terms on the right hand side of Theorem 3 are small, implying a generalization guarantee for learning the joint distribution.*

Proof. Theorem 3 follows by combining the following steps.

Step 1: relating closeness of the *conditional* distributions (i.e. the loss) to closeness of the *joint* distribution. The connection is established through the definition of the block-generalized approximate tensorization of entropy in Definition 8, by which we get:

$$D_{\mathrm{KL}}\left(\tilde{p}, p_{\hat{\theta}}\right) \leq \bar{C}_{AT}(p_{\hat{\theta}})\tilde{L}_{PL}(\hat{\theta})$$

The details are in Proposition 6 in Appendix A.6. By Pinsker's inequality, this implies

$$D_{\mathrm{TV}}\left(\tilde{p}, p_{\hat{\theta}}\right) \leq \sqrt{\frac{1}{2}D_{\mathrm{KL}}\left(\tilde{p}, p_{\hat{\theta}}\right)} \leq \sqrt{\frac{1}{2}\bar{C}_{AT}(p_{\hat{\theta}})\tilde{L}_{PL}(\hat{\theta})} \tag{A.9}$$

Step 2: generalization bound for learning the *conditional* distributions. We show that Assumption 2 and Assumption 3 imply a generalization guarantee for learning the *conditional* distributions from a finite sample of sequences and masked positions. We show that with probability at least $1-\frac{\delta}{2}$, we have

$$\left|L_{PL}(\hat{\theta}; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) - \tilde{L}_{PL}(\hat{\theta})\right| < \left(\sqrt{\frac{2^{3N}C_{\epsilon}(\Theta)}{|\mathcal{S}_{\mathcal{K}}|\cdot\delta}} + \sqrt{\frac{\ln\frac{8C_{\epsilon}(\Theta)}{\delta}}{2|\mathcal{S}_{\mathcal{X}}|}}\right) \cdot \ln\frac{1}{\beta} + \epsilon \tag{A.10}$$

Proof detail are in Corollary 2 in Appendix A.6.

Step 3: *empirical* joint distribution converges to *population* joint distribution. Proof is standard and details are in Lemma 6 in Appendix A.6. With probability at least $1-\frac{\delta}{2}$, we have

$$D_{\mathrm{TV}}\left(\tilde{p}, p\right) < \sqrt{\frac{|\Omega|^{3N}}{8\delta\,|\mathcal{S}_{\mathcal{X}}|}} \tag{A.11}$$

Step 4: union bound and triangle inequality By union bound, with probability at least $1-\delta$, both Equation (A.10) and Equation (A.11) hold. Therefore, putting together the previous steps, we get:

$$D_{\mathrm{TV}}\left(p_{\hat{\theta}}, p\right) \leq D_{\mathrm{TV}}\left(\tilde{p}, p_{\hat{\theta}}\right) + D_{\mathrm{TV}}\left(\tilde{p}, p\right) \quad \text{(by triangle inequality)}$$

$$\leq \sqrt{\frac{1}{2}\bar{C}_{AT}(p_{\hat{\theta}})\tilde{L}_{\mathrm{PL}}(\hat{\theta})} + \sqrt{\frac{|\Omega|^{3N}}{8\delta\,|\mathcal{S}_{\mathcal{X}}|}} \quad \text{(by Equation (A.9) and Equation (A.11))}$$

$$< \sqrt{\frac{1}{2}\bar{C}_{AT}(p_{\hat{\theta}})\left(L_{PL}(\hat{\theta}; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) + \left(\sqrt{\frac{2^{3N}C_{\epsilon}(\Theta)}{|\mathcal{S}_{\mathcal{K}}|\cdot\delta}} + \sqrt{\frac{\ln\frac{8C_{\epsilon}(\Theta)}{\delta}}{2|\mathcal{S}_{\mathcal{X}}|}}\right) \cdot \ln\frac{1}{\beta} + \epsilon\right)} + \sqrt{\frac{|\Omega|^{3N}}{8\delta\,|\mathcal{S}_{\mathcal{X}}|}}$$

(by Equation (A.10))

□

Proposition 6. $D_{\mathrm{KL}}(p, p_\theta) \leq \bar{C}_{AT}(p_\theta) L_{PL}(\theta)$ *and* $D_{\mathrm{KL}}(\tilde{p}, p_\theta) \leq \bar{C}_{AT}(p_\theta) \tilde{L}_{PL}(\theta)$

Proof. By definition of block-generalized approximate tensorization of entropy in Definition 8

$$\begin{aligned} D_{\mathrm{KL}}(p, p_\theta) &\leq \bar{C}_{AT}(p_\theta) \mathbb{E}_{X \sim p} \left[\mathbb{E}_{K \subset [N]} \left[D_{\mathrm{KL}}\left(p(\cdot \mid X_{-K}), p_\theta(\cdot \mid X_{-K})\right) \right] \right] \\ &= \bar{C}_{AT}(p_\theta) L_{\mathrm{PL}}(\theta) \end{aligned}$$

Likewise the latter holds when we replace p with $\tilde{p}$.

□

Proposition 7 (KL is bounded)**.** *Under Assumption 2,*

$$D_{\mathrm{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right) \in [0, \ln \frac{1}{\beta}]$$

Proof. By definition of D_{KL},

$$\begin{aligned} 0 \leq D_{\mathrm{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right) &= \sum_{X_K \in \Omega^{|K|}} \tilde{p}(X_K|X_{-K}) \ln \frac{\tilde{p}(X_K|X_{-K})}{p_\theta(X_K|X_{-K})} \\ &\leq \sum_{X_K \in \Omega^{|K|}} \tilde{p}(X_K|X_{-K}) \ln \frac{1}{p_\theta(X_K|X_{-K})} \\ &\leq \sum_{X_K \in \Omega^{|K|}} \tilde{p}(X_K|X_{-K}) \ln \frac{1}{\beta} \quad \text{(by Assumption 2)} \\ &= \ln \frac{1}{\beta} \end{aligned}$$

□

Lemma 4 (Hoeffding's inequality (Hoeffding, 1994))**.** *Let $Y_1, \cdots, Y_n$ be independent random variables such that $a \leq Y_i \leq b$ almost surely. Consider the sum of these random variables, $S_n = Y_1 + \cdots + Y_n$ whose expectation is $\mathbb{E}[S_n]$. Then, $\forall t > 0$, with probability at least $1 - 2e^{-\frac{2t^2}{n(b-a)^2}}$, we have*

$$|S_n - \mathbb{E}[S_n]| < t$$

Proof. See Hoeffding (1994). □

Lemma 5 (Point-wise generalization bound for learning conditional distributions)**.** *Fix a $\theta \in \Theta$ satisfying Assumption 2. $\forall \epsilon > 0, t > 0$, with probability at least $1 - \frac{2^{N-2}}{\epsilon^2 |\mathcal{S}_\mathcal{K}|} - 2e^{-\frac{2t^2}{|\mathcal{S}_\mathcal{X}| \cdot \left(\ln \frac{1}{\beta}\right)^2}}$, we have*

$$\left| L_{PL}(\theta; \mathcal{S}_\mathcal{X}, \mathcal{S}_\mathcal{K}) - \tilde{L}_{PL}(\theta) \right| < 2^N \epsilon \cdot \ln \frac{1}{\beta} + \frac{t}{|\mathcal{S}_\mathcal{X}|}$$

Proof. **Step 1: concentration over masked configurations $\mathcal{S}_\mathcal{K}$.**

We first prove that $L_{\mathrm{PL}}(\theta; \mathcal{S}_\mathcal{X}, \mathcal{S}_\mathcal{K})$ (Definition 2) converges to the expectation over masked positions K as $|\mathcal{S}_\mathcal{K}|$ increases. [18]

Denote

$$f(X) \coloneqq \mathbb{E}_K \left[D_{\mathrm{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right) \right] \tag{A.12}$$

[18] Note that the terms $D_{\mathrm{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right)$ are (generally) not independent for different K. Besides, the terms $\sum_{K \in \mathcal{S}_\mathcal{K}} D_{\mathrm{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right)$ are (generally) not independent for different $\mathcal{S}_\mathcal{K}$.

Then the expectation of $L_{\text{PL}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}})$ over the randomness of $\mathcal{S}_{\mathcal{K}}$ is:

$$\begin{aligned}
\mathbb{E}_{\{\mathcal{S}_{\mathcal{K}}(j) \mid j \in [|\mathcal{S}_{\mathcal{X}}|]\}} \left[L_{\text{PL}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}})\right] &= \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{j, X \in \mathcal{S}_{\mathcal{X}}} \frac{1}{|\mathcal{S}_{\mathcal{K}}|} \mathbb{E}_{\mathcal{S}_{\mathcal{K}}(j)} \left[\sum_{K \in \mathcal{S}_{\mathcal{K}}(j)} D_{\text{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right) \right] \\
&= \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{j, X \in \mathcal{S}_{\mathcal{X}}} \mathbb{E}_K \left[D_{\text{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right)\right] \\
&= \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{X \in \mathcal{S}_{\mathcal{X}}} f(X) \qquad \text{(A.13)}
\end{aligned}$$

