Fine-Tuning Improves Information Conveyance in Language Models

Yuwei Cheng [*1] **Weiyi Tian** [*1] **Haifeng Xu** [1]

Abstract

Fine-tuning is often believed to reduce uncertainty and diversity in large language models, but existing analyses overlook output length, a key confounder, and therefore fail to capture how uncertainty is distributed across an entire generation rollout. To address this, we propose *Canopy Entropy* ($\mathrm{CE}^\star$), a measure that views language generation from a tree perspective, where "canopy" represents the space of all possible rollouts, making $\mathrm{CE}^\star$ naturally quantify the *effective size of the generation space*. $\mathrm{CE}^\star$ jointly captures uncertainty in both the output length N and the generated sequence $Y_{1:N}$ – indeed, we show that it equals total Shannon entropy $H(N, Y_{1:N} \mid X)$, where X denotes the prompt. This formulation yields interpretable metrics, including a length–entropy correlation term $\rho(N, r_N)$, where r_N is the entropy rate, quantifying *information conveyance efficiency* by indicating whether longer outputs are more or less informative per token. Empirically, across tasks and model families, we find that fine-tuned models consistently exhibit more positive correlation $\rho(N, r_N)$, even when total entropy decreases. Furthermore, after controlling for model family, task, prompt, and output-length effects, we find that fine-tuning nearly *triples* the correlation strength between entropy rate and semantic diversity, suggesting that aligned models convert token uncertainty into semantic diversity more efficiently. Overall, these results demonstrate that fine-tuning does not simply reduce uncertainty, but fundamentally reorganizes it into more informative and semantically meaningful generations. Our code is available at `https://github.com/WeiyiTian/canopy-entropy`.

[*]Equal contribution [1]The University of Chicago. Correspondence to: Yuwei Cheng <yuweicheng@uchicago.edu>, Weiyi Tian <weiyitian@uchicago.edu>.

ICML 2026 Workshop on Reinforcement Learning from World Feedback (RLxF), Seoul, South Korea, 2026.

1. Introduction

Large language models (LLMs) have achieved strong performance across tasks such as natural language understanding (Brown et al., 2020; Hendrycks et al., 2020), code generation (Chen et al., 2021; Roziere et al., 2023), and mathematical reasoning (Lightman et al., 2024; Lu et al., 2025). A key driver of this success is fine-tuning, including instruction tuning (Wei et al., 2021; Chung et al., 2024) and alignment with human preferences (Ouyang et al., 2022; Rafailov et al., 2023). However, alignment also systematically changes generation behavior, often producing substantially longer responses (Singhal et al., 2023). This has led to an important open question: do aligned language models merely become more verbose, or do they maintain informative and semantically meaningful generation over longer trajectories?

At the same time, a growing body of work suggests that fine-tuning reduces output diversity and compresses the effective generation space of language models (Wang et al., 2025; Yang et al., 2025b; Lake et al., 2025; West & Potts, 2025). Prior studies report that base models exhibit greater randomness and creativity, whereas aligned models produce more concentrated and similar responses across samples (West & Potts, 2025). Similar conclusions have been drawn using branching factor (Yang et al., 2025b), lexical diversity (Lake et al., 2025), and token-level uncertainty measures (Agarwal et al., 2025; Wang et al., 2026), reinforcing the view that fine-tuning makes language models more deterministic and less diverse.

However, existing analyses suffer from an important limitation – they do not control for output length, despite length being a major confounder in diversity measurement (Shaib et al., 2024). Classical metrics such as type-token ratio are biased by text length, often making shorter responses appear more diverse (Shaib et al., 2024). Such metrics that ignore generation length cannot distinguish genuinely diverse outputs from statistical artifacts induced by shorter outputs. This limitation is especially problematic in aligned language models because fine-tuning changes not only what models generate, but also how long they generate. Base models often produce shorter responses, while instruction-tuned models generate longer and more structured outputs (Singhal et al., 2023). As illustrated in Figure 2, base models typically exhibit decreasing entropy rates over generation,

implying that longer continuations become increasingly predictable and redundant. In contrast, fine-tuned models tend to maintain more stable entropy rates across longer trajectories, suggesting that additional tokens remain informative rather than collapsing into low-information continuation.

More importantly, prior work largely treats diversity as a static property of outputs, without studying how uncertainty evolves along generation trajectories. To the best of our knowledge, no existing work analyzes the relationship between generation length and entropy rate (see Definition 3)—that is, whether longer generations become more informative or increasingly redundant. Yet this interaction is crucial, since fine-tuning learns not only *what* to say, but also *when* to elaborate and *when* to stop. As a result, output length itself becomes part of the learned generation strategy, making diversity comparisons incomplete without accounting for how uncertainty is allocated across trajectories.

Meanwhile, recent studies suggest a more nuanced picture than simple diversity reduction. Fine-tuning can increase certain forms of diversity, such as semantic diversity in code generation, even while reducing lexical variation (Shypula et al., 2025). These findings suggest that fine-tuning may reorganize uncertainty and generation structure, rather than uniformly suppress diversity, leaving the relationship between uncertainty allocation, semantic diversity, and generation trajectories still poorly understood.

Our Contribution. In this work, we revisit diversity in language models through a principled information-theoretic lens. We propose *Canopy Entropy* ($\mathrm{CE}^\star$), an information-theoretic measure that views language generation from a tree perspective. Just as a biological canopy represents the total spread and reach of a tree's branches, our metric views the "canopy" as the set of all potential leaves—or finished rollouts—available to the model. This allows $\mathrm{CE}^\star$ to naturally quantify the effective volume of the generation space by accounting for both how *widely* the model branches and how *deeply* it explores. $\mathrm{CE}^\star$ is introduced to jointly capture output length and content uncertainty, and we show that it admits an exact characterization as the total Shannon entropy $H(N, Y_{1:N} \mid X)$, where X denotes the prompt, N is the generated length, and $Y_{1:N}$ is the generated sequence. Building on this formulation, we derive interpretable metrics and introduce a length–entropy correlation term that measures how uncertainty is allocated across generation trajectories (see section 3). Since $\mathrm{CE}^\star$ exhausts the entire space of rollouts, a key challenge is to estimate it efficiently. Towards that end, we develop Monte Carlo estimators and establish their consistency (see section 4).

Empirically, we evaluate multiple model families across sentence completion, mathematical reasoning, coding, and story generation, while explicitly controlling for output length. Our results reveal three consistent findings: **(i)** fine-tuning generally reduces token-level and trajectory-level uncertainty, although the magnitude of reduction is highly task-dependent; **(ii)** fine-tuning systematically shifts $\rho(N, r_N)$ from negative toward positive, indicating that longer generations become more informative rather than being redundant as often seen in base models; and **(iii)** entropy rate becomes a substantially stronger predictor of semantic diversity after fine-tuning, suggesting that aligned models utilize uncertainty more efficiently and translate it more effectively into meaningful semantic diversity (see section 5). Overall, our results suggest that fine-tuning does not simply reduce uncertainty or shrink the generation space. Rather, it restructures how uncertainty is allocated across generation trajectories, producing outputs that are more organized, semantically meaningful, and information-efficient.

2. Related Work

Length as a Fundamental Confounder in Diversity Measurement. Recent work (Singhal et al., 2023) shows that RLHF and instruction tuning systematically increase LLM output length, since longer responses are often positively correlated with human preference scores. This makes output length a critical confounder in diversity evaluation. Metrics that favor shorter generations may therefore unfairly penalize instruct models simply because they produce longer responses. For example, n-gram diversity metrics (Li et al., 2016) naturally decrease with sequence length, as longer texts inevitably reuse a finite vocabulary, reducing the ratio of unique n-grams even when the underlying content remains semantically diverse. Embedding-based semantic diversity measures (Tevet & Berant, 2021) suffer from a related issue: longer generation trajectories must be compressed into fixed-dimensional representations, often through mean pooling, which can dilute semantic diversity. Moreover, many existing diversity metrics rely on heuristic sampling procedures and lack a unified probabilistic foundation (West & Potts, 2025). Consequently, current approaches cannot rigorously distinguish whether the increased verbosity of instruct models reflects genuine information gain or merely redundant continuation, motivating the need for a principled length-aware and model-intrinsic measure of generation diversity.

From Local Branching to Global Canopy. Conceptualizing autoregressive generation as a stochastic, path-dependent branching tree, Branching Factor (BF) (Yang et al., 2025b) was introduced to quantify the probability concentration induced by model alignment. However, their framework lacks an intuitive heuristic for interpreting the measure and focuses primarily on the *width* of the generation tree (branching) while ignoring its *depth* (length) and the coupling between the two. $\mathrm{CE}^\star$ addresses this gap by providing a model-intrinsic characterization of the gener-

ation space by jointly modeling content uncertainty and stochastic stopping behavior. By deriving the BF as a normalized version of $\mathrm{CE}^\star$, we provide a principled theoretical foundation for its use. Crucially, our length–entropy correlation demonstrates that while fine-tuning reduces the absolute BF, it restructures the generation tree to sustain information density across longer paths while translating uncertainty into semantic diversity more efficiently. To the best of our knowledge, this is the first work to provide a unified measure of the global LLM generation space that accounts for both the local branching uncertainty and the stochasticity of trajectory length, while also studying the redistribution of uncertainty across generation trajectories.

