Local Normalization Distortion and the Thermodynamic Formalism of Decoding Strategies for Large Language Models

Tom Kempton[1,*], **Stuart Burrell**[2,*]

[1]Department of Mathematics, University of Manchester
[2]Innovation Lab, Featurespace

Correspondence: thomas.kempton@manchester.ac.uk

Abstract

Advances in hardware and language model architecture have spurred a revolution in natural language generation. However, autoregressive models compute probability distributions over next-token choices, and sampling from these distributions, known as decoding, has received significantly less attention than other design choices. Existing decoding strategies are largely based on heuristics, resulting in methods that are difficult to apply or improve in a principled manner. We develop the theory of decoding strategies for language models by expressing popular decoding algorithms as equilibrium states in the language of ergodic theory and stating the objective functions they optimize. Using this, we analyze the effect of the local normalization step required to make probabilities sum to one in top-k, nucleus, and temperature sampling. We argue that local normalization distortion is a fundamental defect of decoding strategies and quantify the size of this distortion and its effect on mathematical proxies for the quality and diversity of generated text. This yields conclusions for the design of decoding algorithms and the detection of machine-generated text.

1 Introduction

Autoregressive large-language models are poised to transform industries such as healthcare, finance and education (Zhao et al., 2024). Rapid advancements in this field have been fueled by scaling (Hoffmann et al., 2024), transformer network architectures (Vaswani et al., 2017), and alignment processes (Christiano et al., 2017; Rafailov et al., 2024). However, autoregressive language models produce conditional *distributions* over predicted next tokens, and these must be iteratively sampled during inference. A suite of *decoding* methods exist for this sampling, such as top-k, nucleus, or temperature sampling, but these are based on heuristics and our understanding of these methods is in its infancy, with most being developed through trial and error (Wiher et al., 2022).

This is surprising, since the choice of decoding strategy has a profound impact on the quality of the generated text and, in some cases, may be more important than the model architecture (Wiher et al., 2022). For example, greedy sampling, or the related concept of beam search, tends to produce accurate but repetitive or dull text (Holtzman et al., 2020). In contrast, pure sampling produces much more varied and interesting text but can produce text which is 'incoherent and almost unrelated to the context' (Holtzman et al., 2020).

Our primary goal is to fill this gap and initiate the theoretical study of decoding strategies. As an application of this theory, we investigate a well-known defect of popular decoding strategies, which results from truncating the distribution of possible next tokens at each step during inference. Truncation necessitates repeated renormalization to obtain valid probability distributions, and results in a phenomenon we term *local normalization distortion*. The fact that local normalization distorts the resulting probability distribution is well-known, but not well-understood.

Our contributions are as follows.

1. Develop a theoretical framework for analyzing decoding strategies. This involves describing the probability distributions q produced by decoding algorithms as equilibrium states, drawing heavily on the language of ergodic theory and thermodynamic formalism (Bowen, 2008). We precisely state the objective function maximized by each decoding strategy; see Section 5.

2. Quantify the effect of local normalization distortion by showing how the probability of ran-

[*]Equal contribution.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 22216–22231
November 4-9, 2025

domly chosen strings change when we replace a locally normalized decoding strategy with a globally normalized equivalent. The effect is large for top-k and temperature sampling, but much smaller for nucleus sampling; see Section 6.1.

3. Evaluate language models and decoding strategies in terms of a quality-diversity trade-off, as in Caccia et al. (2019). We show, both theoretically and empirically, that local normalization distortion negatively affects the performance of the decoding strategy.[1]

These results show that in the ongoing search for better decoding strategies for large language models, careful attention should be paid to local normalization distortion, as it may have a large effect on proxies for the quality of generated text, and the size of this effect varies considerably with choice of decoding strategy. Additionally, in a follow-up article (Kempton et al., 2025) we show how to use local normalization distortion to detect machine-generated text.

2 A Motivating Example

Many autoregressive models for natural language generation work broadly as follows. Given a vocabulary $\mathcal{V}$, one builds a large neural network to estimate the likelihood $p(y_t|\mathbf{y_{<t}})$ that the next token in a sequence is equal to $y_t \in \mathcal{V}$, given the previous tokens $\mathbf{y_{<t}} = y_0 \cdots y_{t-1} \in \mathcal{V}^t$. Then, one decides on a decoding strategy (sampling algorithm), which is a way of using the collection of likelihoods

$$\{p(y_t|\mathbf{y_{<t}}) : y_t \in \mathcal{V}\}$$

associated with context $\mathbf{y_{<t}}$ to choose the next token y_t. For example, one could always choose the token y_t which has highest likelihood $p(y_t|\mathbf{y_{<t}})$ (greedy sampling), or one could allow each token y_t to be chosen with probability equal to the likelihood $p(y_t|\mathbf{y_{<t}})$ (pure sampling). Let $q(\cdot|\mathbf{y_{<t}})$ denote the distribution of choices of y_t given the context and chosen decoding strategy.

Having chosen the token y_t, one repeats the process to choose y_{t+1} given the new context $\mathbf{y_{<t+1}}$. One computes the probability of a string $y_0 \cdots y_T \in \mathcal{V}^*$ by

$$q(y_0 \cdots y_T) = \prod_{t=1}^{T} q(y_t|\mathbf{y_{<t}}).$$

In many settings, rather than conditioning on the whole history $\mathbf{y_{<t}}$ one allows only a finite context length L and conditions on $y_{t-L} \cdots y_{t-1}$, making the process of generating texts an L-step Markov process.

A simple example shows how the decoding strategy can dramatically influence q. Suppose we have a language model that produces model likelihoods p. We feed in the context 'The cat sat on' and obtain the likelihoods of the various choices of the next two tokens. Let these likelihoods be as described in Figure 1. Assume further that we decide that various choices of tokens are unreliable and so we wish to restrict our choices to the two most likely tokens depicted in blue boxes. We now have probabilities which do not sum to one, and so have to decide how to normalize them.

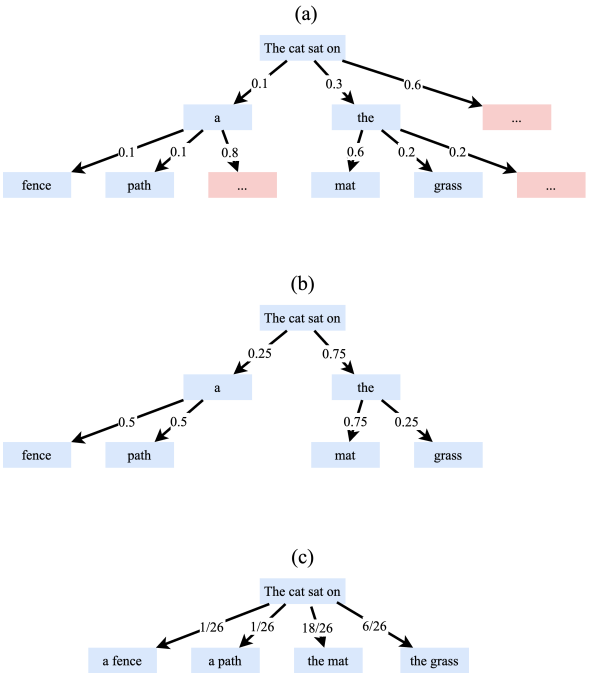

Figure 1: Distortion due to local normalization significantly impacts the implied probability distribution during decoding. (a) shows pure sampling, (b) shows locally normalized sampling, and (c) shows globally normalized sampling.

Option 1: Global normalization. Compute the probabilities of complete strings and divide by the sum of the probabilities of complete strings. In our

[1]Imperfect mathematical proxies for quality and diversity of text are used in this analysis, see Section 4. Code to reproduce our experiments is available at https://github.com/TMKempton/lnd.

example, the sum of the probabilities of complete strings is $0.26 = 0.01 + 0.01 + 0.18 + 0.06$. We end up with 'The cat sat on a fence' being selected with probability $0.01/0.26 \approx 0.038$.

Option 2: Local normalization. Renormalize conditional probabilities locally so that the probabilities of outward edges from each node sum to one. For example, we choose the first word 'a' with probability $0.1/0.4$ and 'the' with probability $0.3/0.4$. With local normalization, we end up with 'The cat sat on a fence' being selected with probability $0.25 \times 0.5 = 0.125$.

Observe that these two different methods of normalizing probabilities yield very different probability distributions.