Moreover, for each K, the empirical probability $p_S(K)$ of training on $\tilde{p}(\cdot|X_{-K})$ converges to the true probability $p(K)$ as $|\mathcal{S}_{\mathcal{K}}|$ increases, because the count, $p_S(K)\,|\mathcal{S}_{\mathcal{K}}|$ follows the binomial distribution

$$\text{Binomial}(|\mathcal{S}_{\mathcal{K}}|\,, p(K))$$

More specifically, by Chebyshev's inequality, $\forall \epsilon > 0$:

$$\begin{aligned}
\mathbb{P}\left\{|p_S(K) - p(K)| \geq \epsilon\right\} &= \mathbb{P}\left\{p_S(K)\,|\mathcal{S}_{\mathcal{K}}| - p(K)\,|\mathcal{S}_{\mathcal{K}}| \geq \epsilon\,|\mathcal{S}_{\mathcal{K}}|\right\} \\
&\leq \frac{\text{Var}\left(p_S(K)\,|\mathcal{S}_{\mathcal{K}}|\right)}{\epsilon^2\,|\mathcal{S}_{\mathcal{K}}|^2} \quad \text{(Chebyshev's inequality)} \\
&= \frac{|\mathcal{S}_{\mathcal{K}}|\,p(K)(1-p(K))}{\epsilon^2\,|\mathcal{S}_{\mathcal{K}}|^2} \quad \text{(since } p_S(K)\,|\mathcal{S}_{\mathcal{K}}| \sim \text{Binomial}(|\mathcal{S}_{\mathcal{K}}|\,, p(K))\text{)} \\
&= \frac{p(K)(1-p(K))}{\epsilon^2\,|\mathcal{S}_{\mathcal{K}}|} \\
&\leq \frac{1}{4\epsilon^2\,|\mathcal{S}_{\mathcal{K}}|}
\end{aligned}$$

Applying union bound over $K \in \{0,1\}^N$,

$$\mathbb{P}\left\{|p_S(K) - p(K)| < \epsilon,\, \forall K \in \{0,1\}^N\right\} \geq 1 - \frac{2^N}{4\epsilon^2\,|\mathcal{S}_{\mathcal{K}}|} = 1 - \frac{2^{N-2}}{\epsilon^2\,|\mathcal{S}_{\mathcal{K}}|} \qquad \text{(A.14)}$$

Plugging into Equation (A.12) and Equation (A.13), we get with probability at least $1 - \frac{2^{N-2}}{\epsilon^2|\mathcal{S}_{\mathcal{K}}|}$,

$$\begin{aligned}
&\left| L_{\text{PL}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) - \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{X \in \mathcal{S}_{\mathcal{X}}} f(X) \right| \\
&= \left| \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{j, X \in \mathcal{S}_{\mathcal{X}}} \frac{1}{|\mathcal{S}_{\mathcal{K}}(j)|} \sum_{K \in \mathcal{S}_{\mathcal{K}}(j)} D_{\text{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right) - \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{X \in \mathcal{S}_{\mathcal{X}}} \mathbb{E}_K \left[D_{\text{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right)\right] \right| \\
&\leq \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{j, X \in \mathcal{S}_{\mathcal{X}}} \sum_{K \in \{0,1\}^N} |p_S(K) - p(K)| \cdot D_{\text{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right) \quad \text{(triangle inequality)} \\
&< \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{j, X \in \mathcal{S}_{\mathcal{X}}} \sum_{K \in \{0,1\}^N} \epsilon \cdot D_{\text{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right) \quad \text{(by Equation (A.14))} \\
&\leq \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{j, X \in \mathcal{S}_{\mathcal{X}}} \sum_{K \in \{0,1\}^N} \epsilon \cdot \ln \frac{1}{\beta} \quad \text{(by Proposition 7)} \\
&= \frac{1}{|\mathcal{S}_{\mathcal{X}}|} \sum_{j, X \in \mathcal{S}_{\mathcal{X}}} 2^N \epsilon \cdot \ln \frac{1}{\beta} \quad = 2^N \epsilon \cdot \ln \frac{1}{\beta} \qquad \text{(A.15)}
\end{aligned}$$

Step 2: concentration over sequences X in training data.

Recall $f(X)$ defined in Equation (A.12).

In the following we prove that $\frac{1}{|\mathcal{S}_\mathcal{X}|}\sum_{X\in\mathcal{S}_\mathcal{X}} f(X)$ converges to:

$$\begin{aligned}\mathbb{E}\left[f(X)\right] &= \mathbb{E}_{X\sim\tilde{p}}\left[\mathbb{E}_{\mathcal{S}_\mathcal{K}(X)\sim\text{size-}k\text{ subsets of }[N]}\left[\frac{1}{|\mathcal{S}_\mathcal{K}(X)|}\sum_{K\in\mathcal{S}_\mathcal{K}(X)} D_{\mathrm{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right)\right]\right] \\ &= \mathbb{E}_{X\sim\tilde{p}}\left[\mathbb{E}_{K\subset[N],|K|=k}\left[D_{\mathrm{KL}}\left(\tilde{p}(\cdot|X_{-K}), p_\theta(\cdot|X_{-K})\right)\right]\right] \\ &= \tilde{L}_{\text{PL}}(\theta)\end{aligned}$$

Note that $f(X)\in[0,\ln\frac{1}{\beta}]$ by Proposition 7.

Thus, applying Hoeffding's inequality (Lemma 4), $\forall t > 0$, with probability at least $1 - 2e^{-\frac{2t^2}{|\mathcal{S}_\mathcal{X}|\cdot\left(\ln\frac{1}{\beta}\right)^2}}$, we have

$$\left|\frac{1}{|\mathcal{S}_\mathcal{X}|}\sum_{X\in\mathcal{S}_\mathcal{X}} f(X) - \mathbb{E}\left[f(X)\right]\right| < \frac{t}{|\mathcal{S}_\mathcal{X}|} \tag{A.16}$$

Step 3: combining results: concentration over both masks K and sequences X.

By union bound, with probability at least

$$1 - \frac{2^{N-2}}{\epsilon^2\,|\mathcal{S}_\mathcal{K}|} - 2e^{-\frac{2t^2}{|\mathcal{S}_\mathcal{X}|\cdot\left(\ln\frac{1}{\beta}\right)^2}}$$

both Equation (A.15) and Equation (A.16) hold, giving us,

$$\begin{aligned}&\left|L_{\text{PL}}(\theta;\mathcal{S}_\mathcal{X},\mathcal{S}_\mathcal{K}) - \tilde{L}_{\text{PL}}(\theta)\right| \\ \le&\left|L_{\text{PL}}(\theta;\mathcal{S}_\mathcal{X},\mathcal{S}_\mathcal{K}) - \frac{1}{|\mathcal{S}_\mathcal{X}|}\sum_{X\in\mathcal{S}_\mathcal{X}} f(X)\right| + \left|\frac{1}{|\mathcal{S}_\mathcal{X}|}\sum_{X\in\mathcal{S}_\mathcal{X}} f(X) - \tilde{L}_{\text{PL}}(\theta)\right| \quad \text{(triangle inequality)} \\ <& 2^N\epsilon\cdot\ln\frac{1}{\beta} + \frac{t}{|\mathcal{S}_\mathcal{X}|}\end{aligned}$$

□

Remark 4. *The two terms in the bound given by Lemma 5, i.e. $2^N\epsilon\cdot\ln\frac{1}{\beta}$ and $\frac{t}{|\mathcal{S}_\mathcal{X}|}$, can be controlled by setting appropriate ϵ and t based on $|\mathcal{S}_\mathcal{K}|$ and $|\mathcal{S}_\mathcal{X}|$, respectively. These two terms can reduce by increasing $|\mathcal{S}_\mathcal{K}|$ and $|\mathcal{S}_\mathcal{X}|$, respectively, as we will show in the subsequent corollary. This is intuitive: we expect a smaller generalization gap when the model is trained on more mask configurations for each sequence, and when more sequences are included in the data. The first term grows with N — this is also intuitive: when the sequences are longer, it is natural to require observing more mask configurations.*

Corollary 1 (Point-wise generalization bound for learning conditional distributions, special case)**.** *Fix a $\theta\in\Theta$ satisfying Assumption 2. with probability at least $1-\delta$, we have*

$$\left|L_{PL}(\theta;\mathcal{S}_\mathcal{X},\mathcal{S}_\mathcal{K}) - \tilde{L}_{PL}(\theta)\right| < \left(\sqrt{\frac{2^{3N-1}}{|\mathcal{S}_\mathcal{K}|\cdot\delta}} + \sqrt{\frac{\ln\frac{4}{\delta}}{2|\mathcal{S}_\mathcal{X}|}}\right)\cdot\ln\frac{1}{\beta}$$

Proof. Apply Lemma 5 with ϵ and t satisfying

$$\begin{aligned}\frac{\delta}{2} &= \frac{2^{N-2}}{\epsilon^2\,|\mathcal{S}_\mathcal{K}|} \\ \frac{\delta}{2} &= 2e^{-\frac{2t^2}{|\mathcal{S}_\mathcal{X}|\cdot\left(\ln\frac{1}{\beta}\right)^2}}\end{aligned}$$

which solves to

$$\epsilon = \sqrt{\frac{2^{N-1}}{|\mathcal{S}_{\mathcal{K}}| \cdot \delta}}$$
$$t = \sqrt{\frac{\ln \frac{4}{\delta} \cdot |\mathcal{S}_{\mathcal{X}}|}{2}} \ln \frac{1}{\beta}$$

Therefore, by Lemma 5, with probability at least $1 - \delta$, we have

$$\begin{aligned} &\left| L_{\text{PL}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) - \tilde{L}_{\text{PL}}(\theta) \right| \\ &< 2^N \epsilon \cdot \ln \frac{1}{\beta} + \frac{t}{|\mathcal{S}_{\mathcal{X}}|} \\ &= \left(\sqrt{\frac{2^{3N-1}}{|\mathcal{S}_{\mathcal{K}}| \cdot \delta}} + \sqrt{\frac{\ln \frac{4}{\delta}}{2|\mathcal{S}_{\mathcal{X}}|}} \right) \cdot \ln \frac{1}{\beta} \end{aligned}$$