3. Methodology

Notation Convention. For any discrete random variable Z with probability $q(z)$ for each $z \in \mathcal{Z}$, its Shannon entropy (Shannon, 1948) is $H(Z) = -\sum_{z\in\mathcal{Z}} q(z)\log q(z)$. Let X be another random variable and X, Z have joint probability $q(z, x)$, then $H(Z|X = x) = -\sum_{z\in\mathcal{Z}} q(z|x)\log q(z|x)$ is the conditional entropy of Z given realized x, whereas $H(Z|X) = -\sum_x H(Z|x)q(x)$ is the expected conditional entropy of Z.

3.1. Measuring the Size of LLM Generation Space From a Tree Perspective

Autoregressive Generation as a Stochastic Tree. Autoregressive language models induce a stochastic generation process that can naturally be represented as a rooted, path-dependent tree. Let $X \sim p_X$ denote a prompt sampled from a prompt distribution over a space $\mathcal{X}$, and let $\mathcal{V}$ denote the vocabulary. Conditioned on $X = x$ and a decoding policy (e.g., temperature, top-k, or nucleus sampling), generation proceeds sequentially by sampling tokens from a conditional distribution. Formally, let $Y_1, Y_2, \ldots,$ denote the autoregressive token process generated by the model. For each step $t \geq 1$, define the generation history as $G_t := (Y_1, \ldots, Y_{t-1})$, with $G_1 := \emptyset$. The next-step random variable is denoted by $Z_t \in \mathcal{Z} := \mathcal{V} \cup \{\texttt{EOS}\}$, where EOS denotes termination. The conditional distribution $q(Z_t \mid G_t, X)$ governs the next-step generation process.[1] Under this construction, each node of the generation tree (see Figure 1) corresponds to a history G_t. Each outgoing edge corresponds to a possible realization of Z_t. A complete generated sequence therefore corresponds to a root-to-leaf trajectory. A key feature of this construction is its path dependence. The next-step distribution q depends on the entire generation history G_t, not merely the most recent token. Thus, two trajectories ending with the same token may evolve very differently if they arise from different histories.

[1]This formulation is equivalent to a two-stage stochastic decision process. Specifically, the model first makes a binary decision between STOP (terminate) and CONT (continue). Conditional on CONT, it then samples the next token from the vocabulary $\mathcal{V}$. We provide a formal proof of this equivalence in subsection A.1.

Width and Depth of the Generation Tree. The size of the generation tree is determined by two complementary factors. First, at each history G_t, the spread of $q(\cdot \mid G_t, X)$ determines the number of plausible continuations, capturing the local branching behavior (*width*) of the tree. Second, generation proceeds for a random number of steps before termination, inducing variability in trajectory depth. We define the stopping time $N := \inf\{t \geq 1 : Z_t = \texttt{EOS}\}$, which represents the generation length (*depth*). Under this formulation, the stochastic process jointly models both continuation uncertainty and stopping behavior, together determining the overall size of the generation space.

A Unified Measure of Generation Space. To quantify local uncertainty, we use Shannon entropy (Shannon, 1948). For a given prompt $X = x$ and a realized generation history $G_t = g_t$, define

$$\begin{aligned}\widetilde{H}(g_t \mid x) &:= H(Z_t \mid G_t = g_t, X = x) \\ &= -\sum_{z\in\mathcal{Z}} q(z \mid g_t, x)\log q(z \mid g_t, x).\end{aligned} \tag{1}$$

This quantity measures uncertainty via entropy in the immediate next generation step, including both continuation-token selection and termination. Note that this quantity summarizes only the uncertainty of the next-step generation conditioned on a realized history g_t and prompt x. In the following, we slightly overload the notation by also using uppercase G_t and X to denote realized histories and prompts whenever the distinction between random variables and their realizations is clear from context. To jointly account for local branching diversity and random trajectory length, we introduce the following uncertainty measure.

Definition 1 (Canopy Entropy)**.** *We define* Canopy Entropy *as* $\mathrm{CE}^\star := \mathbb{E}\left[\sum_{t=1}^{N(x)} \widetilde{H}(g_t \mid x)\right]$. *More formally,*

$$\mathrm{CE}^\star = \mathbb{E}_{x,\{g_t\}_{t=1}^{N(x)}}\left[\sum_{t=1}^{N(x)} H(Z_t \mid G_t = g_t, X = x)\right]$$

where the $\mathbb{E}$ *is over the randomness of prompt* x*,* $N(x)$ *(the length), and* $\{g_t\}_{t=1}^{N(x)}$ *(all possible partial histories of rollouts).*

That is, the quantity $\mathrm{CE}^\star$ aggregates entropy-based uncertainty along generation trajectories and across prompts. A single generated sequence corresponds to a specific traversal from the tree root (prompt X) to a leaf (EOS termination). The summation $\sum_{t=1}^{N(X)}$ thus represents the total uncertainty accumulated along that entire path. By taking the expectation over all trajectories, $\mathrm{CE}^\star$ captures the aggregate "canopy" of the model's output distribution across

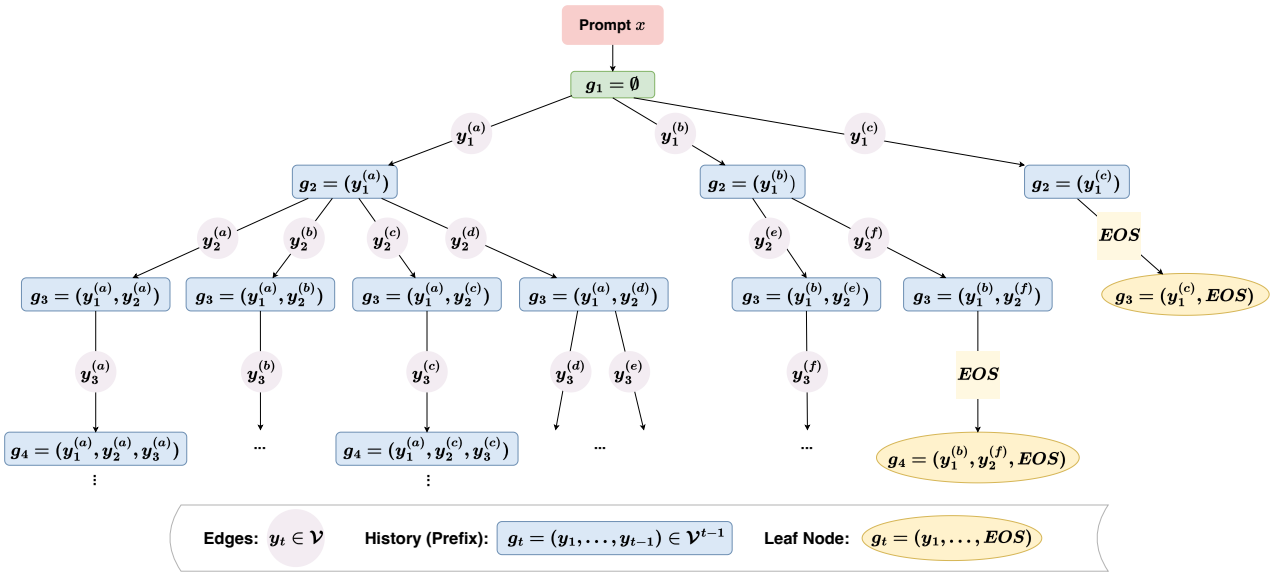

Figure 1. Path-dependent generation tree induced by autoregressive LLMs.

all possible rollouts. As such, it provides a unified measure of generation space by accounting for both *local branching width* (the diversity of choices at each node) and *trajectory depth* (the variability in generation length).

CE* as the Joint Entropy of Length and Content. Besides being an intuitive measure, the above construction turns out to also admit a clean information-theoretic characterization. In particular, the cumulative uncertainty along a trajectory can be interpreted as the total entropy of the generation process. For ease of reference, we call $H(N, Y_{1:N}|X)$ the *total uncertainty*, $H(N|X)$ the *length uncertainty*, and $H(Y_{1:N} \mid N, X)$ the *content uncertainty*. The following theorem formalizes the connection.

Theorem 1 (Equivalence of CE* and total uncertainty)**.** $\mathrm{CE}^\star = H(N, Y_{1:N} \mid X)$.

Theorem 1 (see subsection A.2 for a detailed proof) shows that the Canopy Entropy coincides with the total uncertainty of the generation process. By the chain rule of entropy, $H(N, Y_{1:N} \mid X) = H(N \mid X) + H(Y_{1:N} \mid N, X)$, which decomposes the total uncertainty into two components. The *length uncertainty* $H(N \mid X)$ captures how unpredictable the stopping time is, while the *content uncertainty* $H(Y_{1:N} \mid N, X)$ captures how diverse the generated sequences are once the length is fixed. This decomposition aligns naturally with the tree perspective: length uncertainty corresponds to variability in the depth of trajectories, while content uncertainty corresponds to the diversity of paths of a given length. We can interpret $\exp(\mathrm{CE}^\star)$ as the effective number of plausible full sequences, analogous to how perplexity represents the effective branching at the token level.

3.2. Two Normalized Variants of Canopy Entropy

While $\mathrm{CE}^\star = H(N, Y_{1:N} \mid X)$ provides a clean characterization of the generation space, it aggregates uncertainty across both sequence lengths and content diversity, making its direct interpretation difficult. To obtain more interpretable metrics, we introduce two normalized variants of CE*, respectively coined *Generation Perplexity* and *Branching Factor* (as introduced in (Yang et al., 2025b)). Intuitively, the former corresponds to the token-level normalization of CE*, whereas the latter is trajectory-level normalization. In both definitions, the exp function converts entropy uncertainty to effective diversity.