3 Decoding Strategies

The decoding strategies we study in this article fall into two classes, truncation sampling algorithms (such as nucleus sampling and top-k) and temperature sampling. Truncation sampling algorithms work by defining an allowed set $\mathcal{A}_{\mathbf{y_{<t}}}$ of tokens that can follow some context, restricting the model conditional probability distribution $p(\cdot|\mathbf{y_{<t}})$ to this set and renormalizing it to have mass one.

Definition 3.1 (Truncation Sampling Algorithms)**.** *Given a language model p, a context $\mathbf{y_{<t}}$ and an allowed set $\mathcal{A}_{\mathbf{y_{<t}}}$, define*

$$Z(\mathbf{y_{<t}}) = \sum_{w_t \in \mathcal{A}_{\mathbf{y_{<t}}}} p(w_t|\mathbf{y_{<t}}).$$

Choose element y_t of $\mathcal{A}_{\mathbf{y_{<t}}}$ with probability

$$q(y_t|\mathbf{y_{<t}}) := \frac{p(y_t|\mathbf{y_{<t}})}{Z(\mathbf{y_{<t}})}$$

and set

$$q(y_t|\mathbf{y_{<t}}) = 0$$

for $y_t \notin \mathcal{A}_{\mathbf{y_{<t}}}$.

Top-k sampling (Fan et al., 2018) is an example of a truncation sampling strategy. Given some value of $k \geq 1$, it is defined by setting the allowed set $\mathcal{A}_{\mathbf{y_{<t}}}$ to be the set of those k tokens with highest model probabilities $p(y_t|\mathbf{y_{<t}})$. Top-k sampling restricts to the k most likely tokens at each stage of language generation, and renormalizes mass at each stage of generation by dividing model probabilities by the sum of the probabilities of the top-k tokens.

Nucleus (top-p) sampling (Holtzman et al., 2020) is defined by choosing a value $\pi \in [0, 1]$, ordering tokens $w_1, w_2 \cdots \in \mathcal{V}$ in order of decreasing model probability $p(w_i|\mathbf{y_{<t}})$ and setting $\mathcal{A}_{\mathbf{y_{<t}}} = \{w_1, \cdots, w_r\}$ where the threshold r is the smallest natural number satisfying $\sum_{i=1}^{r} p(w_i|\mathbf{y_{<t}}) \geq \pi$. Rather than choosing from exactly k tokens at each stage, nucleus sampling samples from (roughly) the top proportion π of the probability distribution. It is worth stressing that $\sum_{i=1}^{r} p(w_i|\mathbf{y_{<t}})$ is often much larger than π, and so the normalizing constant $Z(\mathbf{y_{<t}}) = \sum_{i=1}^{r} p(w_i|\mathbf{y_{<t}})$ can vary significantly at different contexts.

Several newer truncation-based decoding strategies have been introduced with different clever ways of choosing the allowed sets $\mathcal{A}_{\mathbf{y_{<t}}}$, see, for example, locally typical sampling (Meister et al., 2023), η-sampling (Hewitt et al., 2022), basis-aware truncation (Finlayson et al., 2023) and mirostat (Basu et al., 2020). One observation about this body of research is that careful motivation is always given for the choice of allowed set, but little if any consideration is given to the way in which the resulting decoding strategy apportions probability mass within this allowed set.

Finally, we mention temperature sampling, which is the only widely used stochastic sampling algorithm not to fall into the framework of truncation sampling.

Definition 3.2 (Temperature Sampling (Guo et al., 2017))**.** *Given some context $\mathbf{y_{<t}}$ and a parameter $\tau > 0$ (usually $\tau \in (0, 1)$), define*

$$Z_\tau(\mathbf{y_{<t}}) = \sum_{w_t \in \mathcal{V}} (p(w_t|\mathbf{y_{<t}})^{\frac{1}{\tau}}).$$

The distribution q_τ given by temperature sampling is defined by

$$q_\tau(y_t|\mathbf{y_{<t}}) = \frac{p(y_t|\mathbf{y_{<t}})^{\frac{1}{\tau}}}{Z_\tau(\mathbf{y_{<t}})}.$$

3.1 Global Normalization

As in our toy example, we could replace the local normalization in top-k, nucleus and temperature sampling with a global normalization, in which rather than normalizing conditional probabilities by dividing by $Z(\mathbf{y_{<m+i}})$ at each step, one normalizes the joint distribution over complete sequences $w_1 \cdots w_T$. For example, if we let $\mathcal{A}_\mathbf{T}$ denote the set of sequences of length T for which each token is in the top-k set, globally normalized top-k

sampling selects string $y_1 \cdots y_T \in$ with probability

$$q'_k(y_1 \cdots y_T) = \frac{p(y_1 \cdots y_T)}{\sum_{w_1 \cdots w_T \in \mathcal{A}_{\mathbf{T}}} p(w_1 \cdots w_T)}. \tag{1}$$

That is, globally normalized top-k sampling samples according to the measure p conditioned on the subset $\mathcal{A}_{\mathbf{T}}$. Similar statements hold for globally normalized variants of any restriction sampling algorithm. Globally normalized temperature sampling selects tokens with probability proportional to $p(\cdot|\mathbf{y_{<t}})^{\frac{1}{\tau}}$, as is the case when temperature is used in statistical physics, ergodic theory and fractal geometry (see Appendix C).

We let q'_k, q'_π and q'_τ denote the globally normalized alternatives to top-k, nucleus and temperature sampling respectively. Globally normalized decoding strategies are computationally infeasible, even for fairly small values of T. We introduce them here as a theoretical tool to better understand how problematic local normalization distortion is. In Appendix B we explain how to sample from q'_k, q'_π and q'_τ using rejection sampling.

3.2 Local Normalization Distortion

Definition 3.3. *Let q be the distribution produced by a locally normalized decoding strategy, and let q' be its globally normalized counterpart. Given a context $\mathbf{y_{<t}}$, the local normalization distortion associated to completion $y_t \cdots y_T$ is defined as*

$$\frac{q(y_t \cdots y_T|\mathbf{y_{<t}})}{q'(y_t \cdots y_T|\mathbf{y_{<t}})}.$$

In the case of top-k sampling, given context $\mathbf{y_{<t}}$, there is a constant C such that each completion $y_t \cdots y_T$ has local normalization distortion

$$\frac{1}{C} \cdot \frac{1}{\prod_{i=0}^{T-t} Z_k(\mathbf{y_{<t+i}})},$$

where Z_k is the mass of the top-k tokens at context $\mathbf{y_{<t+i}}$. The constant C is the normalizing constant associated to global normalization, which is hard to compute but can be bypassed in empirical investigations, see Section 6.

There is a body of work studying global normalization in the context of constrained decoding; see, for example, Lipkin et al. (2025); Loula et al. (2025) and references therein. In these works, local normalization distortion is seen as a problem and various algorithms for sampling approximately from the globally normalized distribution are proposed.

By contrast, Gareev et al. (2024) argue that global decoding underperforms local decoding for top-k and top-π sampling. The primary driver of the effects they observe seems to be the fact that local and global sampling produce texts of markedly different lengths (differing by a factor of four for some parameter settings), whereas we study generations of fixed length. Additionally, their approach differs from ours in that they compare quality of local and global top-k sampling at the equal values of k, whereas we compare quality of local and global top-k sampling at equal values of diversity, following the approach of Caccia et al. (2019). We justify our experimental approach in Section 6 and give further details on the difference between our work and Gareev et al. (2024) in Appendix H.

We conclude this section with a proposition that further motivates the study of local normalization distortion.

Proposition 3.1. *Let p be a language model, q_τ denote the distribution arising from temperature sampling, and q'_τ the distribution arising from a globally normalized version of temperature sampling. Then, as the temperature parameter τ tends to zero, q_τ converges to the distribution putting all of its mass on the output of greedy sampling, whereas q'_τ converges to the distribution putting all of its mass on the sequence with globally maximal* log *probability.*

Proof. See Appendix D.1. □

Given the substantial interest in implementing expensive search algorithms such as beam search to find sequences with approximately the globally maximal log probability, we present Proposition 3.1 as initial evidence that local normalization distortion can have a substantial negative effect and is worthy of further investigation.

4 Evaluating Decoding Strategies through a Quality-Diversity Trade Off

When generating text from a language model, one may have different preferences for the 'quality' and 'diversity' of the text according to the task being performed (Caccia et al., 2019; Wiher et al., 2022). By diversity, we mean the capacity for a model to produce different samples, while by quality we mean the average human judged quality of an individual sample.