□

Corollary 2 (Uniform convergence generalization bound for learning conditional distributions)**.** *Under Assumption 2 and Assumption 3, $\forall \delta \in (0, 1)$, $\forall \epsilon > 0$, with probability at least $1 - \delta$, we have*

$$\left| L_{PL}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) - \tilde{L}_{PL}(\theta) \right| < \left(\sqrt{\frac{2^{3N-1} C_\epsilon(\Theta)}{|\mathcal{S}_{\mathcal{K}}| \cdot \delta}} + \sqrt{\frac{\ln \frac{4C_\epsilon(\Theta)}{\delta}}{2|\mathcal{S}_{\mathcal{X}}|}} \right) \cdot \ln \frac{1}{\beta} + \epsilon$$

Proof. By Assumption 3, let $C_\epsilon(\Theta)$ denote the complexity of parameter space Θ, with the corresponding partition $\text{Par}_\epsilon(\Theta) = \{\Theta_1, \cdots, \Theta_{C_\epsilon(\Theta)}\}$. Moreover, for each $i \in [C_\epsilon(\Theta)]$, arbitrarily select any point $\theta_i^* \in \Theta_i$ (as a "representative" of that region of the parameter space). Let the set of "representative points" be $\Theta^* = \{\theta_i^* \mid i \in [C_\epsilon(\Theta)]\}$.

By Corollary 1, fixing any $\theta \in \Theta$ satisfying Assumption 2, then with probability at least $1 - \frac{\delta}{C_\epsilon(\Theta)}$, we have

$$\left| L_{\text{PL}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) - \tilde{L}_{\text{PL}}(\theta) \right| < \left(\sqrt{\frac{2^{3N-1} C_\epsilon(\Theta)}{|\mathcal{S}_{\mathcal{K}}| \cdot \delta}} + \sqrt{\frac{\ln \frac{4C_\epsilon(\Theta)}{\delta}}{2|\mathcal{S}_{\mathcal{X}}|}} \right) \cdot \ln \frac{1}{\beta}$$

Applying union bound over $\theta_i^* \in \Theta^*$, since $|\Theta^*| = C_\epsilon(\Theta)$, with probability at least $1 - \delta$,

$$\forall i \in [C_\epsilon(\Theta)], \quad \left| L_{\text{PL}}(\theta_i^*; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) - \tilde{L}_{\text{PL}}(\theta_i^*) \right| < \left(\sqrt{\frac{2^{3N-1} C_\epsilon(\Theta)}{|\mathcal{S}_{\mathcal{K}}| \cdot \delta}} + \sqrt{\frac{\ln \frac{4C_\epsilon(\Theta)}{\delta}}{2|\mathcal{S}_{\mathcal{X}}|}} \right) \cdot \ln \frac{1}{\beta} \tag{A.17}$$

Finally, by Assumption 3, $\forall \theta \in \Theta$, there exists $i \in [C_\epsilon(\Theta)]$ such that $\theta \in \Theta_i$ (i.e. θ falls into that partition), and

$$\begin{aligned} \left| \tilde{L}_{\text{PL}}(\theta) - \tilde{L}_{\text{PL}}(\theta_i^*) \right| &\leq \frac{\epsilon}{2} \\ |L_{\text{PL}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) - L_{\text{PL}}(\theta_i^*; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}})| &\leq \frac{\epsilon}{2} \end{aligned} \tag{A.18}$$

Combining Equation (A.17) and Equation (A.18) gives

$$\begin{aligned}
&\left|L_{\text{PL}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) - \tilde{L}_{\text{PL}}(\theta)\right| \\
&\leq \left|L_{\text{PL}}(\theta; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) - L_{\text{PL}}(\theta_i^*; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}})\right| + \left|L_{\text{PL}}(\theta_i^*; \mathcal{S}_{\mathcal{X}}, \mathcal{S}_{\mathcal{K}}) - \tilde{L}_{\text{PL}}(\theta_i^*)\right| \\
&\quad + \left|\tilde{L}_{\text{PL}}(\theta_i^*) - \tilde{L}_{\text{PL}}(\theta)\right| \quad \text{(by triangle inequality)} \\
&< \frac{\epsilon}{2} + \left(\sqrt{\frac{2^{3N-1} C_\epsilon(\Theta)}{|\mathcal{S}_{\mathcal{K}}| \cdot \delta}} + \sqrt{\frac{\ln \frac{4C_\epsilon(\Theta)}{\delta}}{2|\mathcal{S}_{\mathcal{X}}|}}\right) \cdot \ln \frac{1}{\beta} + \frac{\epsilon}{2} \quad \text{(by Equation (A.17) and Equation (A.18))} \\
&= \left(\sqrt{\frac{2^{3N-1} C_\epsilon(\Theta)}{|\mathcal{S}_{\mathcal{K}}| \cdot \delta}} + \sqrt{\frac{\ln \frac{4C_\epsilon(\Theta)}{\delta}}{2|\mathcal{S}_{\mathcal{X}}|}}\right) \cdot \ln \frac{1}{\beta} + \epsilon
\end{aligned}$$

□

Lemma 6 (Empirical PMF converges to population PMF)**.** *Let the population joint distribution p, the finite set of training samples $\mathcal{S}_{\mathcal{X}}$ drawn IID from p, and the (noisy) empirical joint distribution $\tilde{p}$ on $\mathcal{S}_{\mathcal{X}}$ be defined as in Section 2. Then, $\forall \delta > 0$, with probability at least $1 - \delta$, we have*

$$D_{\text{TV}}(\tilde{p}, p) < \sqrt{\frac{|\Omega|^{3N}}{16\delta\, |\mathcal{S}_{\mathcal{X}}|}}$$

Proof. $\forall X \in \Omega^N$, the number of times that X appears in the training data $\mathcal{S}_{\mathcal{X}}$ follows the binomial distribution

$$\tilde{p}\, |\mathcal{S}_{\mathcal{X}}| \sim \text{Binomial}(|\mathcal{S}_{\mathcal{X}}|, p(X))$$

with mean $|\mathcal{S}_{\mathcal{X}}|\, p(X)$ and variance $|\mathcal{S}_{\mathcal{X}}|\, p(X)(1 - p(X))$. Hence, by Chebyshev's inequality, $\forall \epsilon > 0$

$$\begin{aligned}
\mathbb{P}\left\{|\tilde{p}(X) - p(X)| \geq \epsilon\right\} &= \mathbb{P}\left\{\tilde{p}(X)\, |\mathcal{S}_{\mathcal{X}}| - p(X)\, |\mathcal{S}_{\mathcal{X}}| \geq \epsilon\, |\mathcal{S}_{\mathcal{X}}|\right\} \\
&\leq \frac{\text{Var}\left(\tilde{p}(X)\, |\mathcal{S}_{\mathcal{X}}|\right)}{\epsilon^2\, |\mathcal{S}_{\mathcal{X}}|^2} \quad \text{(Chebyshev's inequality)} \\
&= \frac{|\mathcal{S}_{\mathcal{X}}|\, p(X)(1 - p(X))}{\epsilon^2\, |\mathcal{S}_{\mathcal{X}}|^2} \quad \text{(since } \tilde{p}(X)\, |\mathcal{S}_{\mathcal{X}}| \sim \text{Binomial}(|\mathcal{S}_{\mathcal{X}}|, p(X))\text{)} \\
&= \frac{p(X)(1 - p(X))}{\epsilon^2\, |\mathcal{S}_{\mathcal{X}}|} \\
&\leq \frac{1}{4\epsilon^2\, |\mathcal{S}_{\mathcal{K}}|}
\end{aligned}$$

Applying union bound over $X \in \Omega^N$,

$$\mathbb{P}\left\{|\tilde{p}(X) - p(X)| < \epsilon,\ \forall X \in \Omega^N\right\} \geq 1 - \frac{|\Omega|^N}{4\epsilon^2\, |\mathcal{S}_{\mathcal{X}}|} \tag{A.19}$$

Hence, we get with probability at least $1 - \frac{|\Omega|^N}{4\epsilon^2 |\mathcal{S}_{\mathcal{X}}|}$,

$$D_{\text{TV}}(\tilde{p}, p) = \frac{1}{2} \sum_{X \in \Omega^N} |\tilde{p}(X) - p(X)| < \frac{1}{2} \sum_{X \in \Omega^N} \epsilon = \frac{1}{2}\, |\Omega|^N\, \epsilon \tag{A.20}$$

Finally, aligning the probabilities: solving for

$$\delta = \frac{|\Omega|^N}{4\epsilon^2\, |\mathcal{S}_{\mathcal{X}}|}$$

gives

$$\epsilon = \sqrt{\frac{|\Omega|^N}{4\delta\,|\mathcal{S}_{\mathcal{X}}|}}$$

Therefore, by Equation (A.19), with probability at least $1-\delta$, we have

$$D_{\mathrm{TV}}\left(\tilde{p}, p\right) < \frac{1}{2}\,|\Omega|^N\,\epsilon = \sqrt{\frac{|\Omega|^{3N}}{16\delta\,|\mathcal{S}_{\mathcal{X}}|}}$$

□

A.7 Proof of Proposition 8: Modes of the strongly ferromagnetic Ising model

This section provides additional information for the discussion under Assumption 4 in Section 3.2.