Definition 2 (GenPPL)**.** *Generation Perplexity is defined as* $\mathrm{GenPPL} := \exp\left(\frac{\mathrm{CE}^\star}{\mathbb{E}[N]}\right)$*, which is equivalent to* $\exp\left(\frac{\sum_{X\sim p_X} H(N, Y_{1:N}|X=x)}{\sum_{X\sim p_X} \mathbb{E}[N|X=x]}\right)$.

Definition 3 (BF, (Yang et al., 2025b))**.** *Branching Factor is defined as* $\mathrm{BF} := \exp\left(\mathbb{E}_X \mathbb{E}_N[r_N \mid X = x]\right)$*, where* $r_N := \frac{H(Y_{1:N}|N=n, X=x)}{n}$ *is the entropy rate (i.e., per-token content uncertainty).*[2]

Invoking Theorem 1, we observe that the only difference between Canopy Entropy CE* and GenPPL or BF is that – excluding the "exp" function – CE* does not have the normalization term $1/(\sum_{X\sim p_X} \mathbb{E}[N \mid X = x])$ or $1/n$. It is worthwhile to think more deeply about the difference between GenPPL and BF, due to their different normalization factors, which will be prominent when the generated sequence length N can differ a lot. In such a case, GenPPL

[2]In information theory, the entropy rate quantifies the average amount of information produced per generated token by a stochastic process, and can be interpreted as a measure of information density or non-redundancy.

treats each token equally, regardless of whether it comes from a short or long sequence, through a "global" normalization factor $1/(\sum_{X\sim p_X} \mathbb{E}[N \mid X = x])$. However, BF as the expectation of r_N *treats each trajectory equally* and consequently will discount the influence of tokens in long sequences much more than those in short sequences. Therefore, at a high level, GenPPL reflects the diversity of a *typical token* in the generation whereas BF reflects the diversity of a *typical trajectory*. For further intuition, we refer readers to subsection B.1 for a concrete example illustrating the distinction between GenPPL and BF.

GenPPL and BF together provide a more complete view of generation behavior across tasks. In particular, we observe that structured tasks such as mathematical reasoning or coding are better characterized by token-level uncertainty (GenPPL), whereas open-ended tasks like story generation are better captured by trajectory-level diversity (BF). Our framework provides a principled formulation of GenPPL as a generative analogue of perplexity (Jelinek et al., 1977) for variable-length processes, and offers a unified information-theoretic interpretation of BF, initially proposed by Yang et al. (2025b), showing that both arise naturally as normalized variants of Canopy Entropy. See Appendix B.2 for guidance on when to use BF or GenPPL, and Appendix B.3 for a comparison with perplexity.

3.3. The Length–Entropy Correlation: How Entropy Evolves with Generation Length

Aggregate measures such as Canopy Entropy, GenPPL, and BF quantify the overall magnitude of uncertainty. However, they fail to capture how uncertainty evolves across generation rollouts or how effectively uncertainty is translated into meaningful variation as generation length increases. Whether longer generations remain information-rich or become progressively redundant (Singhal et al., 2023; Shaib et al., 2024) remains poorly understood despite rich previous studies on the diversity of language generation (Tevet & Berant, 2021; Li et al., 2016; Friedman & Dieng, 2022). Understanding this interaction is crucial for characterizing the *information-conveying efficiency* of language models—namely, how much uncertainty is maintained per token as generation progresses.

To study this phenomenon, we introduce the prompt-controlled correlation $\rho(N, r_N)$ between the length N and entropy rate r_N, defined below.

Definition 4 (The length–entropy correlation)**.** *For each prompt, the length–entropy correlation is defined as*

$$\rho(N, r_N) = \mathbb{E}_{X\sim p_X}\left\{\texttt{Corr}(N, r_N \mid X = x)\right\},$$

where $r_N = \frac{H(Y_{1:N}|N=n,X=x)}{n}$ *is the entropy rate and* $\texttt{Corr}$ *is any standard correlation notion from the set {Pearson correlation, Spearman correlation, Kendall correlation}.*

A positive correlation $\rho(N, r_N)$ indicates that longer sequences remain informative per token, while a negative correlation indicates that longer outputs become increasingly predictable. This quantity characterizes uncertainty allocation along trajectories and is later used in section 5 to compare base and fine-tuned models across tasks and model families. In our evaluation, we observe consistent trends across the three standard correlation notions as mentioned in the definition above, which illustrates the robustness of the discovered correlations among generation length and entropy rate. Notably, the correlation $\rho(N, r_N)$ is evaluated over the randomness of rollouts drawn from an exponentially large space; we discuss efficient estimation methods in the section below, with full details deferred to subsection A.6.

4. Efficient and Consistent Estimation of $\mathrm{CE}^\star$

The previous section introduced $\mathrm{CE}^\star = H(N, Y_{1:N} \mid X)$ and its derived metrics such as GenPPL, BF and length–entropy correlation. However, efficiently computing these metrics in practice is challenging since they are expectations over exponentially many trajectories (Hu et al., 2023; Yang et al., 2025b). This section develops computationally tractable approaches to estimate $\mathrm{CE}^\star$ which are also generalizable to estimating the derived metrics. In particular, we introduce a Monte Carlo estimator under a maximum length constraint $T_{\max}$, and analyze the consistency of this estimator.

For a random prompt X, let $N(X)$ be the generation's (random) stopping time. Given a maximum length $T_{\max}$, define the truncated stopping time $N_{\max}(X) := \min\{N(X), T_{\max}\}$. The corresponding truncated functional is $\mathrm{CE}^\star_{\max} := \mathbb{E}\left[\sum_{t=1}^{N_{\max}(X)} \widetilde{H}(G_t \mid X)\right]$. Now suppose $x_1, \ldots, x_P$ are sampled i.i.d. from p_X, the prompt distribution. For each prompt x_p, we generate M independent rollouts. For the i-th rollout, define the truncated cumulative local entropy by $S^{(p,i)}_{\max} := \sum_{t=1}^{N^{(p,i)}_{\max}} \widetilde{H}(G^{(p,i)}_t \mid x_p)$, which measures the total branching uncertainty along the generated trajectory up to truncation. We then define the Monte Carlo estimator (see Algorithm 1 for more detail)

$$\widehat{\mathrm{CE}}^\star_{\max,P,M} := \frac{1}{PM}\sum_{p=1}^{P}\sum_{i=1}^{M} S^{(p,i)}_{\max}.$$

The estimator's convergence and consistency require minor technical assumptions. First, let $g(T_{\max}) = \mathrm{CE}^\star - \mathrm{CE}^\star_{\max}$ denote the truncation bias. Naturally, it always satisfies $g(T_{\max}) \geq 0$. Convergence requires **diminishing truncation bias** — i.e., $g(T_{\max}) \to 0$ as $T_{\max} \to \infty$. Second, in our estimation, $T_{\max}$ increases as parameter P, M increases. Convergence requires **modest truncation length**

— i.e., the length cap $T_{\max}(P, M)$ is picked such that $\nu^2_{\max}(\frac{1}{P} + \frac{1}{PM}) \to 0$ as $T_{\max} = T_{\max}(P, M) \to \infty$, where $\nu^2_{\max} := \mathbb{E}\left[N_{\max}(X)^2\right]$ is the second raw moment of the truncated stopping time. Both conditions are easily satisfied when the tail of N is not too heavy, which we observe in our empirical evaluation (see Figure 3). The following theorem characterizes the convergence of $\widehat{\mathrm{CE}}^{\star}_{\max,P,M}$ under these technical conditions.

Theorem 2 (Consistency and non-asymptotic error bounds of the Monte Carlo estimator)**.** *Assume diminishing truncation bias $g(T_{\max})$ and modest truncation length. Then $\widehat{\mathrm{CE}}^{\star}_{\max,P,M}$ converges to $\mathrm{CE}^{\star}$ in probability when $P, M \to \infty$. Moreover, $\left|\widehat{\mathrm{CE}}^{\star}_{\max,P,M} - \mathrm{CE}^{\star}\right| = O_{\mathbb{P}}\left(\nu_{\max}\sqrt{\frac{1}{P} + \frac{1}{PM}} + g(T_{\max})\right)$, where $\nu^2_{\max} := \mathbb{E}\left[N_{\max}(X)^2\right]$.*

Theorem 2 shows that the estimator $\widehat{\mathrm{CE}}^{\star}_{\max,P,M}$ is consistent, with estimation error naturally decomposing into a Monte Carlo error term and a truncation bias term. The Monte Carlo error is of order $O_{\mathbb{P}}\left(\frac{1}{\sqrt{P}} + \frac{1}{\sqrt{PM}}\right)$, arising from sampling P prompts and M rollouts per prompt.

The second component is a truncation bias $g(T_{\max})$, which captures the bias introduced by truncating trajectories at length $T_{\max}$ instead of allowing full generations. Empirically, we observe that the generation length distribution of LLMs typically exhibits exponential decay across tasks and model architectures (see Figure 3). This implies that long trajectories are increasingly rare, so the truncation bias $g(T_{\max})$ decays rapidly with $T_{\max}$. We defer the proof of Theorem 2 to Appendix A.5.

Notably, the same Monte Carlo framework extends to the estimation of GenPPL, BF, and prompt-controlled correlation. Their consistency follows by the continuous mapping theorem (Mann & Wald, 1943) together with mild integrability and truncation-bias conditions, which are empirically supported by the observed low truncation rates and rapidly decaying active-rollout distributions (see Appendix A.6 for detailed proofs and discussions).