This framing of choice of decoding strategy choice as a trade-off of quality and diversity is stud-

ied in Caccia et al. (2019); Ippolito et al. (2019); Nadeem et al. (2020); Zhang et al. (2021). In particular, in Caccia et al. (2019), the authors propose using a 'temperature sweep' to find a parameter τ for which temperature sampling best matches this preference. Similarly, one can adjust the parameters in top-k or nucleus sampling according to one's preferences for diversity versus quality, since restricting token choices to the top of the distribution prioritizes quality over diversity.

There are no universally accepted definitions of the diversity and quality of text. One way of evaluating the diversity of stochastically generated text is to look at the entropy $H(q)$ of the distribution q of the text, given by

$$H(q) = \sum_{\mathbf{y} \in \mathcal{V}^*} q(\mathbf{y}) \log q(\mathbf{y}).$$

The sum here is taken over complete strings.

The gold standard for evaluating the quality of the generated text is to get human judgment scores, although this is expensive and fraught with difficulty (Clark et al., 2021). Often the model log-likelihood $\log(p)$ is used as a proxy for quality, so the quality of a distribution q over possible texts would be given by

$$Q(q) = \sum_{\mathbf{y} \in \mathcal{V}^*} q(\mathbf{y}) \log p(\mathbf{y}).$$

This notion of quality is not without its issues (Meister et al., 2022), although Zhang et al. (2021) has a rather compelling graph suggesting that human judgements of quality of text are well correlated with $Q(q)$ except when $Q(q)$ is very high.

While entropy and average log-likelihood of a distribution q are imperfect, albeit frequently used, proxies for diversity and quality, they are precisely the right objects to describe mathematically the objective functions maximized by the distributions resulting from top-k sampling, nucleus sampling and temperature sampling.

5 Decoding Strategies as Equilibrium States

In the last section we reviewed the literature on what a decoding strategy ought to maximize. In this section we prove results about what popular decoding strategies actually optimize. In particular, we state results of the form 'given some context $\mathbf{y}_{<\mathbf{m}}$, the probability distribution q on the set $\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},k}$ obtained by sampling according to a certain decoding strategy is the unique probability distribution on $\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},k}$ that maximizes the following objective function...'.

In the language of ergodic theory, what we are doing is describing the outcome of a decoding strategy as an *equilibrium state* associated to a certain potential. While the mathematics of this section is not hard, it is useful as it allows us to ask whether the function that our decoding strategy maximizes is well aligned with the theoretical goals of a decoding strategy. In a follow-up paper (Kempton et al., 2025) we apply the ideas of this section to build a novel algorithm for detecting text generated by a language model.

We use the following standard result. As usual, let $0 \log 0 := 0$.

Lemma 5.1 ((Bowen, 2008), Lemma 1.1)**.** *Let $X = \{1, \cdots, k\}$ be a finite set and let $R = (r_1, \cdots, r_k)$ be a probability measure on X assigning mass r_i to symbol i. Then R is the unique probability measure maximizing the quantity*

$$\underbrace{-\sum_{i \in X} \mu_i \log \mu_i}_{\textit{Entropy } H(\mu)} + \underbrace{\sum_{i \in X} \mu_i \log r_i}_{\textit{Average log probability}}$$

among probability measures $\mu = (\mu_1, \cdots \mu_k)$ on X.

The results of this section follow as direct corollaries to Lemma 5.1 by analyzing the measures $(r_1, ..., r_k)$ given by various decoding strategies. Details are given in the appendix D. Our first result concerns top-k decoding.

Corollary 5.1. *Given some context $\mathbf{y}_{<\mathbf{m}}$ and a choice of k, the distribution q_k on $\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},k}$ produced by top-k sampling is the unique distribution maximizing the quantity*

$$\underbrace{H(\mu)}_{\textit{Proxy for diversity}} + \underbrace{\sum_{\mathbf{w} \in \mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},k}} \mu(\mathbf{w}|\mathbf{y}_{<\mathbf{m}}) \log p(\mathbf{w}|\mathbf{y}_{<\mathbf{m}})}_{\textit{Proxy for quality}} + \underbrace{\sum_{\mathbf{w} \in \mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},k}} \mu(\mathbf{w}|\mathbf{y}_{<\mathbf{m}}) \log \epsilon_k(\mathbf{w})}_{\textit{Distortion term}}$$

among distributions μ on $\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},k}$. Here

$$\epsilon_k(\mathbf{w}) := \frac{1}{\Pi_{i=0}^{T-m} Z(y_0 \cdots y_m w_{m+1} \cdots w_{m+i-1})},$$

which is the product along the sequence of the inverse of the mass of the top k tokens.

If our equation above had only the first two terms, it would show that q_k maximizes the sum of mathematical proxies for diversity and quality among distributions supported on $\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},k}$. The third term however, which is an artifact of local normalization, distorts this goal. If the third term was constant across sequences $\mathbf{y}$ then it would have no effect, but issues arise when it has a large variance; see Section 6.1. Next, we consider nucleus sampling.

Corollary 5.2. *Given some context $\mathbf{y}_{<\mathbf{m}}$ and a choice of π, the distribution q_π on $\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},\pi}$ generated by nucleus sampling is the unique distribution maximizing the quantity*

$$H(\mu) \quad + \sum_{\mathbf{w}\in\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},\pi}} \mu(\mathbf{w}|\mathbf{y}_{<\mathbf{m}}) \log p(\mathbf{w}|\mathbf{y}_{<\mathbf{m}}) \\ + \sum_{\mathbf{w}\in\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},\pi}} \mu(\mathbf{w}|\mathbf{y}_{<\mathbf{m}}) \log \epsilon_\pi(\mathbf{w})$$

among distributions μ on $\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},\pi}$. Here

$$\epsilon_\pi(\mathbf{w}) := \frac{1}{\Pi_{i=0}^{T-m} Z(y_0 \cdots y_m w_{m+1} \cdots w_{m+i-1})},$$

which is the inverse of the product along the sequence $w_{m+1} \cdots w_T$ of the total mass of those tokens allowed by nucleus sampling.

Thus nucleus sampling produces a distribution q_π maximizing a goal related to quality, diversity and an error term related to both by the length of the sequence y and the extent to which the mass of the tokens selected by nucleus sampling overshoots the target π.

Finally, we consider temperature sampling.

Corollary 5.3. *Given some choice of temperature τ and context $\mathbf{y}_{<\mathbf{m}}$, the distribution q_τ is the distribution maximizing the quantity*

$$H(\mu) + \frac{1}{\tau} \sum_{\mathbf{w}\in\mathcal{V}^*} \mu(y_1 \cdots y_m \mathbf{w}) \log(p(y_1 \cdots y_m \mathbf{w})) \\ + \sum_{\mathbf{w}\in\mathcal{V}^*} \mu(y_1 \cdots y_m \mathbf{w}) \epsilon_\tau(\mathbf{w})$$

among distributions μ on $\mathcal{V}^*$, where

$$\epsilon_\tau(\mathbf{w}) = \frac{1}{\Pi_{i=0}^{T-m} Z_\tau(y_0 \cdots y_m w_{m+1} \cdots w_{m+i-1})}.$$

When nucleus or top-k sampling set $\pi < 1$ or $k < |\mathcal{V}|$, they produce distributions with lower entropy than p, since token choice has been reduced. This process also redistributes mass from low probability tokens to higher probability tokens, reducing the average log-likelihood. Thus the parameters associated with nucleus or top-k sampling allow one to prioritize quality or diversity. This is somewhat hidden in the statements of Corollaries 5.1 and 5.2, it appears only in that it is the space of distributions on $\mathcal{A}^*$ upon which the proxy for diversity, proxy for quality, and distortion term are maximized.

Temperature sampling is not a truncation algorithm, and the way it modulates between prioritizing quality and prioritizing diversity is in the factor $1/\tau$ preceding the mathematical proxy for quality in the main equation of Corollary 5.3. As before, the third term here is not related to the goal of maximizing quality or diversity. It is broadly similar to the distortion introduced by top-k, except rather than dividing by the product along a sequence of the mass contained in the top-k tokens, it divides by the product along a sequence $y_m \cdots y_T$ of the $L_{1/\tau}$ norm of the distribution $p(\cdot|\mathbf{y}_{<\mathbf{m}+\mathbf{i}})$.

Finally, we see that global normalization removes the problematic third term in each of the quantities maximized by our decoding strategies.