Proposition 8 (Modes of the strongly ferromagnetic Ising model)**.** *On Ising model G in Equation* (3) *under Assumption 4, the high-probability regions $\mathcal{R}_1$ and $\mathcal{R}_{-1}$ defined in Equation* (4) *and Equation* (5) *satisfy*

1. $\forall \boldsymbol{x} \in \mathcal{R}_1$, $\forall \boldsymbol{y} \in \mathcal{R}_{-1}$, $\forall \boldsymbol{z} \in \{-1,1\}^N \backslash \mathcal{R}_1 \backslash \mathcal{R}_{-1}$, $p_G(\boldsymbol{x}) > p_G(\boldsymbol{y}) > e^{2J_0} p_G(\boldsymbol{z})$
2. *There exists a bijection $f : \mathcal{R}_1 \mapsto \mathcal{R}_{-1}$ such that* $\forall \boldsymbol{x} \in \mathcal{R}_1$, $p_G(\boldsymbol{x}) = e^{2h_G} p_G(f(\boldsymbol{x}))$

Proof. $\forall \boldsymbol{x} \in \mathcal{R}_1, \forall \boldsymbol{y} \in \mathcal{R}_{-1}$,

$$
\begin{aligned}
\frac{p_G(\boldsymbol{x})}{p_G(\boldsymbol{y})} &= \frac{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{x}_i + \sum_{i\neq j \in C_G \subset [N]} J \mathbf{x}_i \mathbf{x}_j\right)}{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{y}_i + \sum_{i\neq j \in C_G \subset [N]} J \mathbf{y}_i \mathbf{y}_j\right)} \\
&= \frac{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{x}_i\right)}{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{y}_i\right)} \quad (\text{since } \mathbf{x}_i\mathbf{x}_j = \mathbf{y}_i\mathbf{y}_j = 1\, \forall \boldsymbol{x} \in \mathcal{R}_1, \forall \boldsymbol{y} \in \mathcal{R}_{-1}) \\
&= \frac{\exp\left(\sum_{i\in C_G} \boldsymbol{h}_i \mathbf{x}_i + \sum_{i\notin C_G} \boldsymbol{h}_i \mathbf{x}_i\right)}{\exp\left(\sum_{i\in C_G} \boldsymbol{h}_i \mathbf{y}_i + \sum_{i\notin C_G} \boldsymbol{h}_i \mathbf{y}_i\right)} \\
&= \frac{\exp\left(\sum_{i\in C_G} \boldsymbol{h}_i + \sum_{i\notin C_G} \boldsymbol{h}_i \mathbf{x}_i\right)}{\exp\left(-\sum_{i\in C_G} \boldsymbol{h}_i + \sum_{i\notin C_G} \boldsymbol{h}_i \mathbf{y}_i\right)} \quad (\text{since } \boldsymbol{x} \in \mathcal{R}_1, \boldsymbol{y} \in \mathcal{R}_{-1}) \\
&\geq \frac{\exp\left(\sum_{i\in C_G} \boldsymbol{h}_i - \sum_{i\notin C_G} |\boldsymbol{h}_i|\right)}{\exp\left(-\sum_{i\in C_G} \boldsymbol{h}_i + \sum_{i\notin C_G} |\boldsymbol{h}_i|\right)} \quad (\text{since } \mathbf{x}_i, \mathbf{y}_i \in \pm 1) \\
&= \exp\left(2\sum_{i\in C_G} \boldsymbol{h}_i - 2\sum_{i\notin C_G} |\boldsymbol{h}_i|\right) \\
&> \exp(0) = 1 \quad (\text{by Assumption 4})
\end{aligned}
$$

$\forall \boldsymbol{y} \in \mathcal{R}_{-1}, \forall \boldsymbol{z} \in \{-1,1\}^N \backslash \mathcal{R}_1 \backslash \mathcal{R}_{-1}$,

$$
\begin{aligned}
\frac{p_G(\boldsymbol{y})}{p_G(\boldsymbol{z})} &= \frac{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{y}_i + \sum_{i\neq j \in C_G \subset [N]} J \mathbf{y}_i \mathbf{y}_j\right)}{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{z}_i + \sum_{i\neq j \in C_G \subset [N]} J \mathbf{z}_i \mathbf{z}_j\right)} \\
&= \frac{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{y}_i + \sum_{i\neq j \in C_G \subset [N]} J\right)}{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{z}_i + \sum_{i\neq j \in C_G \subset [N]} J \mathbf{z}_i \mathbf{z}_j\right)} \quad (\text{since } \mathbf{y}_i\mathbf{y}_j = 1\, \forall \boldsymbol{y} \in \mathcal{R}_{-1}) \\
&\geq \frac{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{y}_i + \sum_{i\neq j \in C_G \subset [N]} J\right)}{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{z}_i + \sum_{i\neq j \in C_G \subset [N]} J - 2(|C_G|-1)J\right)} \\
&\quad (\text{since } \boldsymbol{z} \in \{-1,1\}^N \backslash \mathcal{R}_1 \backslash \mathcal{R}_{-1} \text{ and consider min edge number in bipartite graph}) \\
&= \frac{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{y}_i\right)}{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{z}_i - 2(|C_G|-1)J\right)} \geq \frac{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{y}_i\right)}{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{z}_i - 2J\right)} \geq \frac{\exp(-\|\boldsymbol{h}\|_1)}{\exp(\|\boldsymbol{h}\|_1 - 2J)} \\
&= \exp(2(J - \|\boldsymbol{h}\|_1)) \geq \exp(2J_0) \quad (\text{by Assumption 4})
\end{aligned}
$$

For part 2, let $f : \mathcal{R}_1 \mapsto \mathcal{R}_{-1}$ be defined as

$$
\forall \boldsymbol{x} \in \mathcal{R}_1, \quad f(\boldsymbol{x})_i = \begin{cases} -1, & \text{if } i \in C_G \\ \boldsymbol{x}_i, & \text{if } i \notin C_G \end{cases}
$$

Let $\boldsymbol{w} \coloneqq f(\boldsymbol{x})$. Then,

$$
\begin{aligned}
\frac{p_G(\boldsymbol{x})}{p_G(\boldsymbol{w})} &= \frac{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{x}_i + \sum_{i\neq j\in C_G\subset[N]} J\mathbf{x}_i\mathbf{x}_j\right)}{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{w}_i + \sum_{i\neq j\in C_G\subset[N]} J\mathbf{w}_i\mathbf{w}_j\right)} \\
&= \frac{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{x}_i\right)}{\exp\left(\sum_{i\in[N]} \boldsymbol{h}_i \mathbf{w}_i\right)} \quad \text{(since } \mathbf{x}_i\mathbf{x}_j = \mathbf{w}_i\mathbf{w}_j = 1\, \forall \boldsymbol{x} \in \mathcal{R}_1, \forall \boldsymbol{w} \in \mathcal{R}_{-1}) \\
&= \frac{\exp\left(\sum_{i\in C_G} \boldsymbol{h}_i \mathbf{x}_i + \sum_{i\notin C_G} \boldsymbol{h}_i \mathbf{x}_i\right)}{\exp\left(\sum_{i\in C_G} \boldsymbol{h}_i \mathbf{w}_i + \sum_{i\notin C_G} \boldsymbol{h}_i \mathbf{w}_i\right)} \\
&= \frac{\exp\left(\sum_{i\in C_G} \boldsymbol{h}_i + \sum_{i\notin C_G} \boldsymbol{h}_i \mathbf{x}_i\right)}{\exp\left(-\sum_{i\in C_G} \boldsymbol{h}_i + \sum_{i\notin C_G} \boldsymbol{h}_i \mathbf{w}_i\right)} \quad \text{(since } \boldsymbol{x} \in \mathcal{R}_1, \boldsymbol{w} \in \mathcal{R}_{-1}) \\
&= \frac{\exp\left(\sum_{i\in C_G} \boldsymbol{h}_i\right)}{\exp\left(-\sum_{i\in C_G} \boldsymbol{h}_i\right)} \quad \text{(since } \boldsymbol{w}_i = \boldsymbol{x}_i,\ \forall i \notin C_G) \\
&= \exp\left(2\sum_{i\in C_G} \boldsymbol{h}_i\right) \\
&= \exp\left(2h_G\right) \quad \text{(by Assumption 4)}
\end{aligned}
$$

□

A.8 PROOF OF PROPOSITION 3: k-GIBBS SAMPLER CAN REACH THE MODE FAST

Proposition 3 (k-Gibbs sampler sampling can reach the mode fast)**.** *On Ising model G in Equation* (3) *under Assumption 4, with any initial* $\boldsymbol{X}^{(0)}$*,* $\forall\delta \in (0,1)$*, with probability at least* $1-\delta$*, after* $T := \left\lceil \log_{c_{\mathcal{R}_1}} \delta \right\rceil$ *steps of* k***-Gibbs sampler*** *(Gibbs sampler 1) with* $k \geq |C_G|$*, we have* $\{\boldsymbol{X}^{(t)} | t \in [T]\} \cap \mathcal{R}_1 \neq \emptyset$ *in which the constant* $c_{\mathcal{R}_1} \in (0,1)$*,* $c_{\mathcal{R}_1} := 1 - \frac{\binom{N-|C_G|}{k-|C_G|}}{\binom{N}{k}} \frac{e^{2(J_0+h_G)}}{e^{2(J_0+h_G)}+e^{2J_0}+2^{|C_G|}-2}$