Remark 4.1 (Connecting truncation bias to the distribution tail of N)**.** *The truncation bias can be expressed as*

$$g(T_{\max}) = \mathrm{CE}^{\star} - \mathrm{CE}^{\star}_{\max} = \mathbb{E}\Big[\sum_{t=T_{\max}+1}^{N(X)} \widetilde{H}(G_t \mid X)\mathbf{1}\{N(X) > T_{\max}\}\Big]. \tag{2}$$

Let $H_{\max}$ denote the entropy upper bound, i.e., $\widetilde{H}(G_t \mid X) \le H_{\max}$ for any G_t. We then have $g(T_{\max}) \le H_{\max}\mathbb{E}_{X\sim p_X}\left[\mathbb{E}\left[(N(X) - T_{\max})_+ | X\right]\right]$ by the tail-sum formula $\mathbb{E}[(N(x) - T_{\max})_+ \mid X = x] = \sum_{t=T_{\max}+1}^{\infty}\mathbb{P}(N(x) \ge t \mid X = x)$. Therefore, $H_{\max}\mathbb{E}_{X\sim p_X}\left[\sum_{t=T_{\max}+1}^{\infty}\mathbb{P}(N(X) \ge t \mid X)\right]$, with $H_{\max} := \log|\mathcal{Z}|$, is a natural upper bound of $g(T_{\max})$, showing that the truncation bias is governed by the prompt-averaged tail behavior of the stopping time. We refer readers to subsection A.4 for examples of prompt-averaged bias decay rates.

5. Empirical Studies

In this section, we present our experimental results, organized into two complementary parts. First, we examine the global uncertainty structure of the LLM generation space using the metrics introduced in section 3, including $\mathrm{CE}^{\star}$, GenPPL, BF, and length–entropy correlation. Second, building on this analysis, we investigate how uncertainty is translated into semantic diversity through a regression framework that links entropy rate to semantic diversity. We evaluate three model families—Llama-3.1-8B (Grattafiori et al., 2024), Qwen3-8B (Yang et al., 2025a), and Gemma-3-12B (Team et al., 2025)—testing both their pre-trained (base) and instruction-tuned (instruct) versions. We evaluate these models on four tasks spanning a spectrum from highly structured settings, including mathematical reasoning (Cobbe et al., 2021), coding (Zhuo et al., 2024), and sentence completion (Zellers et al., 2019), to more open-ended creative story generation (Ismayilzada et al., 2025). For each task, we sample $P = 100$ prompts, and for each sampled prompt X_p, we generate $M = 100$ independent rollouts using stochastic decoding with temperature equal to 1 and top-$k = 100$ (renormalized next-token probabilities),[3] where each rollout produces a sequence $Y_{1:N}$ with stopping time N up to a maximum of 4000 new tokens, i.e. $T_{\max} = 4000$ (see subsection C.1 for detailed description). To ensure consistent evaluation across model families, we disable the thinking mode in Qwen3-8B Instruct.

5.1. Global Effects of Fine-Tuning on Uncertainty

Across all model families and tasks, we observe three consistent patterns.

Fine-Tuning Reduces Both Token-Level and Trajectory-Level Normalized Uncertainty. Instruction fine-tuning consistently reduces both token-level (measured by GenPPL) and trajectory-level (measured by BF) normalized uncertainty, as evidenced by systematic declines in GenPPL and BF across nearly all models and tasks (see Table 1). Nevertheless, the magnitude of this reduction varies substantially across settings. For example, Qwen3-8B on mathematical reasoning exhibits only a mild decrease of

[3] Specifically, the one step local entropy $\tilde{H}(G_t|X)$ is computed as $-\sum_{z\in\mathcal{Z}_{100}} q(z|G_t, X)\log q(z|G_t, X)$, where $\mathcal{Z}_{100}$ is the set of the top-100 tokens (including EOS if present) and q is the renormalized distribution over the sampled subset.

Table 1. Task-dependent changes in generation-space uncertainty after instruction tuning across different model families. We report three complementary uncertainty measures: GenPPL, BF, and $\mathrm{CE}^\star$ (see subsection A.6). For each task and model family, we compare the base and instruct variants and report the relative percentage change after instruction tuning. Standard errors (SE) are estimated via prompt-cluster bootstrapping (2K iterations, resampled with replacement) as in subsection C.3. Blue cells indicate reductions in uncertainty after instruction tuning, while red cells indicate increases. Darker color intensity corresponds to larger magnitude changes.

Task	Type	GenPPL			BF			$\mathrm{CE}^\star$		
		Qwen3-8B	Llama-3.1-8B	Gemma-3-12B	Qwen3-8B	Llama-3.1-8B	Gemma-3-12B	Qwen3-8B	Llama-3.1-8B	Gemma-3-12B
Math	Base	1.23 ± 0.0052	3.57 ± 0.041	4.56 ± 0.052	1.23 ± 0.0064	5.31 ± 0.12	4.98 ± 0.055	55.7 ± 2.0	458 ± 18	487 ± 11
	Instruct	1.17 ± 0.0067	1.74 ± 0.063	1.17 ± 0.0098	1.16 ± 0.0057	1.57 ± 0.024	1.15 ± 0.0054	40.5 ± 2.5	98.7 ± 9.1	59.1 ± 5.1
	% Change	$-4.44 \pm 0.47\%$	$-51.2 \pm 2.0\%$	$-74.4 \pm 0.34\%$	$-5.76 \pm 0.38\%$	$-70.5 \pm 0.73\%$	$-76.9 \pm 0.24\%$	$-27.3 \pm 3.3\%$	$-78.4 \pm 2.2\%$	$-87.9 \pm 1.1\%$
Coding	Base	1.68 ± 0.017	3.16 ± 0.055	2.60 ± 0.041	2.08 ± 0.034	4.96 ± 0.057	3.24 ± 0.038	207 ± 4.8	326 ± 13	492 ± 25
	Instruct	1.23 ± 0.0053	1.31 ± 0.0088	1.05 ± 0.0039	1.21 ± 0.0061	1.31 ± 0.0081	1.05 ± 0.0021	81.2 ± 4.0	135 ± 4.3	17.7 ± 1.6
	% Change	$-27.1 \pm 0.79\%$	$-58.4 \pm 0.74\%$	$-59.7 \pm 0.64\%$	$-41.9 \pm 0.99\%$	$-73.7 \pm 0.35\%$	$-67.8 \pm 0.38\%$	$-60.7 \pm 1.6\%$	$-58.6 \pm 1.7\%$	$-96.4 \pm 0.39\%$
Sentence Completion	Base	4.77 ± 0.16	6.73 ± 0.12	8.81 ± 0.11	4.20 ± 0.11	7.80 ± 0.15	9.49 ± 0.11	826 ± 55	969 ± 40	1084 ± 36
	Instruct	1.68 ± 0.025	2.56 ± 0.069	1.80 ± 0.019	1.55 ± 0.018	2.75 ± 0.045	1.74 ± 0.024	112 ± 11	201 ± 11	515 ± 29
	% Change	$-64.9 \pm 1.1\%$	$-62.0 \pm 1.3\%$	$-79.6 \pm 0.30\%$	$-63.0 \pm 0.88\%$	$-64.8 \pm 0.83\%$	$-81.6 \pm 0.30\%$	$-86.5 \pm 1.0\%$	$-79.3 \pm 1.3\%$	$-52.5 \pm 2.8\%$
Story Generation	Base	3.57 ± 0.11	7.40 ± 0.071	7.98 ± 0.062	4.03 ± 0.092	8.24 ± 0.073	8.68 ± 0.070	445 ± 20	1320 ± 41	692 ± 23
	Instruct	2.28 ± 0.037	3.24 ± 0.092	2.22 ± 0.023	2.11 ± 0.039	3.37 ± 0.083	2.17 ± 0.025	454 ± 28	543 ± 22	925 ± 40
	% Change	$-36.0 \pm 1.3\%$	$-56.2 \pm 1.4\%$	$-72.2 \pm 0.28\%$	$-47.6 \pm 1.3\%$	$-59.1 \pm 1.2\%$	$-75.0 \pm 0.28\%$	$1.92 \pm 3.5\%$	$-58.9 \pm 1.6\%$	$33.7 \pm 8.3\%$

roughly 5%, whereas many other model–task combinations show much larger reductions, often in the range of 60%–80%. These results suggest that fine-tuning contracts the effective generation space at both the token and trajectory levels, producing outputs that are more concentrated and deterministic. At the same time, the heterogeneous magnitude of the reduction suggests that alignment does not impose a uniform compression of uncertainty. Instead, the extent of contraction depends on the underlying task structure and model behavior, a phenomenon we discuss in the following paragraph. Overall, our findings are consistent with prior observations that alignment reduces diversity and increases probability concentration (Wang et al., 2025; Yang et al., 2025b; Lake et al., 2025; West & Potts, 2025), although the shrinkage we observe is less extreme than the order-of-magnitude reductions reported in the previous study (Yang et al., 2025b).

The Effect of Fine-Tuning on Canopy Entropy is *Task-Dependent* and Could Exhibit Opposite Trends. In structured domains such as math, coding, and sentence completion, $\mathrm{CE}^\star$ decreases significantly after fine-tuning. In contrast, story generation exhibits a different pattern: Qwen3-8B shows a slight increase of roughly 2%, while Gemma-3-12B shows a much larger increase of about 34% (see Table 1). This suggests that fine-tuning does not uniformly reduce canopy entropy, but instead behaves differently in constrained versus open-ended generation settings.

For deterministic tasks such as math and coding, the valid generation space is naturally constrained, so fine-tuning acts as a pruning force that collapses the model's generation space into a small set of correct trajectories. Consequently, both stopping-time variance and local token-level entropy are significantly reduced. By contrast, on creative tasks, fine-tuned models exhibit a notable rightward shift in sequence length distributions (see Figure 3), reflecting a tendency toward longer and more elaborative generations. In the same domain, the corresponding base models frequently fall into "completion traps," treating prompts as short text continuation tasks rather than directly answering the underlying questions (see subsection C.2).