Corollary 5.4. *In each of corollaries 5.1-5.3, if the locally normalized probability distribution (q_k, q_π, or q_τ) is replaced by its globally normalized equivalent, the statement of the theorem remains the same except without the third term (i.e. the distortion term) in the expression for the maximized quantity.*

This is important because it shows that, in terms of the quality-diversity trade off goals of the previous section, locally normalized decoding strategies perform strictly worse than their globally normalized counterparts. In Section 6.2 we quantify this effect.

6 Experiments

In this section we quantify the size of effects predicted in our theoretical sections. In particular, we measure the size of local normalization distortion and its effect on the quality-diversity trade off, measured through log-likelihood and entropy.

While one may, in some circumstances, care more about semantic diversity than lexical diversity, or about human judgements of quality than language model log-likelihood, effects on these metrics would be downstream of the true mathematical effect of local normalization distortion, and likely highly dependent on both the language model

and the task being performed. Thus, for reasons both of stability and of quantifying our theoretical results directly, we run our experiments on metrics present in the objective functions maximized by decoding strategies.

The experiments of this section are run using Llama 2-7B (Touvron et al., 2023) on a single A100 GPU. Detailed setup for each experiment is contained in the corresponding sections and appendices. In Appendix E, we repeat our experiments using Pythia 1B and 2.8B (Biderman et al., 2023), and Llama 3.2 1B and 3.2 3B (Grattafiori et al., 2024). As predicted above, our experimental results remain qualitatively unchanged.

The primary challenge to running these experiments is the computational cost of global normalization. To do this efficiently, we introduce a process based on rejection sampling in Appendix B.

6.1 How Large is the Distortion due to Local Normalization under Different Decoding Strategies?

In assessing how much local normalization distortion affects the mass of a completion $y_m \cdots y_T$, we need to compare how much $q(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})$ is boosted by local normalization against how much it would have been boosted by global normalization, as in Definition 3.3. Considering, for example, top-k sampling, if q_k denotes the distribution produced by top-k sampling and q'_k denotes its globally normalized equivalent, we would like to compute

$$\frac{q_k(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})}{q'_k(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})}.$$

This is difficult to compute because globally normalized top-k assigns mass

$$q'_k(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}}) = \frac{p(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})}{C}$$

for some constant C, which is very expensive to compute, particularly on long generations. Instead, we generate pairs of completions $y_m \cdots y_T$ and $z_m \cdots z_T$ by top-k sampling and then compute the ratio of the two local normalization distortions by computing

$$\frac{q_k(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})}{q'_k(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})} \Big/ \frac{q_k(z_m \cdots z_T|\mathbf{y}_{<\mathbf{m}})}{q'_k(z_m \cdots z_T|\mathbf{y}_{<\mathbf{m}})} \quad (2)$$

which is equal to

$$\frac{q_k(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})}{p(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})} \cdot \frac{p(z_m \cdots z_T|\mathbf{y}_{<\mathbf{m}})}{q_k(z_m \cdots z_T|\mathbf{y}_{<\mathbf{m}})}. \quad (3)$$

Since we are taking the ratio of the two local normalization distortions, the global constants C cancel out and so do not need computing.

For $k = 5, 50, 150$, we start by finding values of π and τ such that on average, for a randomly chosen context, $\mathbf{y}_{<\mathbf{t}}$, $Z_k(\mathbf{y}_{<\mathbf{t}}) \approx Z_\pi(\mathbf{y}_{<\mathbf{t}}) \approx Z_\tau(\mathbf{y}_{<\mathbf{t}})$. We are seeking here to tune our parameters so that the average amount of renormalizing done by top-k, nucleus and temperature sampling is the same.

Having tuned parameters, we then compare the local normalization distortion across the three decoding strategies. Starting with the single word context 'The ', we generate 1000 pairs of completions of 100 tokens each for each of our decoding strategies and parameter choices. We compute the relative local normalization distortion given by the quantity (3) for top-k sampling and equivalent quantities for nucleus and temperature sampling, see Appendix A.

Our results are presented in Figure 3 and Table 1. We have two key findings.

Table 1: Local normalization distortion ratios over a range of comparable decoding strategies. The table is partitioned into three groups of parameters tuned so that the average amount of renormalization is approximately equal. Reported quantiles are the absolute value of the natural log of the ratio (2).

Decoding	Quantile				
Strategy	10%	25%	50%	75%	90%
$k = 5$	1.84	4.50	9.82	16.93	25.07
$\tau = 0.86$	1.18	2.76	5.97	10.80	15.65
$p = 0.65$	**0.73**	**2.05**	**4.58**	**7.91**	**11.32**
$k = 50$	0.60	1.43	3.26	5.60	7.92
$\tau = 0.95$	0.40	1.05	2.29	3.93	5.83
$p = 0.88$	**0.18**	**0.44**	**0.95**	**1.76**	**2.70**
$k = 150$	0.36	0.88	1.89	3.35	4.83
$\tau = 0.98$	0.18	0.47	0.95	1.68	2.43
$p = 0.95$	**0.06**	**0.16**	**0.34**	**0.62**	**0.96**

Finding 1. Local normalization distortion has a large effect. For example, when using temperature sampling with parameter $\tau = 0.86$ to sample pairs of sequences $\mathbf{w}, \mathbf{z}$, each of length 100, we see that in half of cases, the ratio $q_\tau(\mathbf{w}|\mathbf{c})/q_\tau(\mathbf{z}|\mathbf{c})$ differs from the ratio $p(\mathbf{w}|\mathbf{c})^{\frac{1}{\tau}}/p(\mathbf{z}|\mathbf{c})^{\frac{1}{\tau}}$ by a factor of at least $\exp(5.97) \approx 392$. That is, the effect of local normalization distortion is to distort the relative probabilities of two completions by a factor of 392.

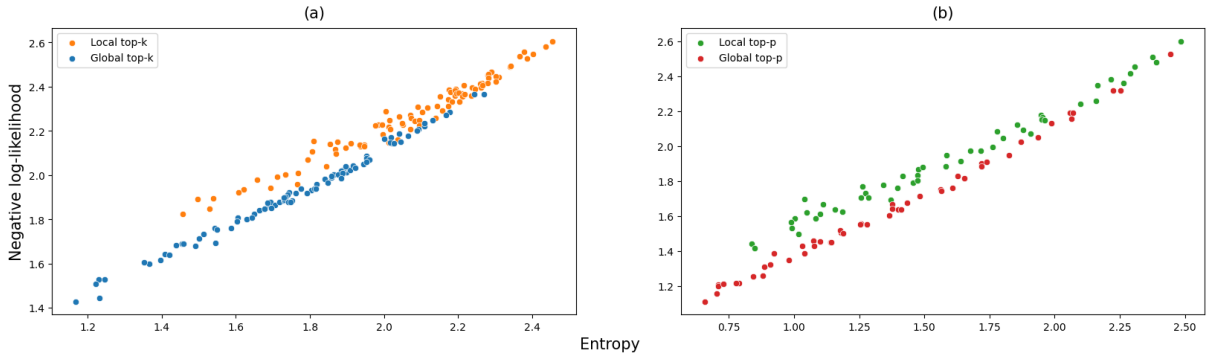

Figure 2: Evaluating quality against diversity at different parameter values when decoding with top-k, nucleus sampling, and globally normalized top-k. The ranges were $k = 10, 11, \ldots, 100$ and $p = 0.4, 0.41, \ldots, 0.9$. Higher values of entropy are better, lower values of negative-log-likelihood are better. We find that at any fixed value of entropy, globally normalized sampling tends to produce texts with higher log-likelihood.

It is worth stressing that temperature sampling is more often used with temperatures 0.7 or 0.8, in which case one would see even larger local normalization distortion.

Finding 2. When parameters k, τ and π are tuned so that the typical renormalization factors Z_k, Z_τ and Z_π are similar, nucleus sampling results in a much smaller local normalization distortion than temperature sampling, which in turn gives rise to a much smaller distortion than top-k sampling.