Proof. At any step, let K (with $|K| = k$) denote the set of coordinates to re-sample. We first consider the probability of $C_G \subset K$, which allows the whole C_G to be updated jointly:

$$\mathbb{P}\{C_G \subset K\} = \frac{\binom{N-|C_G|}{k-|C_G|}}{\binom{N}{k}} \tag{A.21}$$

$\forall t \in \mathbb{N}, \forall \boldsymbol{X}^{(t)} \in \{-1,1\}^N$, and $K \in [N]$ such that $|K| = k$ and $C_G \subset K$, consider $X_K^{(t+1)} \sim p_G(\cdot \mid X_{-K}^{(t)})$. There are three cases (a partition of all possibilities):

1. $\boldsymbol{X}^{(t+1)} \in \mathcal{R}_1$
2. $\boldsymbol{X}^{(t+1)} \in \mathcal{R}_{-1}$
3. $\boldsymbol{X}^{(t+1)} \in \{-1,1\}^N \backslash \mathcal{R}_1 \backslash \mathcal{R}_{-1}$

Then by Proposition 8 we show that Case 1 occurs with probability at least a (not arbitrarily small) constant,

$$\mathbb{P}\left\{\boldsymbol{X}^{(t+1)} \in \mathcal{R}_1\right\} = e^{2h_G}\mathbb{P}\left\{\boldsymbol{X}^{(t+1)} \in \mathcal{R}_{-1}\right\}$$

$$\frac{\mathbb{P}\left\{\boldsymbol{X}^{(t+1)} \in \mathcal{R}_{-1}\right\}}{\mathbb{P}\left\{\boldsymbol{X}^{(t+1)} \in \{-1,1\}^N \backslash \mathcal{R}_1 \backslash \mathcal{R}_{-1}\right\}} \geq e^{2J_0} \frac{|\mathcal{R}_{-1}|}{|\{-1,1\}^N \backslash \mathcal{R}_1 \backslash \mathcal{R}_{-1}|} = \frac{e^{2J_0}}{2^{|C_G|}-2}$$

Since the probabilities of the three cases sum up to 1,

$$\mathbb{P}\left\{\boldsymbol{X}^{(t+1)} \in \mathcal{R}_1\right\} \geq \frac{e^{2(J_0+h_G)}}{e^{2(J_0+h_G)} + e^{2J_0} + 2^{|C_G|} - 2}$$

Therefore, $\forall t \in \mathbb{N}, \forall \boldsymbol{X}^{(t)} \in \{-1,1\}^N$, combined with Equation (A.21),

$$\mathbb{P}\left\{\boldsymbol{X}^{(t+1)} \in \mathcal{R}_1\right\} \geq \mathbb{P}\left\{C_G \subset K, \boldsymbol{X}^{(t+1)} \in \mathcal{R}_1\right\} \geq \frac{\binom{N-|C_G|}{k-|C_G|}}{\binom{N}{k}} \frac{e^{2(J_0+h_G)}}{e^{2(J_0+h_G)} + e^{2J_0} + 2^{|C_G|} - 2} := 1 - c_{\mathcal{R}_1}$$

i.e. let constant $c_{\mathcal{R}_1}$ denote the above upper bound of $\mathbb{P}\left\{\boldsymbol{X}^{(t+1)} \notin \mathcal{R}_1\right\}$. Then

$$\mathbb{P}\left\{\{\boldsymbol{X}^{(t)} | t \in [T]\} \cap \mathcal{R}_1 = \emptyset\right\} \leq c_{\mathcal{R}_1}^T$$

Therefore, when $T \geq \log_{c_{\mathcal{R}_1}} \delta$,

$$\mathbb{P}\left\{\{\boldsymbol{X}^{(t)} | t \in [T]\} \cap \mathcal{R}_1 = \emptyset\right\} \leq \delta$$

□

A.9 Proof of Proposition 4 independent parallel sampling stuck in bad samples

Proposition 4 (Independent parallel sampling stuck in bad samples)**.** *On Ising model G in Equation* (3) *under Assumption 4, if the initial $\boldsymbol{X}^{(0)}$ is such that $\sum_{i\in C_G}\boldsymbol{X}_i^{(0)}\le -2$, $\forall\delta\in(0,1)$, with probability at least $1-\delta$, after $T:=\left\lfloor \frac{\delta}{2}\exp\left(c_{stuck}\right)\right\rfloor$ steps of* ***independent parallel*** *(Gibbs sampler 2), we have $\forall t\in[T], \sum_{i\in C_G}\boldsymbol{X}_i^{(t)}\le -2$, in which $c_{stuck}:=\frac{2\left(-1+\frac{1-\exp(-2J_0)}{\exp(-2J_0)+1}\frac{|C_G|}{2}\right)^2}{|C_G|}$*

Proof. Suppose at step t, $\boldsymbol{X}^{(t)}$ is such that $\sum_{i\in C_G}\boldsymbol{X}_i^{(t)}\le -2$ (satisfied at $t=0$), then

$$\forall j\in C_G,\ \sum_{i\in C_G, i\neq j}\boldsymbol{X}_i^{(t)}\le -1 \tag{A.22}$$

Hence its next-step distribution $X_j^{(t+1)}\sim p(\cdot\mid X_{-\{j\}}^{(t)})$ satisfies

$$\begin{aligned}
\frac{\mathbb{P}\left\{X_j^{(t+1)}=1\right\}}{\mathbb{P}\left\{X_j^{(t+1)}=-1\right\}} &= \frac{\exp\left(\sum_{i\in[N]}\boldsymbol{h}_i\mathbf{x}_i+\sum_{i\neq j\in C_G\subset[N]}J\mathbf{x}_i\mathbf{x}_j\right)|_{\mathbf{x}_j=1}}{\exp\left(\sum_{i\in[N]}\boldsymbol{h}_i\mathbf{x}_i+\sum_{i\neq j\in C_G\subset[N]}J\mathbf{x}_i\mathbf{x}_j\right)|_{\mathbf{x}_j=-1}} \quad \text{(by definition in Equation (3))}\\
&= \frac{\exp\left(\boldsymbol{h}_j+\sum_{i\in C_G,i\neq j}J\mathbf{x}_i\right)}{\exp\left(-\boldsymbol{h}_j-\sum_{i\in C_G,i\neq j}J\mathbf{x}_i\right)} \quad \text{(canceling the same terms)}\\
&= \exp\left(2\boldsymbol{h}_j+2J\sum_{i\in C_G,i\neq j}\mathbf{x}_i\right)\\
&\le \exp\left(2\boldsymbol{h}_j-2J\right) \quad \text{(by Equation (A.22))}\\
&\le \exp\left(-2J_0\right) \quad \text{(by Assumption 4)}
\end{aligned}$$

Therefore

$$X_j^{(t+1)}=\begin{cases}1, & \text{with prob } \le \frac{\exp(-2J_0)}{\exp(-2J_0)+1}\\ -1, & \text{with prob } \ge \frac{1}{\exp(-2J_0)+1}\end{cases} \tag{A.23}$$

Denote

$$Y_j:=\frac{X_j^{(t+1)}+1}{2} \tag{A.24}$$

Note that $\{Y_j\mid j\in[N]\}$ are independent Bernoulli random variables.

By Lemma 4, $\forall r>0$, with probability at least $1-2e^{-\frac{2r^2}{|C_G|}}$,

$$\begin{aligned}
\frac{1}{|C_G|}\sum_{j\in C_G}Y_j &< \mathbb{E}_{j\in C_G}\left[Y_j\right]+\frac{r}{|C_G|} \quad \text{(by Hoeffding's inequality Lemma 4)}\\
&\le \frac{\exp(-2J_0)}{\exp(-2J_0)+1}+\frac{r}{|C_G|} \quad \text{(by Equation (A.23) and definition of } Y_j \text{ in Equation (A.24))}
\end{aligned}$$

implying that with probability at least $1-2e^{-\frac{2r^2}{|C_G|}}$,

$$\frac{1}{|C_G|}\sum_{j\in C_G}X_j^{(t+1)}=2\frac{1}{|C_G|}\sum_{j\in C_G}Y_j-1<2\left(\frac{\exp(-2J_0)}{\exp(-2J_0)+1}+\frac{r}{|C_G|}\right)-1$$

i.e.

$$\sum_{j\in C_G}X_j^{(t+1)}<\frac{\exp(-2J_0)-1}{\exp(-2J_0)+1}|C_G|+2r$$

Setting RHS to -2 solves to

$$r = -1 + \frac{1 - \exp(-2J_0)}{\exp(-2J_0) + 1} \frac{|C_G|}{2}$$

Hence

$$\text{with probability at least } 1 - 2e^{-\frac{2\left(-1+\frac{1-\exp(-2J_0)}{\exp(-2J_0)+1}\frac{|C_G|}{2}\right)^2}{|C_G|}}, \quad \sum_{j \in C_G} X_j^{(t+1)} < -2 \tag{A.25}$$

By union bound, $\forall T \in \mathbb{N}_+$,

$$\text{with probability at least } 1 - 2Te^{-\frac{2\left(-1+\frac{1-\exp(-2J_0)}{\exp(-2J_0)+1}\frac{|C_G|}{2}\right)^2}{|C_G|}}, \quad \forall t \in [T], \quad \sum_{j \in C_G} X_j^{(t)} < -2 \tag{A.26}$$

Note that when $\sum_{j \in C_G} X_j^{(t)} < -2$, $\boldsymbol{X}^{(t)} \notin \mathcal{R}_1$.