Although instruction tuning reduces local branching diversity, it also increases stopping-time uncertainty, and sustains generation over much deeper trajectories. As illustrated in Figure 2, while the base model has higher local entropy rates (*width*), it lacks enough trajectory length (*depth*) to accumulate a large Canopy Entropy ($\mathrm{CE}^\star$). Since $\mathrm{CE}^\star$ acts as a measure of the total *volume* of the generation tree by accumulating uncertainty across the entire rollout, including stopping behavior, the increase in trajectory depth can mathematically outweigh the reduction in local branching.

These findings suggest that fine-tuning changes not only the magnitude of uncertainty, but also how uncertainty is distributed across the trajectory, particularly through its interaction with generation length and stopping behavior.

Fine-Tuning Consistently Increases Length–Entropy Correlation. To better understand how uncertainty is distributed across trajectories, we analyze the relationship between output length N and entropy rate r_N. Empirically, fine-tuning consistently shifts the correlation (see Definition 4) between N and r_N from negative toward positive values in structured tasks, while making the correlation less negative in open-ended tasks such as story generation. Table 2 shows the concrete results under Pearson correlation. Similar results based on Spearman correlation and Kendall's τ exhibit the same trends, and are deferred to Table 6 in the appendix to avoid repetition.

This reveals a fundamental change in generation behavior. In base models, longer outputs are often associated with lower entropy rates, meaning later tokens become

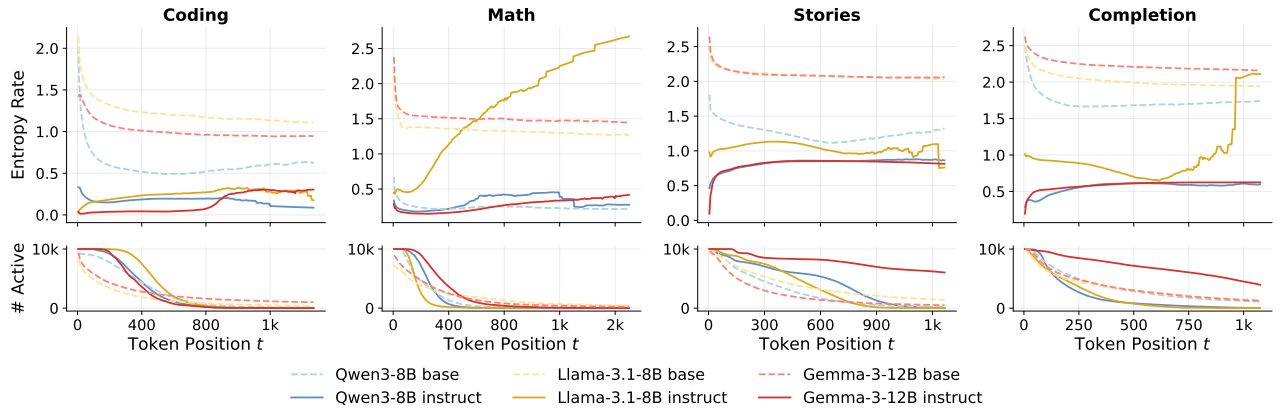

Figure 2. Running entropy rate vs. token position. We plot mean running entropy rate $\bar{R}_{\leq t} = \frac{1}{t}\sum_{s=1}^{t} H(Z_s \mid X, G_s)$ averaged across active rollouts at position t, with the corresponding active rollout count beneath. Dashed curves denote base models and solid curves denote fine-tuned models.

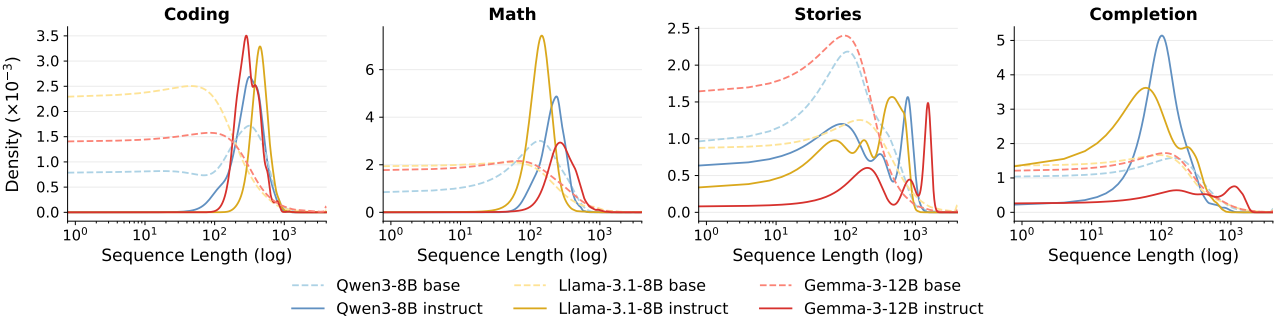

Figure 3. Kernel density estimates of generated sequence lengths across task domains with sequence length shown on a logarithmic scale.

increasingly redundant and contribute little new information. In contrast, fine-tuned models exhibit much higher length–entropy correlation: longer generations tend to sustain higher entropy rates and remain informative throughout the trajectory (see Figure 2).

The distinction is especially important in the presence of length confounding. As shown in Figure 3, fine-tuned models generally produce longer outputs than their base counterparts, consistent with observations in prior work (Singhal et al., 2023). Under conventional diversity metrics that do not explicitly account for length (Tevet & Berant, 2021; Li et al., 2016), these longer generations can artificially appear less diverse due to normalization effects or repetition accumulation, potentially biasing diversity comparisons (Friedman & Dieng, 2022; Shaib et al., 2024). The positive length–entropy correlation observed here demonstrates that increased length in fine-tuned models reflects sustained information content rather than simple redundancy.

Overall, these results suggest that fine-tuning does not merely reduce the quantity of uncertainty, but also improves its organization and efficiency. Rather than uniformly suppressing diversity, fine-tuning restructures uncertainty to better align with task requirements – reducing unnecessary randomness while preserving informative variation across longer trajectories.

5.2. Fine-Tuning More Effectively Converts Token Uncertainty Into Semantic Diversity

After establishing how fine-tuning reshapes entropy allocation, we next investigate how these changes in entropy rate translate into semantic diversity (Guo et al., 2025). Specifically, we study how generation uncertainty is converted into semantic diversity across sampled trajectories.

For each task–prompt–model tuple (t, p, m), we measure semantic diversity by computing the average pairwise cosine distance between generated outputs: $D_{tpm} = \frac{2}{M(M-1)}\sum_{i<j} d(e_{t,p,m,i}, e_{t,p,m,j})$ (Tevet & Berant, 2021), where $e_{t,p,m,i}$ denotes the embedding of the i-th Monte Carlo rollout, prepended with the retrieval prefix (`"search_document:"`) to ensure alignment with the encoder's training objective for ModernBERT embedding model (modernbert-embed-large) (Warner et al., 2024). Here $d(\cdot,\cdot)$ denotes cosine distance. Since cosine distance is bounded in $(0, 1)$, we employ the Beta mixed-effects regression, which is a standard method to accommodate bounded responses while allowing flexible modeling of dispersion and hierarchical variation (Figueroa-Zúñiga et al., 2013).[4]

[4]In contrast, Gaussian linear models are inappropriate for bounded responses and exhibit clear heteroscedasticity and residual misspecification in diagnostic analyses.

Table 2. Base–instruct comparison of the length–entropy Pearson correlation, measured by $\rho(N, r_N)$ (see Definition 4), for the Qwen3-8B, Llama-3.1-8B, and Gemma-3-12B model families. Each entry reports the estimate $\pm$ standard error, where the absolute change is computed as the instruct value minus the corresponding base value.

Metric	Model	MATH			CODING			SENTENCE COMPLETION			STORY GENERATION		
		Base	Instruct	Abs. Change	Base	Instruct	Abs. Change	Base	Instruct	Abs. Change	Base	Instruct	Abs. Change
$\rho(N, r_N)$	Qwen	-0.018 $\pm$ 0.016	0.087 $\pm$ 0.037	0.11 $\pm$ 0.042	-0.41 $\pm$ 0.014	0.61 $\pm$ 0.034	1.0 $\pm$ 0.037	0.062 $\pm$ 0.017	0.52 $\pm$ 0.024	0.46 $\pm$ 0.027	-0.27 $\pm$ 0.017	0.16 $\pm$ 0.021	0.43 $\pm$ 0.027
	Llama	-0.27 $\pm$ 0.016	0.38 $\pm$ 0.040	0.65 $\pm$ 0.044	-0.38 $\pm$ 0.011	0.29 $\pm$ 0.021	0.67 $\pm$ 0.021	-0.26 $\pm$ 0.015	-0.20 $\pm$ 0.024	0.053 $\pm$ 0.028	-0.24 $\pm$ 0.012	-0.10 $\pm$ 0.020	0.14 $\pm$ 0.022
	Gemma	-0.13 $\pm$ 0.014	0.19 $\pm$ 0.040	0.32 $\pm$ 0.043	-0.27 $\pm$ 0.012	0.39 $\pm$ 0.035	0.67 $\pm$ 0.037	-0.23 $\pm$ 0.015	0.12 $\pm$ 0.038	0.36 $\pm$ 0.041	-0.17 $\pm$ 0.012	-0.049 $\pm$ 0.027	0.12 $\pm$ 0.032