6.2 How does Local Normalization Distortion affect the Quality-Diversity Trade Off?

Given a single word context $w_1 =$ 'The ', we generate 30 length 15 samples by top-k sampling, nucleus sampling, temperature sampling and their globally normalized equivalents. We do this over a range of values of k, π and τ. For each sample $w_2 \cdots w_{16}$, we assess the quality by computing the negative log probability $-\log(p(w_2 \cdots w_{16}|w_1))$. In addition, we evaluate the diversity of our sample generation process by approximating the entropy of our generation process using the Shannon-McMillan-Breimann theorem (Walters, 2000). That is, for each choice of decoding strategy and parameter value and each completion $w_2 \cdots w_{16}$ generated by decoding strategy q, we compute $-\log(q(w_2 \cdots w_{16}|w_1))$. We average these values over the different completions generated for each particular decoding strategy and parameter value to approximate $H(q)$. Figure 2 shows these proxies for quality and diversity. Note that lower negative log probability and higher entropy are preferable.

Finding 3. For both top-k and nucleus sampling, globally normalized sampling outperforms locally normalized sampling on the quality-diversity trade off. That is, at any fixed value of entropy, globally normalized sampling produces texts with higher log-likelihood.

It is not tractable to compute globally normalized temperature sampling for sensible ranges of τ, due to the extreme rejection rate of our rejection sampling algorithm. Fortunately, we do not require experimental results in this setting thanks to Corollary 5.4, which states that globally normalized temperature sampling at temperature τ yields the unique distribution maximizing entropy plus $1/\tau$ log-likelihood among measures on on $\mathcal{A}^T$.

7 Conclusions

Our primary contribution is to express popular decoding strategies as equilibrium states. In doing so, one can see that the quantity which they maximize contains a term relating to local normalization of probability mass which seems unrelated to any reasonable goal of a decoding strategy. In particular, this term pulls the resulting probability distribution away from the quality-diversity maximizing curve. We have shown experimentally that the effect of local normalization distortion on the probability of selecting a string is typically very large (Section 6.1), and that it has a strongly negative effect on the quality-diversity tradeoff (Section 6.2) when these quantities are measured through entropy and log-likelihood. In a follow-up work (Kempton et al., 2025) we build a detector of machine generated text based on local normalization distortion which

outperforms state of the art alternatives. These factors lead us to the conclusion that local normalization distortion may have a negative effect on machine-generated text and that it should be carefully considered both when practitioners choose a decoding strategy and in the design of future methods for detecting machine-generated text.

8 Ethical Considerations

This work considers current decoding strategies for language models and ways in which these decoding strategies fall short. The most likely practical applications of it are in the detection of machine-generated text and in improving language models so as to make their outputs more human-like.

Although there are no specific ethical concerns about this work, we do inherit wider ethical questions around building human-like language models and detecting machine-generated text. A discussion of these is far beyond the scope of this work; instead, we encourage the reader to seek out the wealth of publicly available material on the issues.

9 Limitations

Our experiments are run on the open-source Llama 2 language model (Touvron et al., 2023). While this is not uncommon for research in computational linguistics, the setting in which decoding strategies such as temperature sampling are most widely deployed is closed-source models such as with ChatGPT (OpenAI, 2022). Our theoretical results hold for all language models and we do not believe the conclusions of our experimental section would change with language model. It is however the case that the magnitude of local normalization distortion would decrease if the entropies of the next token probability distributions were typically lower. Thus, language models with higher certainty about their next token predictions would give rise to smaller numbers in Table 1, for example.

We have used log-likelihood as a proxy for the quality of machine-generated text and entropy as a proxy for its diversity. These metrics are clearly imperfect. In particular, human judgement of text quality may be a better metric for the quality of a text, although obtaining these human judgments is often prohibitive due to cost (Clark et al., 2021).

References

Sourya Basu, Govardana Sachitanandam Ramachandran, Nitish Shirish Keskar, and Lav R Varshney. 2020. Mirostat: A Neural Text Decoding Algorithm that Directly Controls Perplexity. In *International Conference on Learning Representations*.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. 2023. Pythia: a suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Robert Edward Bowen. 2008. *Equilibrium states and the ergodic theory of Anosov diffeomorphisms*, volume 470. Springer Science & Business Media.

Julien Brémont. 2002. Gibbs measures at temperature zero. *Nonlinearity*, 16(2):419.

Massimo Caccia, Lucas Caccia, William Fedus, Hugo Larochelle, Joelle Pineau, and Laurent Charlin. 2019. Language GANs Falling Short. In *International Conference on Learning Representations*.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4302–4310, Red Hook, NY, USA. Curran Associates Inc.

Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A Smith. 2021. All That's 'Human'Is Not Gold: Evaluating Human Evaluation of Generated Text. 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)*, pages 7282–7296.

Kenneth Falconer. 2007. *Fractal geometry: mathematical foundations and applications*. John Wiley & Sons.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical Neural Story Generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Matthew Finlayson, John Hewitt, Alexander Koller, Swabha Swayamdipta, and Ashish Sabharwal. 2023. Closing the Curious Case of Neural Text Degeneration. *CoRR*.

Daniel Gareev, Thomas Hofmann, Ezhilmathi Krishnasamy, and Tiago Pimentel. 2024. Local and Global Decoding in Text Generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14577–14597, Miami, Florida, USA. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR.

John Hewitt, Christopher D Manning, and Percy Liang. 2022. Truncation Sampling as Language Model Desmoothing. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3414–3427.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack W. Rae, and Laurent Sifre. 2024. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The Curious Case of Neural Text Degeneration. In *International Conference on Learning Representations*.

Daphne Ippolito, Reno Kriz, João Sedoc, Maria Kustikova, and Chris Callison-Burch. 2019. Comparison of Diverse Decoding Methods from Conditional Language Models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3752–3762.

Oliver Jenkinson. 2019. Ergodic optimization in dynamical systems. *Ergodic Theory and Dynamical Systems*, 39(10):2593–2618.

Tom Kempton, Stuart Burrell, and Connor J Cheverall. 2025. TempTest: Local Normalization Distortion and the Detection of Machine-generated Text. In *International Conference on Artificial Intelligence and Statistics*, pages 1972–1980. PMLR.

Benjamin Lipkin, Benjamin LeBrun, Jacob Hoover Vigly, João Loula, David R. MacIver, Li Du, Jason Eisner, Ryan Cotterell, Vikash Mansinghka, Timothy J. O'Donnell, Alexander K. Lew, and Tim Vieira. 2025. Fast Controlled Generation from Language Models with Adaptive Weighted Rejection Sampling. *Preprint*, arXiv:2504.05410.

João Loula, Benjamin LeBrun, Li Du, Ben Lipkin, Clemente Pasti, Gabriel Grand, Tianyu Liu, Yahya Emara, Marjorie Freedman, Jason Eisner, Ryan Cotterell, Vikash Mansinghka, Alexander K. Lew, Tim Vieira, and Timothy J. O'Donnell. 2025. Syntactic and Semantic Control of Large Language Models via Sequential Monte Carlo. In *The Thirteenth International Conference on Learning Representations*.

Clara Meister, Tiago Pimentel, Gian Wiher, and Ryan Cotterell. 2023. Locally typical sampling. *Transactions of the Association for Computational Linguistics*, 11:102–121.

Clara Meister, Gian Wiher, Tiago Pimentel, and Ryan Cotterell. 2022. On the probability-quality paradox in language generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 36–45. Association for Computational Linguistics.

Moin Nadeem, Tianxing He, Kyunghyun Cho, and James Glass. 2020. A Systematic Characterization of Sampling Algorithms for Open-ended Language Generation. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 334–346.

OpenAI. 2022. ChatGPT: Optimizing Language Models for Dialogue. https://openai.com/blog/chatgpt.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2024. Direct preference optimization: your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Peter Walters. 2000. *An introduction to ergodic theory*, volume 79. Springer Science & Business Media.

Gian Wiher, Clara Meister, and Ryan Cotterell. 2022. On decoding strategies for neural text generators. *Transactions of the Association for Computational Linguistics*, 10:997–1012.

Hugh Zhang, Daniel Duckworth, Daphne Ippolito, and Arvind Neelakantan. 2021. Trading Off Diversity and Quality in Natural Language Generation. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 25–33.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2024. A Survey of Large Language Models. *Preprint*, arXiv:2303.18223.

A Computing Local Normalization Distortion for Nucleus and Temperature Sampling

In equations 2 and 3 we described how to compute the ratio of the local normalization distortions of two strings $y_m \cdots y_T$ and $z_m \cdots z_T$ in the case of top-k sampling.

The case of nucleus sampling is almost identical, we need only replace q_k with q_π in equation 3.