Finally, aligning the probabilities: setting

$$2Te^{-\frac{2\left(-1+\frac{1-\exp(-2J_0)}{\exp(-2J_0)+1}\frac{|C_G|}{2}\right)^2}{|C_G|}} = \delta$$

solves to

$$T = \frac{\delta}{2} e^{\frac{2\left(-1+\frac{1-\exp(-2J_0)}{\exp(-2J_0)+1}\frac{|C_G|}{2}\right)^2}{|C_G|}}$$

□

A.10 Proof of Corollary 3: Separation between N-Gibbs sampler and independent parallel sampling

This section provides additional information for the discussion at the end of Section 3.2.

Assumption 5 (Strong interactions in Ising model)**.** *On Ising model G in Equation* (3)*, for parameters $\delta \in (0,1)$ and $M \in \mathbb{N}_+$,*

$$|C_G| \geq 8\left(1 + \ln \frac{4M}{\delta}\right)$$
$$h_G \geq \frac{1}{2} \ln \frac{2(2-\delta)}{\delta}$$
$$J_0 \geq \frac{1}{2} |C_G| \ln 2$$

Corollary 3 (Separation between N-Gibbs sampler and independent parallel sampling)**.** *On Ising model G in Equation* (3) *under Assumption 4, $\forall \delta \in (0,1)$, $\forall M \in \mathbb{N}_+$, If G additionally satisfies Assumption 5 and the initial $\boldsymbol{X}^{(0)}$ is such that $\sum_{i \in C_G} \boldsymbol{X}_i^{(0)} \leq -2$, then with probability at least $1-\delta$,*

1. *Running N-**Gibbs sampler**: $\boldsymbol{X}_{N.c.w.}^{(1)} \in \mathcal{R}_1$, and*
2. *Running **independent parallel**: $\{\boldsymbol{X}_{indep}^{(t)} | t \in [M]\} \cap \mathcal{R}_1 = \emptyset$*

Proof. Under the given conditions, with N**-Gibbs sampler**, by Proposition 3,

$$\text{with probability at least } 1 - \frac{\delta}{2}, \quad \{\boldsymbol{X}_{N.c.w.}^{(t)} | t \in [\left\lceil \log_{c_{\mathcal{R}_1}} \frac{\delta}{2} \right\rceil]\} \cap \mathcal{R}_1 \neq \emptyset \tag{A.27}$$

in which the constant

$$c_{\mathcal{R}_1} := 1 - \frac{\binom{N-|C_G|}{N-|C_G|}}{\binom{N}{N}} \frac{e^{2(J_0+h_G)}}{e^{2(J_0+h_G)} + e^{2J_0} + 2^{|C_G|} - 2} = 1 - \frac{e^{2(J_0+h_G)}}{e^{2(J_0+h_G)} + e^{2J_0} + 2^{|C_G|} - 2} \tag{A.28}$$

Applying Assumption 5 to bound parts of the RHS:

$$\frac{e^{2J_0}}{e^{2(J_0+h_G)}} = e^{-2h_G} \leq e^{-\ln \frac{2(2-\delta)}{\delta}} = \frac{\delta}{2(2-\delta)}$$

$$\frac{2^{|C_G|} - 2}{e^{2(J_0+h_G)}} \leq \frac{2^{|C_G|}}{e^{2(J_0+h_G)}} \leq \frac{2^{|C_G|}}{e^{|C_G| \ln 2 + \ln \frac{2(2-\delta)}{\delta}}} = \frac{2^{|C_G|}}{2^{|C_G|} \frac{2(2-\delta)}{\delta}} = \frac{1}{\frac{2(2-\delta)}{\delta}} = \frac{\delta}{2(2-\delta)}$$

Taking the sum:

$$\frac{e^{2J_0} + 2^{|C_G|} - 2}{e^{2(J_0+h_G)}} \leq \frac{\delta}{2-\delta}$$

Adding 1 to both sides:

$$\frac{e^{2(J_0+h_G)} + e^{2J_0} + 2^{|C_G|} - 2}{e^{2(J_0+h_G)}} \leq \frac{2}{2-\delta}$$

Taking the inverse:

$$\frac{e^{2(J_0+h_G)}}{e^{2(J_0+h_G)} + e^{2J_0} + 2^{|C_G|} - 2} \geq \frac{2-\delta}{2}$$

Plugging to Equation (A.28):

$$c_{\mathcal{R}_1} \leq 1 - \frac{2-\delta}{2} = \frac{\delta}{2}$$

Plugging into Equation (A.27):

$$\text{with probability at least } 1 - \frac{\delta}{2}, \quad \{\boldsymbol{X}_{N.c.w.}^{(t)} | t \in [1]\} \cap \mathcal{R}_1 \neq \emptyset \tag{A.29}$$

On the other hand, with **independent parallel**, by Proposition 4,

$$\text{with probability at least } 1-\frac{\delta}{2}, \quad \{\boldsymbol{X}_{indep}^{(t)} | t \in [\left\lfloor \frac{\delta}{4} \exp\left(c_{\text{stuck}}\right) \right\rfloor]\} \cap \mathcal{R}_1 = \emptyset \tag{A.30}$$

in which the constant

$$c_{\text{stuck}} := \frac{2\left(-1+\frac{1-\exp\left(-2J_0\right)}{\exp\left(-2J_0\right)+1}\frac{|C_G|}{2}\right)^2}{|C_G|} \tag{A.31}$$

Applying Assumption 5 to bound parts of the RHS:

$$\frac{1-\exp\left(-2J_0\right)}{\exp\left(-2J_0\right)+1} \geq \frac{1}{2}$$

Plugging into Equation (A.31):

$$\begin{aligned}
c_{\text{stuck}} &\geq \frac{2\left(-1+\frac{1}{2}\frac{|C_G|}{2}\right)^2}{|C_G|} \\
&= \frac{2\left(1-\frac{|C_G|}{2}+\frac{|C_G|^2}{4}\right)}{|C_G|} \\
&\geq -1+\frac{|C_G|}{8} \\
&\geq -1+\left(1+\ln\frac{4M}{\delta}\right) \quad \text{(by Assumption 5)} \\
&= \ln\frac{4M}{\delta}
\end{aligned}$$

Plugging into Equation (A.30):

$$\text{with probability at least } 1-\frac{\delta}{2}, \quad \{\boldsymbol{X}_{indep}^{(t)} | t \in [\left\lfloor \frac{\delta}{4} \cdot \frac{4M}{\delta} \right\rfloor] = [M]\} \cap \mathcal{R}_1 = \emptyset \tag{A.32}$$

By union bound, with probability at least $1-\delta$, both Equation (A.29) and Equation (A.32) hold.

□

A.11 Background and proofs of Proposition 1 and Proposition 2: on the expressive power of Transformers for implementing sequence-to-sequence Markov chains in parallel

Background: Transformer network architecture. The transformer architecture (Vaswani et al., 2017) is a critical building block of many leading approaches to language modeling (Devlin et al., 2019; Brown et al., 2020). We refer the readers to these works for more details on the empirical promise that Transformer-based models have demonstrated. For theoretical understanding of Transformers, we refer the readers to prior works on their representational power (Yun et al., 2020; Yao et al., 2021; Liu et al., 2023; Zhao et al., 2023), statistical sample complexity (Wei et al., 2021; Edelman et al., 2022), optimization process (Lu et al., 2021; Jelassi et al., 2022; Li et al., 2023), and interpretability (Wen et al., 2023), and references cited therein.

Mathematical setup. In the following we adapt and use the mathematical notations for the Transformer network architecture in Yun et al. (2020) and Li et al. (2023).

For each position of an input sequence (N tokens), use a d-dimensional *positional embedding* to represent that *position*, and use a d-dimensional *token embedding* for the *content* at that position. Hence, for the input sequence, both the token embeddings $\boldsymbol{E}$ and the positional embeddings $\boldsymbol{P}$ are matrices in $\mathbb{R}^{d\times N}$. Following empirical convention, let the input to the Transformer be

$$\boldsymbol{X} \coloneqq \boldsymbol{E} + \boldsymbol{P}$$

A *Transformer block* $t^{h,m,r}$ (with h heads, head size m, and feed-forward hidden layer size r) is defined as

$$t^{h,m,r}(\boldsymbol{X}) \coloneqq \mathrm{Attn}(\boldsymbol{X}) + \boldsymbol{W}_2 \cdot \mathrm{ReLU}(\boldsymbol{W}_1 \cdot \mathrm{Attn}(\boldsymbol{X}) + \boldsymbol{b}_1 \boldsymbol{1}_n^T) + \boldsymbol{b}_2 \boldsymbol{1}_n^T \tag{A.33}$$

where

$$\mathrm{Attn}(\boldsymbol{X}) \coloneqq \boldsymbol{X} + \sum\nolimits_{i=1}^{h} \boldsymbol{W}_O^i \boldsymbol{W}_V^i \boldsymbol{X} \cdot \sigma[(\boldsymbol{W}_K^i \boldsymbol{X})^T \boldsymbol{W}_Q^i \boldsymbol{X}] \tag{A.34}$$

where the weight parameters $\boldsymbol{W}_O^i \in \mathbb{R}^{d\times m}$, $\boldsymbol{W}_V^i, \boldsymbol{W}_K^i, \boldsymbol{W}_Q^i \in \mathbb{R}^{m\times d}$, $\boldsymbol{W}_2 \in \mathbb{R}^{d\times r}, \boldsymbol{W}_1 \in \mathbb{R}^{r\times d}, \boldsymbol{b}_2 \in \mathbb{R}^d, \boldsymbol{b}_1 \in \mathbb{R}^r$, and

$$\sigma : \mathbb{R}^{N_1\times N_2} \mapsto (0,1)^{N_1\times N_2}$$

is the column-wise softmax operation, such that

$$\sigma(A)_{ij} = \frac{\exp(A_{ij})}{\sum_{l=1}^{N} \exp(A_{lj})} \tag{A.35}$$

Finally, a Transformer is a composition of Transformer blocks:

$$\mathcal{T} \coloneqq \{g : \mathbb{R}^{d\times N} \to \mathbb{R}^{d\times N} \mid g \text{ is a composition of Transformer blocks } t^{h,m,r}\text{'s}\}. \tag{A.36}$$

and its output $\mathcal{T}(\boldsymbol{X}) \in \mathbb{R}^{d\times N}$ goes through a final affine transform and softmax (Equation (A.35)) to predict a distribution over tokens, for all positions

$$\mathcal{T}_{\text{pred}}(\boldsymbol{X}) \coloneqq \sigma\left(\boldsymbol{W}^{\text{pred}} \mathcal{T}(\boldsymbol{X}) + \boldsymbol{b}^{\text{pred}}\right) \in (0,1)^{|\Omega|\times N} \tag{A.37}$$

where $\boldsymbol{W}^{\text{pred}} \in \mathbb{R}^{|\Omega|\times d}$ and $\boldsymbol{b}^{\text{pred}} \in \mathbb{R}^{|\Omega|}$ are the prediction head weights and biases. Ω is the vocabulary of tokens.