Formally, we model

$$D_{tpm} \sim \mathrm{Beta}(\mu_{tpm}\phi_{tpm}, (1-\mu_{tpm})\phi_{tpm}),$$

where $\mu_{tpm} \in (0,1)$ denotes the conditional mean and $\phi_{tpm} > 0$ is the precision parameter. The conditional mean model is specified as $\mathrm{logit}(\mu_{tpm}) = \alpha + \beta_1 R_{tpm} + \beta_2 \mathbf{1}\{m = \mathrm{FT}\} + f(\mathrm{invN}_{tpm}) + \sum_k \tau_k \mathbf{1}\{t = k\} + \beta_3 (R_{tpm} \cdot \mathbf{1}\{m = \mathrm{FT}\}) + g(\mathrm{invN}_{tpm})\,\mathbf{1}\{m = \mathrm{FT}\} + u_p + v_m$, where R_{tpm} is the entropy rate, invN_{tpm} is the inverse sequence length, and $f(\cdot)$, $g(\cdot)$ are natural spline functions corresponding to nonlinear length effects and their interaction with instruction tuning. For consistency with the measurement of semantic diversity, the reported sequence length and entropy rate are likewise computed by averaging over the same set of M rollouts for each task–prompt–model combination. The coefficients τ_k represent fixed task effects. To account for hierarchical dependence, we include random intercepts $u_p \sim \mathcal{N}(0, \sigma_p^2), v_m \sim \mathcal{N}(0, \sigma_m^2)$, for prompt identity and model family, respectively. To capture heteroscedasticity, we also model the precision parameter ϕ_{tpm} as a function of task, model variant, entropy rate, sequence length, and model family. Residual diagnostics (see Figure 6) indicate good calibration of the mean structure (Kolmogorov-Smirnov Test (Massey Jr, 1951): $p = 0.55$) and no significant dispersion issues ($p = 0.068$) and no outliers ($p = 0.16$).

Main Findings. First, entropy rate exhibits a strong and highly significant positive association with semantic diversity. The coefficient on entropy rate is large and significant ($\beta_1 = 0.708$, $p < 2 \times 10^{-16}$), indicating that higher per-token uncertainty leads to substantially greater semantic diversity across generated outputs. More importantly, the interaction between entropy rate and fine-tuning is strongly positive ($\beta_3 = 1.183$, $p < 2 \times 10^{-16}$). Under the logit link, this implies a substantial amplification of the slope: $\partial\,\mathrm{logit}(\mu)/\partial R = \beta_1 = 0.708$ (base), $\beta_1 + \beta_3 = 0.708 + 1.183 = 1.891$ (fine-tuned). Thus, fine-tuning nearly triples the sensitivity of semantic diversity to entropy rate. This indicates that uncertainty in fine-tuned models is significantly more *semantically productive*: small increases in token-level uncertainty translate into much larger increases in trajectory-level diversity.

At the same time, the main effect of fine-tuning is negative ($\beta_2 = -0.377$, $p < 2 \times 10^{-5}$), indicating that at fixed entropy rate and length, fine-tuned models produce less baseline semantic diversity. This is consistent with alignment compressing the generation space and enforcing more structured outputs. However, the strong positive interaction shows that the remaining uncertainty is used more efficiently. We observe a nonlinear and generally negative relationship between semantic diversity and sequence length, making length control essential in diversity analysis (see Table 3).

Table 3. Beta GLMM with dispersion modeling and interaction effects. For completeness, we direct readers to subsection C.4 for the full conditional mean and dispersion regression tables, along with additional model specifications and regression details.

Conditional model	Estimate	Std. Error	p-value
Intercept	-0.880	0.070	$< 2 \times 10^{-16}$
R	0.708	0.033	$< 2 \times 10^{-16}$
Instruct	-0.377	0.086	1.23×10^{-5}
$\mathrm{ns}(\mathrm{invN}, 4)_1$	0.152	0.041	1.99×10^{-4}
$R\times$ Instruct	1.183	0.046	$< 2 \times 10^{-16}$

6. Conclusion and Limitations

In this work, we revisit diversity in large language models through a trajectory-level information-theoretic perspective. By introducing Canopy Entropy and length–entropy coupling, we show that fine-tuning does not simply reduce uncertainty or shrink the generation space. Instead, aligned models reorganize uncertainty more efficiently across generation trajectories, producing longer generations that remain more semantically informative.

Beyond evaluation, our framework may have implications for RL-based post-training: Canopy Entropy and length–entropy coupling could be explored as regularization targets during preference optimization—for instance, to test whether constraining trajectory-level uncertainty mitigates policy collapse while preserving alignment quality.

Our work also has several limitations. Entropy-based measures characterize uncertainty allocation, but do not directly capture factuality or human preference quality. Our framework is descriptive rather than causal: although we observe systematic changes in length–entropy coupling after fine-tuning, understanding the underlying training mechanisms, including the distinct effects of supervised instruction tuning and RL-based alignment on trajectory-level uncertainty, remains an important direction for future work.

References

Agarwal, S., Zhang, Z., Yuan, L., Han, J., and Peng, H. The unreasonable effectiveness of entropy minimization in llm reasoning. *arXiv preprint arXiv:2505.15134*, 2025.

Brown, T. B., Mann, B., Ryder, N., et al. Language models are few-shot learners. *NeurIPS*, 2020.

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. D. O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, Y., Wang, X., Dehghani, M., Brahma, S., et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Figueroa-Zúñiga, J. I., Arellano-Valle, R. B., and Ferrari, S. L. Mixed beta regression: A bayesian perspective. *Computational Statistics & Data Analysis*, 61:137–147, 2013.

Folland, G. B. *Real analysis: modern techniques and their applications*. John Wiley & Sons, 1999.

Friedman, D. and Dieng, A. B. The vendi score: A diversity evaluation metric for machine learning. *arXiv preprint arXiv:2210.02410*, 2022.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Guo, Y., Shang, G., and Clavel, C. Benchmarking linguistic diversity of large language models. *Transactions of the Association for Computational Linguistics*, 13:1507–1526, 2025.

Hartig, F. Dharma: residual diagnostics for hierarchical (multi-level/mixed) regression models. *CRAN: contributed packages*, 2016.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Hu, E. J., Jain, M., Elmoznino, E., Kaddar, Y., Lajoie, G., Bengio, Y., and Malkin, N. Amortizing intractable inference in large language models. *arXiv preprint arXiv:2310.04363*, 2023.

Ismayilzada, M., Laverghetta Jr., A., Luchini, S. A., Patel, R., Bosselut, A., Plas, L. V. D., and Beaty, R. E. Creative preference optimization. In Christodoulopoulos, C., Chakraborty, T., Rose, C., and Peng, V. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 9580–9609, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.509. URL https://aclanthology.org/2025.findings-emnlp.509/.

Jelinek, F., Mercer, R. L., Bahl, L. R., and Baker, J. K. Perplexity—a measure of the difficulty of speech recognition tasks. *The journal of the Acoustical Society of America*, 62(S1):S63–S63, 1977.

Kendall, M. G. A new measure of rank correlation. *Biometrika*, 30(1-2):81–93, 1938.

Lake, T., Choi, E., and Durrett, G. From distributional to overton pluralism: Investigating large language model alignment. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6794–6814, 2025.

Li, J., Galley, M., Brockett, C., Gao, J., and Dolan, W. B. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pp. 110–119, 2016.

Lightman, H., Kosaraju, V., Burda, Y., Edwards, H., Baker, B., Lee, T., Leike, J., Schulman, J., Sutskever, I., and Cobbe, K. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Lu, W., Luu, R. K., and Buehler, M. J. Fine-tuning large language models for domain adaptation: Exploration of training strategies, scaling, model merging and synergistic capabilities. *npj Computational Materials*, 11(1):84, 2025.

Mann, H. B. and Wald, A. On stochastic limit and order relationships. *The Annals of Mathematical Statistics*, 14 (3):217–226, 1943.

Massey Jr, F. J. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46 (253):68–78, 1951.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36: 53728–53741, 2023.

Roziere, B., Gehring, J., Gloeckle, F., Sootla, S., Gat, I., Tan, X. E., Adi, Y., Liu, J., Sauvestre, R., Remez, T., et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.

Rudin, W. *Principles of Mathematical Analysis*. McGraw-Hill, New York, 3rd edition, 1976.

Shaib, C., Govindarajan, V. S., Barrow, J., Sun, J., Siu, A. F., Wallace, B. C., and Nenkova, A. Standardizing the measurement of text diversity: A tool and a comparative analysis of scores. *arXiv preprint arXiv:2403.00553*, 2024.

Shannon, C. E. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

Shypula, A., Li, S., Zhang, B., Padmakumar, V., Yin, K., and Bastani, O. Does instruction tuning reduce diversity? a case study using code generation, 2025. URL https://openreview.net/forum?id=hMEHnLJyrU.

Singhal, P., Goyal, T., Xu, J., and Durrett, G. A long way to go: Investigating length correlations in rlhf. *arXiv preprint arXiv:2310.03716*, 2023.

Spearman, C. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904.

Team, G., Kamath, A., Ferret, J., Pathak, S., Vieillard, N., Merhej, R., Perrin, S., et al. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Tevet, G. and Berant, J. Evaluating the evaluation of diversity in natural language generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 326–346, 2021.

Wang, S., Xie, Y., Zhang, W., Sun, Y., Chen, Y., Li, Y., and Zhang, Y. On the entropy dynamics in reinforcement fine-tuning of large language models. *arXiv preprint arXiv:2602.03392*, 2026.