For temperature sampling, we note that $q'_\tau(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})$ is not proportional to $p(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})$, but to $p^{\frac{1}{\tau}}(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})$. Thus, to compute the ratio of the local normalization distortions for two strings $y_m \cdots y_T$ and $z_m \cdots z_T$ in the case of temperature sampling, we replace equation 3 with

$$\frac{q_\tau(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})}{p^{\frac{1}{\tau}}(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})} \cdot \frac{p^{\frac{1}{\tau}}(z_m \cdots z_T|\mathbf{y}_{<\mathbf{m}})}{q_\tau(z_m \cdots z_T|\mathbf{y}_{<\mathbf{m}})}.$$

B Rejection sampling algorithms

We can sample from q'_k, q'_π and q'_τ using rejection sampling. This remains incredibly computationally intensive, but it is significantly easier than computing the probability of each possible string $w_1 \cdots w_T$ as in equation (1).

In the case of top-k and nucleus sampling, with set of allowed completions $\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}}}$ given context $\mathbf{y}_{<\mathbf{m}}$, one can sample according to the globally normalized variant of top-k or nucleus as follows:

Step 1. Sample a completion $y_m \cdots y_T$ according to the model probability p.

Step 2. Accept the completion $y_m \cdots y_T$ if it is in the allowed set $\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}}}$, otherwise reject it and repeat step 1.

In the case of temperature sampling, given context $\mathbf{y}_{<\mathbf{m}}$ one can sample according to a globally normalized variant of temperature sampling as follows:

Step 1. Sample a completion $y_m \cdots y_T$ according to the model probability p.

Step 2. Accept this completion with probability $p(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})^{\frac{1}{\tau}-1}$. If the completion is not accepted, return to step 1.

We see that with each attempt to sample according to globally normalized temperature sampling, sample $y_m \cdots y_T$ is generated with probability

$$\begin{aligned} & p(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}}) \times p(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})^{\frac{1}{\tau}-1} \\ = \; & p(y_m \cdots y_T|\mathbf{y}_{<\mathbf{m}})^{\frac{1}{\tau}}. \end{aligned}$$

For further information on rejection sampling and faster algorithms which approximate rejection sampling see Lipkin et al. (2025).

C Global Temperature Normalization in Ergodic Theory and Fractal Geometry

We mentioned in Section 3.1 that global normalization of measures is a standard method in statistical physics, ergodic theory and fractal geometry. A short explanation of this remark is that globally normalized temperature sampling corresponds to taking the Gibbs-equilibrium measure associated to potential $\log p$ at temperature $\frac{1}{\tau}$. Similarly, when using a truncation sampling algorithm with allowed set $\mathcal{A}$, globally normalized sampling corresponds to sampling from the Gibbs-equilibrium measure associated to potential $\log p$ on the sequence space defined by $\mathcal{A}$. For both of these comments, see Bowen (2008).

For a more direct example, consider extremely long texts generated by pure sampling from a language model with finite context length L. The ergodic theory of Markov chains tells us that, with high probability, the average value of the log probability of a token from the text ('time average') will be close to the space average

$$\int_{\text{contexts } v_{<t}} \int_{v \in \mathcal{V}} p(v|v_{<t}) dp(v|v_{<t}) dp(v_{<t}).$$

One might ask what can be said about the set of texts for which the average log probability of tokens takes some different value α. How are typical such texts distributed? How many are there? Such questions are answered through the multifractal analysis of ergodic averages, see, for example, Falconer (2007). Solutions involve globally (rather than locally) normalized temperature sampling.

D Proofs

D.1 Proof of Proposition 3.1

We recall Proposition 3.1, which stated that the limit as temperature tends to zero of locally normalized temperature sampling is greedy decoding, whereas the limit as temperature tends to zero of globally normalized temperature sampling is the distribution achieving globally maximal average log-likelihood.

The statement on local temperature sampling has been widely noted. It is merely the statement that for any probability vector $(p_1, \cdots, p_k)$, with a unique value p_i larger than all other values, the vector

$$\left(\frac{p_1^{\frac{1}{\tau}}}{\sum_{j=1}^k p_j^{\frac{1}{\tau}}}, \frac{p_2^{\frac{1}{\tau}}}{\sum_{j=1}^k p_j^{\frac{1}{\tau}}}, \cdots, \frac{p_k^{\frac{1}{\tau}}}{\sum_{j=1}^k p_j^{\frac{1}{\tau}}}, \right)$$

converges to the unit vector with a 1 in position i as $\tau \to 0$.

The statement on global temperature sampling is more subtle and is a key result linking 'zero temperature limits of Gibbs measures' and 'ergodic optimization', see for example Brémont (2002); Jenkinson (2019).

D.2 Proof of Corollary 5.1.

We take Lemma 5.1 and set $X = \mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},k}$ to be the set of completions $y_m \cdots y_T$ belonging to the top-k set. Our distribution q_k is a probability measure on this set. Then Lemma 5.1 says that q_k is the unique probability measure maximising the quantity

$$H(\mu) + \sum_{\mathbf{w} \in \mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},k}} \mu(\mathbf{w}|\mathbf{y}_{<\mathbf{m}}) \log q_k(\mathbf{w}|\mathbf{y}_{<\mathbf{m}})$$

among probability measures μ on the top-k set $\mathcal{A}^*_{\mathbf{y}_{<\mathbf{m}},k}$. Note that

$$\begin{aligned}
&q_k(\mathbf{w}|\mathbf{y}_{<\mathbf{m}}) \\
&= \prod_{i=0}^{T-m} q_k(w_{m+i}|y_0 \cdots y_m w_{m+1} \cdots w_{m+i-1}) \\
&= \prod_{i=0}^{T-m} \frac{p(w_{m+i}|y_0 \cdots y_m w_{m+1} \cdots w_{m+i-1})}{Z_k(y_0 \cdots y_m w_{m+1} \cdots w_{m+i-1})} \\
&= \frac{p(\mathbf{w}|\mathbf{y}_{<\mathbf{m}})}{\prod_{i=0}^{T-m} Z_k(y_0 \cdots y_m w_{m+1} \cdots w_{m+i-1})}.
\end{aligned}$$

Taking logs and splitting the formula for $\log q_k(\mathbf{w}|\mathbf{y}_{<\mathbf{m}})$ into two distinct terms gives the result.

D.3 Proofs of Corollaries 5.2, 5.3 and 5.4.

These corollaries follow from Lemma 5.1 in an identical manner to the above proof of Corollary 5.1.

E Supplementary Figures and Replication on other Language Models

Figure 3 is a companion to Table 1 giving the results of Section 6.1 across all quantiles.

The experiments reported in the main body of the text were carried out using Llama 2 7B. We

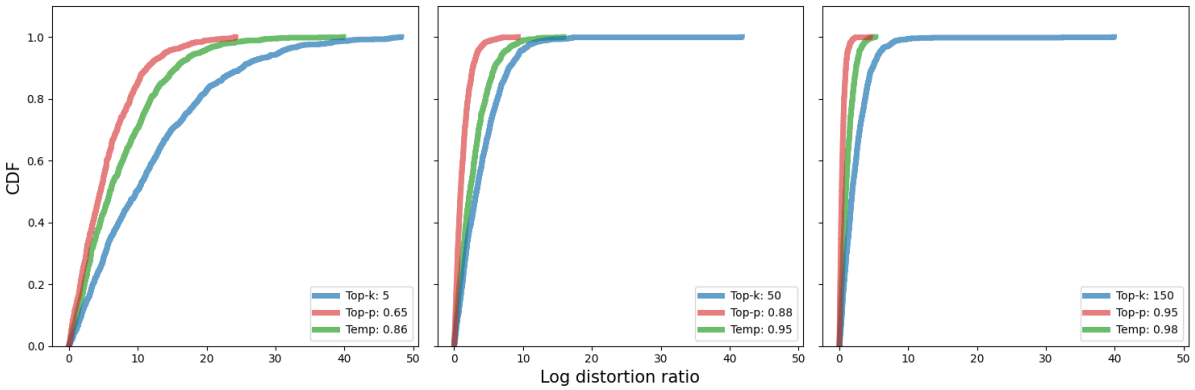

Figure 3: For each graph we generate pairs of texts according to top-k, nucleus or temperature sampling with Llama 2 7B. Parameters for the decoding strategies are tuned so that typical renormalizing quantitites Z_k, Z_π and Z_τ are of the same size. We then plot the log of the ratio of the local normalization distortion for the pair of generated texts, and then plot the cumulative distribution function. We see in each case that nucleus sampling produces the smallest local normalization distortion, followed by temperature sampling and top-k sampling.

repeat these experiments here for other language models from the Llama and Pythia families, and find that results align. We tune values of p and τ to k as described in Section 6.1. This process will have some noise and we note that the tuned values are slightly different for each model.