For each position j, the predicted token τ_j is sampled from the predicted distribution $\mathcal{T}_{\text{pred}}(\boldsymbol{X})_{:,j}$ *independently* with other positions

$$\tau_j \sim \texttt{sample}(\mathcal{T}_{\text{pred}}(\boldsymbol{X})_{:,j}) \quad j \in [N] \tag{A.38}$$

where `sample` can be the standard sampling algorithm for multinomial distributions, or truncating the low-probability tail (Holtzman et al., 2020), or more conservatively, argmax sampling.

Yun et al. (2020) proved the following result on the expressivity of the Transformer network architecture:

Lemma 7 (Universal approximation by Transformers, informal (Yun et al., 2020))**.** *Let $1 \leq p < \infty$ and $\epsilon > 0$, then for any compact set $\mathcal{D} \subset \mathbb{R}^{d \times n}$, for any given function $f : \mathcal{D} \mapsto \mathbb{R}^{d \times n}$, there exists a Transformer network $g \in \mathcal{T}^{2,1,4}$ of $O(N \left(\frac{1}{\delta}\right)^{dN})$ layers such that*

$$\Big(\int \|f(\boldsymbol{X}) - g(\boldsymbol{X})\|_p^p \, d\boldsymbol{X} \Big)^{1/p} \leq \epsilon$$

in which δ is the smallest real number such that $\forall \boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{d \times n}$, if $\|\boldsymbol{X} - \boldsymbol{Y}\|_\infty < \delta$, then $\|f(\boldsymbol{X}) - f(\boldsymbol{Y})\|_p < \epsilon$. Moreover, the bound on the size of the constructed Transformer is asymptotically tight.

Lemma 8 (Transformers can simulate parallel solution to automata, informal (Liu et al., 2023))**.** *Transformers can simulate the length-T output of all semiautomata with states Q, input alphabet Σ, and transition function $\delta : Q \times \Sigma \mapsto Q$. Moreover, the size of the simulating Transformer has depth $O(\log T)$, embedding dimension $O(|Q|)$, attention width $O(|Q|)$, and MLP width $O(|Q|^2)$.*

Remark 5. *Lemma 8 gives a more compact construction than a direct implication of more general universal approximation results Lemma 7 for Transformers.*

A direct corollary is Proposition 1:

Proposition 1 (informal)**.** *Transformers (with sufficient depth and width) can implement any number of transitions of any deterministic Markov Chain over sequences in Ω^N.*

Informal proof sketch. When each transition of a Markov chain is deterministic, i.e. if the next state distribution from any state is always a delta function, then the Markov chain reduces to a deterministic finite state automata, with states Ω^N, length N.

Applying Lemma 8, we get Transformers can simulate length-T output of this automata with depth $O(\log T)$, embedding dimension $O(|\Omega|^N)$, attention width $O(|\Omega|^N)$, and MLP width $O(|\Omega|^{2N})$. □

Proposition 2 (informal)**.** *The class of Markov chains over sequences in Ω^N implementable by (sufficiently wide and deep) Transformers is those whose next-state transition probability distributions are product distributions over the positions, conditioned on the current state.*

Informal proof sketch. The statement involves both a positive result and a negative result.

Positive: if the transition probability distribution is a product distribution conditioned on the current state, then the task of representing a Markov chain can be reduced to universally approximating a continuous function which maps all sequences to the correct logits $\boldsymbol{W}^{\text{pred}}\mathcal{T}(\boldsymbol{X}) + \boldsymbol{b}^{\text{pred}}$ in Equation (A.37), such that after softmax (Equation (A.35)) these logits produce the correct marginal distribution at each position. This is achievable by the construction in Lemma 7.

Negative: if the transition probability distribution is *not* a product distribution conditioned on the current state, then note that the sampling operations (Equation (A.38)) at positions j_1 and j_2 are *independent*, so Transformers cannot implement such Markov chains. □

Remark 6. *As stated in Section 3, the sampling process in Savinov et al. (2022) and our experiments are different from N-Gibbs sampler. Moreover, despite Proposition 2, the sampling process is more different from* ***independent parallel*** *(Gibbs sampler 2): note that* ***independent parallel*** *strictly freezes all $X_{-\{i\}}^{(t)}$ when sampling*

$$X_i^{(t+1)} \sim p(\cdot \mid X_{-\{i\}}^{(t)})$$

whereas in Savinov et al. (2022) and our experiments, the model is trained to update all positions in parallel, which implies a different groundtruth next-iteration token distribution compared with $p(\cdot \mid X_{-\{i\}}^{(t)})$.

Mechanistically, Savinov et al. (2022) and our models in principle can take certain inter-position dependency into consideration (which ***independent parallel*** *cannot): for example, in layer L, position i can attend to [19] other positions e.g. j in the layer-$(L-1)$ representations. This enables the layer-L computation at position i to be conditioned upon the intermediate representations at position j, which are* not independent *from the final prediction at position j.*

[19] via Transformer attention Equation (A.34)

B ADDITIONAL EXPERIMENTAL DETAILS

B.1 INFERENCE AND TRAINING SETTING

B.1.1 INFERENCE

An input sequence X^{source} first goes through the encoder $f^{\text{enc}}_{\theta_e}$ (parameterized by θ_e) to produce the hidden representation h:

$$h = f^{\text{enc}}_{\theta_e}(X^{\text{source}})$$

A length predictor $f^{\text{len}}_{\theta_l}$ (parameterized by θ_l) takes h and predicts B_l most likely target lengths, where $B_l \in \mathbb{N}_+$ (beam size for length prediction) is an inference-time hyperparameter.

For each predicted length N, an initial hypothesis target sequence $X^{(0)} = X^{(0)}_1 \cdots X^{(0)}_N$ in which each $X^{(0)}_i$ can be a `[MASK]` token, or chosen uniformly randomly from the vocabulary of tokens.

For each decoder step $t \in 1 \cdots T$, the decoder $f^{\text{dec}}_{\theta_d}$ (parameterized by θ_d) takes two inputs: h and $X^{(t)}_{1\cdots N}$, and refines the hypothesis target sequence to $X^{(t+1)}_{1\cdots N}$, using one forward pass:

$$X^{(t+1)}_{1\cdots N} = f^{\text{dec}}_{\theta_d}(X^{(t)}_{1\cdots N}, h) \tag{B.39}$$

where $T \in \mathbb{N}_+$ (number of refinement steps) is an inference-time hyperparameter, and we can stop early if $X^{(t+1)} = X^{(t)}$.

B.2 Discussion on modeling and metrics

Remark 7. *In principle, following a similar paradigm, a non-autoregressive* decoder-only *architecture is also possible. In this work we use encoder-decoder for two reasons: (1) Efficiency: in the iterative refinement process of the hypothesis target sequence, each forward pass only involves the decoder, but not the encoder. (2) Benchmarking: the encoder-decoder design is closer to a series of prior works, allowing for more informative comparison on benchmarks.*

We measure BLEU (Papineni et al., 2002) using the SacreBLEU implementation (Post, 2018) with language appropriate tokenizers [20]. For the same model, SacreBLEU on average reports a lower score than BLEU (e.g. see Savinov et al. (2022)). Unfortunately, this does not allow a direct comparison with most of the existing literature. This is a deliberate choice since it has been shown that subtle differences in preprocessing can significantly impact metrics (Schmidt et al., 2022), making comparisons error prone, and SacreBLEU is the recommended metric in Post (2018). Furthermore, common preprocessing steps (lowercasing, separating punctuation, stripping diacritics, etc.) may artificially inflate scores while not being fully reversible, as such preventing real-world uses for such models.

While bridging the gap between autoregressive and non-autoregressive model has so far focused on achieving parity in terms of BLEU scores, we believe this is insufficient. Since BLEU relies on n-gram overlaps between groundtruths and model predictions, it does not capture readability very well. Yet readability is paramount for most practical applications, and it is indisputably something that current autoregressive LMs excel at. To provide additional perspectives, we introduce a word-level stutter metric, computing how often consecutive words are repeated in the model output but not in the reference. For all datasets, we found that word-level stutter is 2 or more times more frequent for non-autoregressive models. Additionally, we also report benchmarking results measured by BLEURT scores (Sellam et al., 2020; Pu et al., 2021) in Table 3, in Appendix B) but found they also do not discriminate much between AR and NAR models.