Wang, Y., Yang, C., Huang, T., Chen, M., May, J., and Lee, M. Optimizing diversity and quality through base-aligned model collaboration. *arXiv preprint arXiv:2511.05650*, 2025.

Warner, B., Chaffin, A., Clavié, B., Weller, O., Hallström, O., Taghadouini, S., Gallagher, A., Biswas, R., Ladhak, F., Aarsen, T., Cooper, N., Adams, G., Howard, J., and Poli, I. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference, 2024. URL https://arxiv.org/abs/2412.13663.

Wei, J., Bosma, M., Zhao, V. Y., Guu, K., Yu, A. W., Lester, B., Du, N., Dai, A. M., and Le, Q. V. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

West, P. and Potts, C. Base models beat aligned models at randomness and creativity. *arXiv preprint arXiv:2505.00047*, 2025.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Yang, C., Li, S., and Holtzman, A. Llm probability concentration: How alignment shrinks the generative horizon. *arXiv preprint arXiv:2506.17871*, 2025b.

Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., and Choi, Y. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

Zhuo, T. Y., Vu, M. C., Chim, J., Hu, H., Yu, W., Widyasari, R., Yusuf, I. N. B., Zhan, H., He, J., Paul, I., et al. Bigcodebench: Benchmarking code generation with diverse function calls and complex instructions. *arXiv preprint arXiv:2406.15877*, 2024.

A. Missing Proofs and Algorithms

A.1. Equivalence to a Two-Stage Stochastic Decision Process

Theorem 3 (Equivalence between autoregressive generation and a two-stage decision process)**.** *Fix a prompt $X = x$ and a generation history $G_t = g_t = (y_1, \ldots, y_{t-1})$. Suppose the autoregressive model defines a probability distribution over the extended vocabulary $Z := \mathcal{V} \cup \{\texttt{EOS}\}$, denoted by $q(z \mid g_t, x), z \in Z$. Then this one-step generation rule is equivalent to the following two-stage stochastic decision process: $A_t \in \{\textsf{STOP}, \textsf{CONT}\}$, where $\mathbb{P}(A_t = \textsf{STOP} \mid g_t, x) = q(\texttt{EOS} \mid g_t, x)$, and $\mathbb{P}(A_t = \textsf{CONT} \mid g_t, x) = 1 - q(\texttt{EOS} \mid g_t, x)$. Conditional on $A_t = \textsf{CONT}$, the next token $Y_t \in \mathcal{V}$ is sampled from $\mathbb{P}(Y_t = v \mid g_t, x, A_t = \textsf{CONT}) = \frac{q(v \mid g_t, x)}{1 - q(\texttt{EOS} \mid g_t, x)}, v \in \mathcal{V}$, whenever $1 - q(\texttt{EOS} \mid g_t, x) > 0$. This two-stage process induces the same distribution over generated sequences as the original autoregressive model.*

Proof. Fix a prompt x and history g_t. Under the original autoregressive model, the next outcome is sampled directly from $q(z \mid g_t, x), z \in \mathcal{V} \cup \{\texttt{EOS}\}$. In particular, generation terminates at step t with probability $q(\texttt{EOS} \mid g_t, x)$. Now consider the two-stage process. The probability of termination is $\mathbb{P}(A_t = \textsf{STOP} \mid g_t, x) = q(\texttt{EOS} \mid g_t, x)$, which matches the probability assigned to EOS by the original model. Next, for any token $v \in \mathcal{V}$, the probability that the two-stage process generates v is

$$\begin{aligned}
\mathbb{P}(Y_t = v \mid g_t, x) &= \mathbb{P}(A_t = \textsf{CONT} \mid g_t, x)\, \mathbb{P}(Y_t = v \mid g_t, x, A_t = \textsf{CONT}) \\
&= (1 - q(\texttt{EOS} \mid g_t, x))\, \frac{q(v \mid g_t, x)}{1 - q(\texttt{EOS} \mid g_t, x)} \\
&= q(v \mid g_t, x).
\end{aligned}$$

Thus, the two-stage process assigns exactly the same probability to every possible next outcome:

$$\mathbb{P}(\textsf{STOP} \mid g_t, x) = q(\texttt{EOS} \mid g_t, x),\ \mathbb{P}(Y_t = v \mid g_t, x) = q(v \mid g_t, x), \quad v \in \mathcal{V}.$$

Therefore, the one-step transition distribution is identical under the two formulations. Since this equality holds for every prompt x, every generation history g_t, and every generation step t, the induced probability of any finite sequence $y_{1:N}$ terminating at length N is the same under both processes. Thus, the two-stage stochastic decision process is equivalent to the original autoregressive generation model. □

A.2. Proof of Theorem 1

Proof. Let $X \sim p_X$ denote the prompt distribution. For each realization $X = x$, consider the sequence of random variables $(Z_t)_{t \geq 1}$ with $Z_t \in \mathcal{Z} := \mathcal{V} \cup \{\texttt{EOS}\}$, where selecting EOS indicates termination. Define the stopping time $N := \min\{t : Z_t = \texttt{EOS}\}$, and let $Y_t = Z_t$ for $t < N$. Adopt the convention that $Z_t = \texttt{EOS}$ for all $t > N$, which induces a one-to-one correspondence between the infinite sequence $(Z_1, Z_2, \ldots)$ and the pair $(N, Y_{1:N})$, conditional on X. Therefore,

$$H(N, Y_{1:N} \mid X) = H(Z_1, Z_2, \ldots \mid X).$$

By the chain rule of entropy, $H(Z_1, Z_2, \ldots \mid X) = \sum_{t=1}^{\infty} H(Z_t \mid Z_{1:t-1}, X)$. Since the history G_t is completely determined by $(Z_{1:t-1}, X)$, we have

$$H(Z_t \mid Z_{1:t-1}, X) = H(Z_t \mid G_t, X) = \mathbb{E}_{G_t, X}\big[\widetilde{H}(G_t \mid X)\big].$$

Therefore, $H(Z_1, Z_2, \ldots \mid X) = \sum_{t=1}^{\infty} \mathbb{E}_{G_t, X}\big[\widetilde{H}(G_t \mid X)\big]$.

Since $\widetilde{H}(G_t \mid X) \geq 0$, we can apply Tonelli's theorem (Folland, 1999) to exchange the sum and expectation.

$$\sum_{t=1}^{\infty} \mathbb{E}_{G_t, X}\big[\widetilde{H}(G_t \mid X)\big] = \mathbb{E}_{G_t, X}\left[\sum_{t=1}^{\infty} \widetilde{H}(G_t \mid X)\right].$$

Moreover, for all $t > N$, $Z_t = \texttt{EOS}$ deterministically, so $\widetilde{H}(G_t \mid X) = 0$. Therefore, $\sum_{t=1}^{\infty} \widetilde{H}(G_t \mid X) = \sum_{t=1}^{N(X)} \widetilde{H}(G_t \mid X)$, almost surely. Combining the above, we have $H(Z_1, Z_2, \ldots \mid X) = \mathbb{E}_{G_t, X}\left[\sum_{t=1}^{N(X)} \widetilde{H}(G_t \mid X)\right]$. Finally, since $(Z_1, Z_2, \ldots)$ is in one-to-one correspondence with $(N, Y_{1:N})$ given X, we obtain $H(N, Y_{1:N} \mid X) = \mathbb{E}_{G_t, X}\left[\sum_{t=1}^{N(X)} \widetilde{H}(G_t \mid X)\right] = \mathrm{CE}^{\star}$. □

A.3. Algorithms

Algorithm 1 Estimating $\mathrm{CE}^{\star}_{\max}$ over a prompt distribution

Require: Prompt distribution p_X; number of prompts P; rollout count per prompt M; maximum length $T_{\max}$; sampling policy π (e.g. temperature/top-p/top-k).
for $p = 1$ to P **do**
  Sample prompt $X_p \sim p_X$;
  for $i = 1$ to M **do**
    Initialize history $G_1^{(p,i)} \leftarrow X_p$; set $S_{\max}^{(p,i)} \leftarrow 0$;
    for $t = 1$ to $T_{\max}$ **do**
      Compute model distribution $q_t^{(p,i)}(\cdot \mid G_t^{(p,i)}, X_p)$ under sampling policy π, including EOS as an outcome;
      Compute one-step local entropy $\tilde{H}(G_t^{(p,i)} \mid X_p) \leftarrow -\sum_{z\in\mathcal{Z}} q_t^{(p,i)}(z \mid G_t^{(p,i)}, X_p) \log q_t^{(p,i)}(z \mid G_t^{(p,i)}, X_p)$;
      Update $S_{\max}^{(p,i)} \leftarrow S_{\max}^{(p,i)} + \tilde{H}(G_t^{(p,i)} \mid X_p)$;
      Sample next outcome $z_t^{(p,i)} \sim q_t^{(p,i)}(\cdot \mid G_t^{(p,i)}, X_p)$;
      If $z_t^{(p,i)} = \texttt{EOS}$ then set $N_{\max}^{(p,i)} \leftarrow t$ and break;
      Else update history $G_{t+1}^{(p,i)} \leftarrow G_t^{(p,i)} \oplus z_t^{(p,i)}$.
    end for
    If no EOS is sampled by $T_{\max}$: set $N_{\max}^{(p,i)} \leftarrow T_{\max}$.
  end for
end for
return $\widehat{\mathrm{CE}}^{\star}_{\max,P,M} = \frac{1}{PM}\sum_{p=1}^{P}\sum_{i=1}^{M} S_{\max}^{(p,i)}$, $\widehat{\mathbb{E}}[N_{\max}] = \frac{1}{PM}\sum_{p=1}^{P}\sum_{i=1}^{M} N_{\max}^{(p,i)}$.