F A Comment on Timescales for Normalization

In this article we have discussed two different timescales for renormalizing probabilities which do not sum to one, namely locally renormalizing conditional probabilities at each timestep and globally renormalizing probabilities of strings $w_1 \cdots w_T$ at time T, where T is the length of the string we wish to generate. There is (at least in theory) a third natural option, which is to renormalize 'at time infinity'. For example, in the case of temperature sampling at temperature τ, this corresponds to taking the Gibbs measure associated to potential $p^{\frac{1}{\tau}}$ (Bowen, 2008). Precisely, there exists a unique measure q''_τ on the set $\mathcal{A}^\infty$ such that there exist constants C, P, independent of T, such that

$$\frac{1}{C} \leq \frac{q''_\tau(\{\underline{z} \in \mathcal{A}^\infty : z_1 \cdots z_T = w_1 \cdots w_T\})}{p^{\frac{1}{\tau}}(w_1 \cdots w_T)\exp(T.P)} \leq C.$$

The constant P, known as the topological pressure, could in theory be explicitly computed as the maximal eigenvalue of a very large matrix.

Normalizing at time infinity has some theoretical appeal, in that it produces a Markov measure whose transition probabilities do not depend on the sequence length T.

G A Note on Quality and Diversity

Quality and diversity are broad terms which could admit several interpretations. The notion that our metrics attempt to capture can be described as follows. Consider the scenario in which a language model is prompted 'please tell me a joke'. In our interpretation, the model would be judged high on the quality metric even if it always responded with the same joke, provided the joke was a good one. The model would be judged high on the diversity metric provided there is a high chance that, when prompted twice, it would produce different outputs, even if these outputs are nonsense, or if an individual output repeats itself, or if the different outputs are semantically similar.

H Comparisons with the Article: Gareev et al. (2024)

We would like to thank one of our anonymous referees for making us aware of the article Gareev et al. (2024). Their headline conclusion, 'in most configurations, global decoding performs worse than the local decoding versions of the same algorithms', seems in direct opposition to ours, while their results are actually entirely consistent with our own. We carefully point out how the apparent inconsistencies arise.

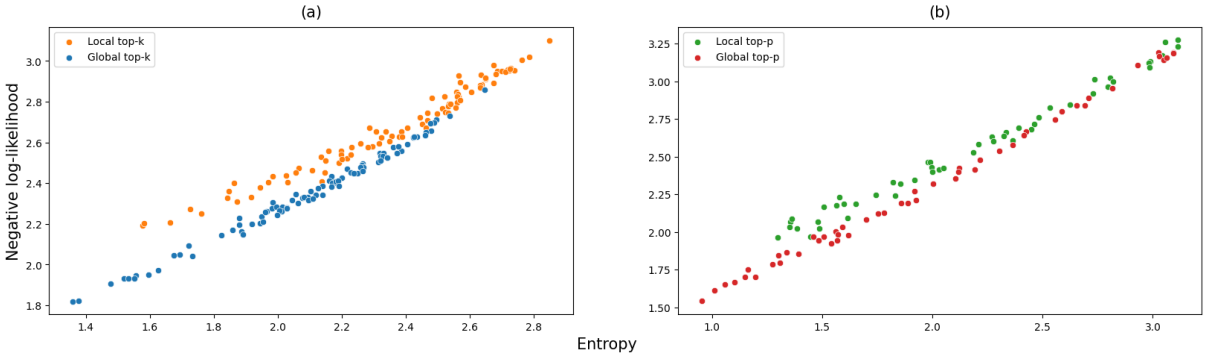

Figure 4: Figure 2 repeated for Llama 3.2 1B.

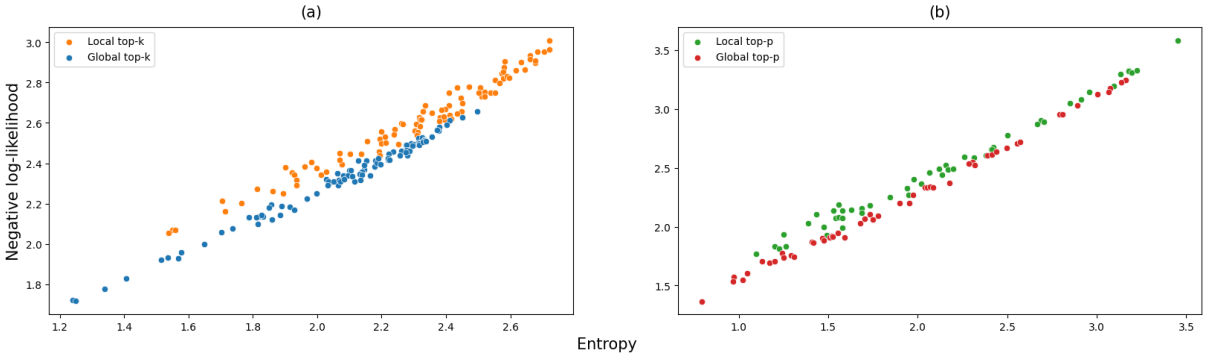

Figure 5: Figure 2 repeated for Llama 3.2 3B.

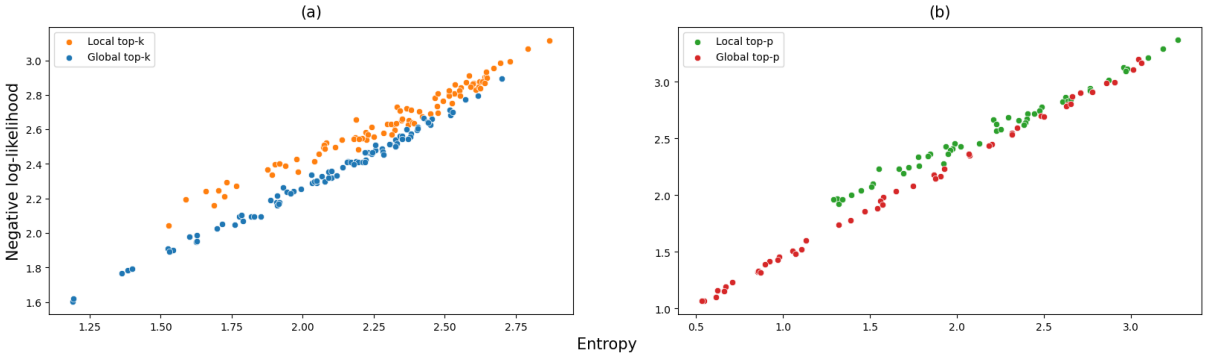

Figure 6: Figure 2 repeated for Pythia 1B.

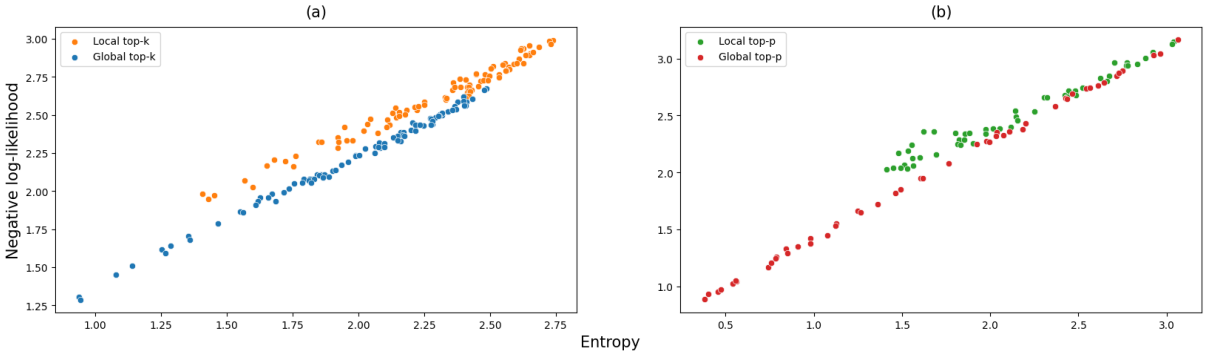

Figure 7: Figure 2 repeated for Pythia 2.8B.