For our experiments in Section 4.3, there are other error modes connected to the challenge of modeling target-side dependency, but they are more ambiguous for measuring and exactly locating. We do not aim to develop decoding algorithms tailored to just reducing stuttering rate. (After all, stuttering can be easily removed by rule-based postprocessing.) Instead, the above are general-purpose hypotheses which are potentially also predictive of other (more complex) failure modes related to target-side dependency.

[20] For public reproducibility: SacreBLEU signatures: BLEU+c.mixed+#.1+s.exp+tok.zh+v.1.3.0 for Chinese and BLEU+c.mixed+#.1+s.exp+tok.13a+v.1.3.0 for other languages.

B.3 QUANTITATIVE EXPERIMENTAL RESULTS

Table 1: Test SacreBLEU scores on three WMT datasets. We report scores without any preprocessing. Our AR baselines are trained on the distilled dataset for a fair comparison. The 'Steps' column indicates the number of decoding iterations. The '# Hyp.' column denotes the number of hypotheses decoded in parallel (beam size for AR models and top_k predicted lengths for NAR models).

			WMT14		WMT16		WMT17	
Model	# Hyp.	Steps	DE→EN	EN→DE	RO→EN	EN→RO	ZH→EN	EN→ZH
AR Baselines	5	N	33.50	29.54	34.89	29.75	-	-
PaDIR	5	4	33.49	28.61	33.98	28.98	26.47	32.59
	5	10	33.63	28.58	33.99	28.97	26.54	32.68

Table 2: Test BLEU scores on three WMT datasets for popular baselines. Note that these rely on a different BLEU implementation and sometimes additional preprocessing than the results reported in the body of our paper.

			WMT14		WMT16		WMT17	
Model	# Hyp.	Steps	DE→EN	EN→DE	RO→EN	EN→RO	ZH→EN	EN→ZH
DisCo AR Baselines	5	N	31.71	28.60	34.46	34.16	24.65	35.01
CMLM	5	4	30.75	26.73	33.02	33.67	22.57	33.58
	5	10	31.24	27.39	33.67	33.33	23.76	34.24
DisCo Easy-First	5	3-6	31.31	27.34	33.25	33.22	23.83	34.63
SUNDAE Stochastic	16	4	32.10	27.94	-	-	-	-
	16	10	32.29	28.33	-	-	-	-

Table 3: Test BLEURT scores on three WMT datasets for our models.

			WMT14		WMT16		WMT17	
Model	# Hyp.	Steps	DE→EN	EN→DE	RO→EN	EN→RO	ZH→EN	EN→ZH
AR Baselines	5	N	73.55	74.97	67.23	71.76	-	-
PaDIR	5	4	71.26	72.08	65.90	70.23	65.16	63.95
	5	10	71.82	73.28	66.09	70.49	66.19	64.30

Table 4: Stuttering positions have similar average last-layer self-attentions compared with non-stuttering adjacent positions. For each pair of adjacent positions in the generated sequence: (1) the 'self-attention scores' include both directions ; (2) The column 'min' denotes only including the minimum among such score over all attention heads, and likewise for 'avg' and 'max'; (3) the entries are mean $\pm$ standard deviation; (4) $\mathbb{P}\{\text{top-}k \text{ overlap}\}$ denotes the chances that the self-attention distribution at one position includes the other position among its top-k "most attended to" positions.

	self-attention scores			$\mathbb{P}\{\text{top-}k \text{ overlap}\}$	
stutter	min	avg	max	$k=1$	$k=2$
yes	0.0004 ± 0.0007	0.032 ± 0.023	0.16 ± 0.11	0.20	0.39
no	0.0005 ± 0.0007	0.033 ± 0.025	0.17 ± 0.12	0.17	0.37

Table 5: Stuttering positions on average have more similar last-layer cross-attentions than non-stuttering adjacent positions. For each pair of adjacent positions in the generated sequence: (1) the 'total variation distance' and 'cosine distance' (both have range $[0, 1]$) are taken for the two corresponding cross-attention distributions; (2) The column 'min' denotes only including the minimum among such distance over all attention heads, and likewise for 'avg' and 'max'; (3) the entries are mean $\pm$ standard deviation; (4) $\mathbb{P}\{\text{top-}k \text{ overlap}\}$ denotes the chances that the two cross-attention distributions overlap in terms of their top-k "most attended to" source positions.

	total variation distance			cosine distance			$\mathbb{P}\{\text{top-}k \text{ overlap}\}$	
stutter	min	avg	max	min	avg	max	$k=1$	$k=2$
yes	0.06 ± 0.05	0.13 ± 0.09	0.23 ± 0.15	0.01 ± 0.01	0.10 ± 0.06	0.25 ± 0.11	0.57	0.89
no	0.11 ± 0.10	0.23 ± 0.14	0.35 ± 0.18	0.04 ± 0.08	0.20 ± 0.11	0.38 ± 0.12	0.40	0.81

C ADDITIONAL RELATED WORKS

Our theory and experiments draw inspirations from a wide variety of domains in natural language processing, generative models, and sampling. While a comprehensive list of all works in these domains is intractable, the following works significantly influenced our thinking:

Our theory is inspired by recent progress in sampling: the connections between pseudolikelihood and approximate tensorization of entropy are discussed in Marton (2013; 2015); Caputo et al. (2015); Caputo & Parisi (2021); Koehler et al. (2023). Benefits of k-Gibbs sampler are discussed in Lee (2023). Our experiments follow the framework that trains generative masked language models and generates samples using parallel decoding by iterative refinement: (Lee et al., 2018; Ghazvininejad et al., 2019; 2020; Kasai et al., 2020; Savinov et al., 2022), which tend to be at least twice faster than autoregressive approaches with a small drop in quality for tasks like machine translation. The inference process, which converts complete noise to full samples, might resemble diffusion models (Hoogeboom et al., 2021; Austin et al., 2021; Li et al., 2022; Gong et al., 2023; Zheng et al., 2023; Lou et al., 2023), but a key conceptual difference is that diffusion models are trained to revert a small amount of noise at each step, whereas the family of models that we study in this work are more similar to *masked autoencoders*: the training objective encourages reconstructing the whole target sequence in each step of decoding.

Non-autoregressive text generation Previous works applied various generative models to text, such as VAEs (Bowman et al., 2016; Bosc & Vincent, 2020), GANs (Che et al., 2017; Yu et al., 2017; Lin et al., 2017; Guo et al., 2018), and normalizing flows (Ziegler & Rush, 2019; Ma et al., 2019; Hoogeboom et al., 2021), but without a strong autoregressive component, the quality of generated text is often suboptimal. Later works achieve high-quality text generation through diffusion models (Hoogeboom et al., 2021; Austin et al., 2021; Li et al., 2022; Gong et al., 2023; Zheng et al., 2023) and energy-based models (Deng et al., 2020; Goyal et al., 2022; Qin et al., 2022), but their generation speeds tend to be much slower than autoregressive language models. Inference latency can be mitigated by approaches like Lee et al. (2020). Unlike the above paradigms that adapt continuous-domain generative models to text, our approach is closer to the following line of works that iteratively refine the generation process through parallel updates in the space of discrete token sequences, which tend to be at least twice faster than autoregressive approaches with a small drop in quality (Lee et al., 2018; Ghazvininejad et al., 2019; Stern et al., 2019; Ghazvininejad et al., 2020; Kasai et al., 2020; Savinov et al., 2022). The generation quality of non-autoregressive models can be further improved by incorporating some autoregressive components (Kong et al., 2020; Reid et al., 2022) or input-output alignment (Chan et al., 2020; Saharia et al., 2020). Insights such as the multimodality problem and components such as sequence-level knowledge distillation and input token fertility prediction were also proposed in (Gu et al., 2018). The benefit of distillation was verified in Kim & Rush (2016); Gu et al. (2018); Zhou et al. (2020); Gu & Kong (2021). Positional attention was tested in Gu et al. (2018); Kreutzer et al. (2020). Related to generation from MLMs, Wang & Cho (2019) use the learned conditionals inside a Gibbs sampler, but when the conditionals are not *consistent*, i.e. there is not a joint distribution that satisfies these conditionals, Gibbs sampler may amplify errors. In general, mathematical understanding about sampling from masked language models is still lagging substantially behind. Additionally, related to MLMs, Meng et al. (2023) analyzes some representational limitations, and Liu et al. (2022) analyzes subtleties from a parameter identifiability view. Related to parallel decoding, recent work (Cai et al., 2024) parallelizes the inference with multiple heads by finetuning autoregressive LLM backbones.

Theory about parallel sampling Koehler et al. (2023) proved a generalization bound for pseudolikelihood estimator via the classic ($k = 1$) approximate tensorization of entropy, in the "proper learning" setting. Our generalization bound (Theorem 3) uses the generalized notion of the approximate tensorization of entropy (Definition 8), also apply to "improper learning" settings, and the proof involves quite different techniques. The classic approximate tensorization of entropy are discussed in Marton (2013; 2015); Caputo et al. (2015), which was more recently generalized to the "α-weighted block" version (Definition 8) in Caputo & Parisi (2021). Lee (2023) proves that k-Gibbs sampler mixes at least k times faster than 1-Gibbs sampler. For future works, recent algorithmic advances in parallel sampling could potentially be incorporated into our framework to achieve finer-grained theoretical analysis or better empirical quality-efficiency trade-off (Anari et al., 2023).