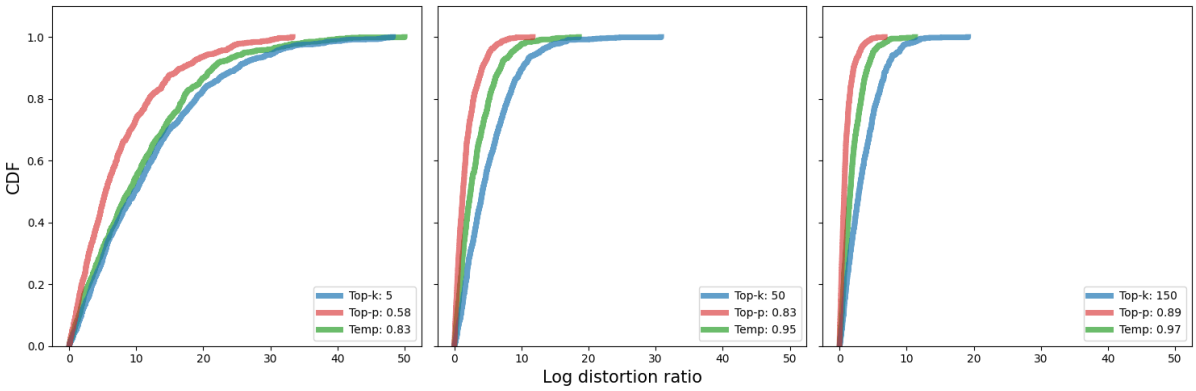

Figure 8: Figure 3 repeated for Llama 3.2 1B.

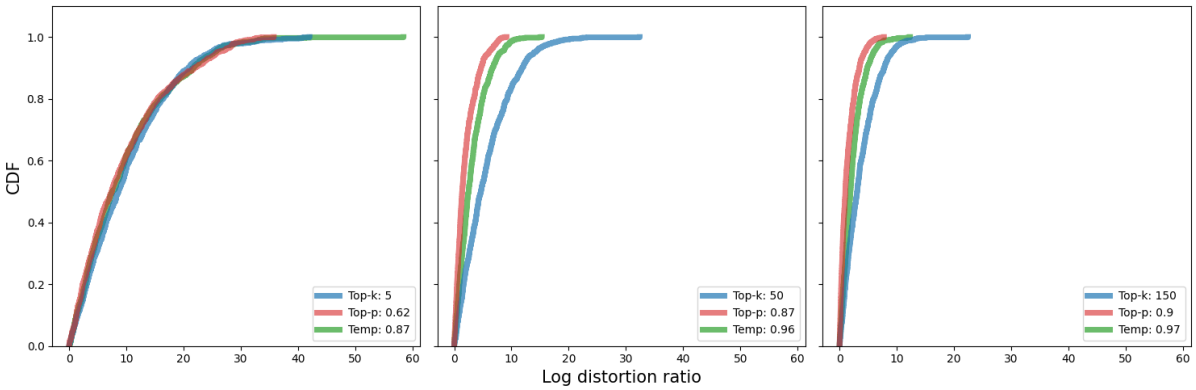

Figure 9: Figure 3 repeated for Llama 3.2 3B.

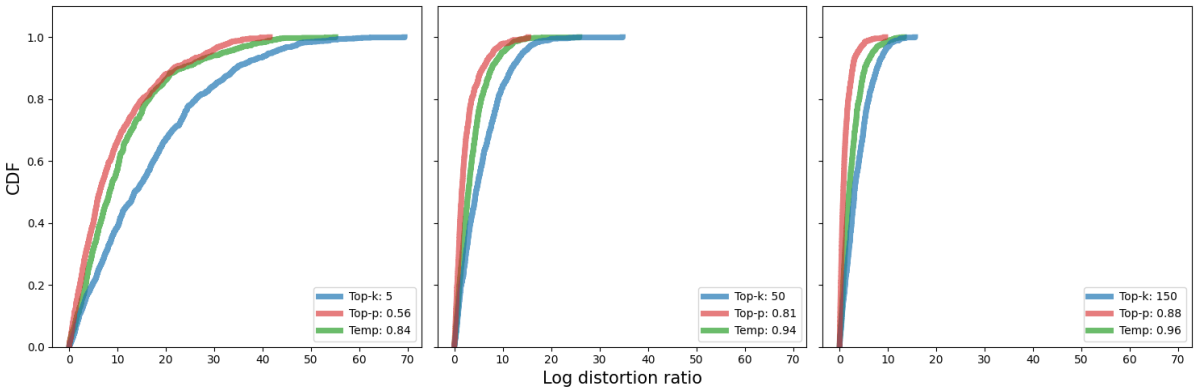

Figure 10: Figure 3 repeated for Pythia 1B.

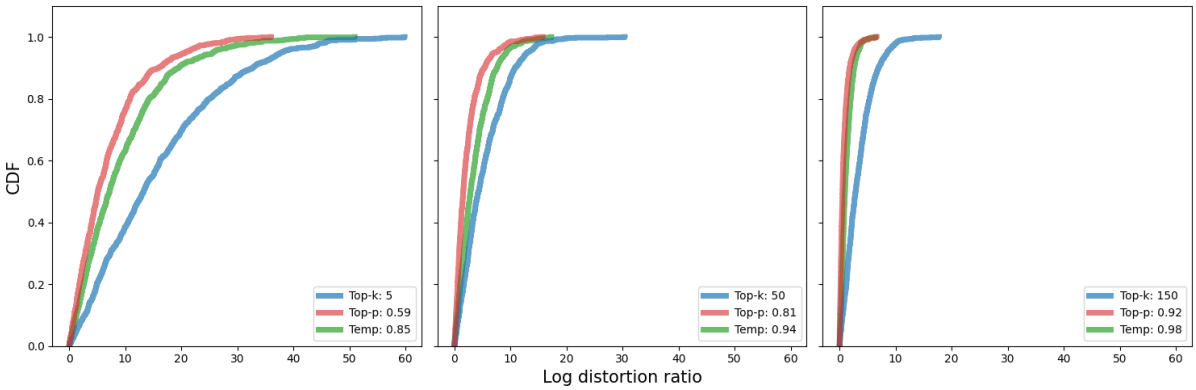

Figure 11: Figure 3 repeated for Pythia 2.8B.

H.1 Parameter Pinning and Results on Diversity

In comparing the effect of local and global decoding on (proxies for) quality and diversity, we follow the approach of Caccia et al. (2019) of plotting quality and diversity on the same plot (as in Figure 2). This allows us to compare, for example, locally normalized top-k vs globally normalized top-k across the whole range of parameters. Gareev et al. (2024) show that, for any fixed value of k, local decoding produces text which is **more** diverse than global decoding. We show that, at any desired level of quality of output, local decoding produces text which is **less** diverse than global decoding. There is no contradiction here and we agree with their result, as can be seen that the blue and red dots in Figure 2 are generally to the left of the orange and green dots. The apparent discrepancy arises since, if one wishes to compare local and global top-k at some fixed threshold for quality, one needs to use different values of k for the local and global decoding.

H.2 Results on Quality

As with diversity, Gareev et al. (2024) plot quality against parameter for local and global decoding, rather than plotting quality and diversity. However, in the case of their results on quality, we do not think this is the cause of the apparent discrepancy between our results and theirs. Instead, we think there are two fundamental factors.

MAUVE, their measure of quality, works by measuring both type 1 errors, where a language model produces un-human like text, and type-2 errors, where a language model fails to capture the full diversity of human language. This does not align with our desired notion of quality, which relates only to type-1 errors, see Section 4. Instead, in our language, MAUVE measures a convex combination of quality and diversity, and as such there is no clear discrepancy between their results and ours.

We suspect however that a much larger factor is at play. In our experiments we produce texts of constant length. Gareev et al. (2024) have a rather clever way of approximately sampling from the globally normalized distribution, which allows them to produce much longer texts. A consequence of not requiring fixed length generation is that their globally sampled texts are much shorter than their locally sampled ones, for some parameters by a factor of nearly 4. This length discrepancy is the first of their three suggested explanations of their results. The fact that global and local sampling produce outputs of such starkly differing lengths is very interesting, but isn't really what we were hoping to measure when we talk about quality. Indeed, the fact that our generated texts are of constant length has allowed us to avoid the thorny question of whether we should be measuring quality, or quality per token, and we hope that further research might look at how our theoretical results would be affected if one were to normalize by generation length.

I Licenses

We have used Llama 2, Llama 3, and Pythia models under their respective community license agreements.[2]. Use for research is consistent with the terms of these licenses.

[2]See https://huggingface.co/meta-llama and https://huggingface.co/EleutherAI.