A.4. Examples of Prompt-Averaged Bias Decay Rates

Corollary 1. *By Remark 4.1, $g(T_{\max})$ admits the following rates.*

Uniform Exponential Tail. *If there exist constants $c, \lambda > 0$ such that, for all prompts $x \in \mathcal{X}$ and all $t \geq 1$, $\mathbb{P}(N(x) \geq t \mid X = x) \leq ce^{-\lambda t}$, then $g(T_{\max}) \leq c \log|\mathcal{Z}| \sum_{t=T_{\max}+1}^{\infty} e^{-\lambda t} = c\log|\mathcal{Z}| \frac{e^{-\lambda(T_{\max}+1)}}{1-e^{-\lambda}} = O(e^{-\lambda T_{\max}})$.*

Uniform Polynomial Tail. *If there exist constants $c > 0$ and $\alpha > 1$ such that, for all prompts $x \in \mathcal{X}$ and all $t \geq 1$, $\mathbb{P}(N(x) \geq t \mid X = x) \leq ct^{-\alpha}$, then $g(T_{\max}) \leq c\log|\mathcal{Z}| \sum_{t=T_{\max}+1}^{\infty} t^{-\alpha} \leq \frac{c\log|\mathcal{Z}|}{\alpha-1} T_{\max}^{1-\alpha} = O(T_{\max}^{1-\alpha})$.*

A.5. Proof of Theorem 2 and MSE Analysis

Proof. Given any fixed prompt $x \in \mathcal{X}$, denote $\mu_{\max}(x) := \mathbb{E}[S_{\max} \mid X = x]$, and $S_{\max} = \sum_{t=1}^{N_{\max}(x)} \widetilde{H}(G_t \mid x)$. Then $\mathrm{CE}^{\star}_{\max} = \mathbb{E}_{X\sim p_X}[\mu_{\max}(X)]$. Since $|\mathcal{Z}| < \infty$, for all g_t, we have $0 \leq \widetilde{H}(g_t \mid x) \leq H_{\max} := \log|\mathcal{Z}|$. Therefore $0 \leq S_{\max} \leq H_{\max} N_{\max}(x)$. Therefore, $\mathbb{E}[S_{\max}^2] \leq H_{\max}^2 \mathbb{E}[N_{\max}(X)^2] = H_{\max}^2 \nu_{\max}^2$, where $\nu_{\max}^2 := E[N_{\max}(X)^2]$.

Now decompose the variance of $\widehat{\mathrm{CE}}^{\star}_{\max,P,M}$. Conditional on $X_1, \ldots, X_P$, the M rollouts for each prompt are independent, by law of total variance, we have

$$\mathrm{Var}\Big(\widehat{\mathrm{CE}}^{\star}_{\max,P,M}\Big) = \mathrm{Var}\left(\frac{1}{P}\sum_{p=1}^{P}\mu_{\max}(X_p)\right) + \mathbb{E}\Big[\mathrm{Var}\Big(\widehat{\mathrm{CE}}^{\star}_{\max,P,M} \mid X_1, \ldots, X_P\Big)\Big].$$

By Jensen's inequality, $\mu_{\max}^2(X) \leq H_{\max}^2\, \mathbb{E}\big[N_{\max}(X)^2 \mid X\big]$, and taking expectations over X gives $\mathbb{E}[\mu_{\max}^2(X)] \leq H_{\max}^2 \nu_{\max}^2$. Therefore

$$\mathrm{Var}\left(\frac{1}{P}\sum_{p=1}^{P}\mu_{\max}(X_p)\right) = \frac{1}{P}\,\mathrm{Var}(\mu_{\max}(X)) \leq \frac{1}{P}\mathbb{E}[\mu_{\max}(X)^2] \leq \frac{H_{\max}^2\nu_{\max}^2}{P}.$$

From $\mathbb{E}\left[\mathrm{Var}\Big(\widehat{\mathrm{CE}}^{\star}_{\max,P,M} \mid X_1, \ldots, X_P\Big)\right] \leq \frac{H_{\max}^2\nu_{\max}^2}{PM}$, we have

$$\mathrm{Var}\left(\widehat{\mathrm{CE}}^{\star}_{\max,P,M}\right) \leq H_{\max}^2\nu_{\max}^2\left(\frac{1}{P} + \frac{1}{PM}\right).$$

Since $\mathbb{E}\left[\widehat{\mathrm{CE}}^{\star}_{\max,P,M}\right] = \mathrm{CE}^{\star}_{\max}$, Chebyshev's inequality gives that, for any $\delta \in (0,1)$, with probability at least $1-\delta$, $\left|\widehat{\mathrm{CE}}^{\star}_{\max,P,M} - \mathrm{CE}^{\star}_{\max}\right| \le H_{\max}\nu_{\max}\sqrt{\frac{1}{\delta}\left(\frac{1}{P}+\frac{1}{PM}\right)}$. By the truncation bias assumption, $0 \le \mathrm{CE}^{\star} - \mathrm{CE}^{\star}_{\max} \le g(T_{\max})$. Thus,

$$\left|\widehat{\mathrm{CE}}^{\star}_{\max,P,M} - \mathrm{CE}^{\star}\right| \le H_{\max}\nu_{\max}\sqrt{\frac{1}{\delta}\left(\frac{1}{P}+\frac{1}{PM}\right)} + g(T_{\max}).$$

Equivalently, $\left|\widehat{\mathrm{CE}}^{\star}_{\max,P,M} - \mathrm{CE}^{\star}\right| = O_{\mathbb{P}}\left(\nu_{\max}\sqrt{\frac{1}{P}+\frac{1}{PM}} + g(T_{\max})\right)$. Thus, if $T_{\max} = T_{\max}(P,M) \to \infty$, $g(T_{\max}) \to 0$, and if $\nu_{\max}(P,M)\sqrt{\frac{1}{P}+\frac{1}{PM}} \to 0$, then we have $\widehat{\mathrm{CE}}^{\star}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathrm{CE}^{\star}$. □

Theorem 4 (MSE of the prompt-averaged truncated CE estimator)**.**

$$\mathbb{E}\left[\left(\widehat{\mathrm{CE}}^{\star}_{\max,P,M} - \mathrm{CE}^{\star}\right)^2\right] \le \frac{H^2_{\max}}{P}\operatorname{Var}_{X\sim p_X}(\mathbb{E}[N_{\max} \mid X]) + \frac{H^2_{\max}}{PM}\mathbb{E}[N^2_{\max}] + g^2(T_{\max}).$$

In Theorem 4, we further characterize the mean squared error (MSE) of $\widehat{\mathrm{CE}}^{\star}_{\max,P,M}$ as the sum of the squared truncation bias and the variance of the estimator. The variance decreases with both the number of prompts P and the total number of rollouts PM, highlighting two distinct sources of statistical efficiency: diversity across prompts and repeated sampling within each prompt.

The variance admits a hierarchical decomposition with two distinct sources of randomness: a $1/P$ term due to variability across prompts, and a $1/PM$ term arising from rollout-level stochasticity within each prompt. Importantly, the prompt-level variance cannot be reduced by increasing the number of rollouts per prompt, reflecting the intrinsic hierarchical structure of the estimator. Moreover, the Monte Carlo error is scaled by the effective trajectory length (captured by $T_{\max}$ or, more precisely, $N_{\max}$), since longer generations accumulate more uncertainty along the trajectory.

These results reveal a fundamental trade-off. Increasing $T_{\max}$ reduces the truncation bias by capturing more of the generation process, but also increases computational cost and variance through longer trajectories.

Proof. Define the per-prompt rollout average $\bar{S}^{(p)}_{\max} := \frac{1}{M}\sum_{i=1}^{M} S^{(p,i)}_{\max}$. Then $\widehat{\mathrm{CE}}^{\star}_{\max,P,M} = \frac{1}{P}\sum_{p=1}^{P}\bar{S}^{(p)}_{\max}$. Conditional on X_p, $\mathbb{E}[\bar{S}^{(p)}_{\max} \mid X_p] = \mu_{\max}(X_p)$, and, since the M rollouts are conditionally independent, $\operatorname{Var}(\bar{S}^{(p)}_{\max} \mid X_p) = \frac{1}{M}\sigma^2_{\max}(X_p)$. By the law of total variance, we have

$$\operatorname{Var}(\bar{S}^{(p)}_{\max}) = \operatorname{Var}_{X_p}\left(\mathbb{E}[\bar{S}^{(p)}_{\max} \mid X_p]\right) + \mathbb{E}_{X_p}\left[\operatorname{Var}(\bar{S}^{(p)}_{\max} \mid X_p)\right].$$

Substituting the conditional mean and variance gives $\operatorname{Var}(\bar{S}^{(p)}_{\max}) = \operatorname{Var}_{X\sim p_X}(\mu_{\max}(X)) + \frac{1}{M}\mathbb{E}_{X\sim p_X}[\sigma^2_{\max}(X)]$. Since the prompts $X_1,\dots,X_P$ are i.i.d., the variables $\bar{S}^{(1)}_{\max},\dots,\bar{S}^{(P)}_{\max}$ are i.i.d. Therefore, $\operatorname{Var}\left(\widehat{\mathrm{CE}}^{\star}_{\max,P,M}\right) = \frac{1}{P}\operatorname{Var}(\bar{S}^{(p)}_{\max}) = \frac{1}{P}\operatorname{Var}_{X\sim p_X}(\mu_{\max}(X)) + \frac{1}{PM}\mathbb{E}_{X\sim p_X}[\sigma^2_{\max}(X)]$.

For the MSE, write $\widehat{\mathrm{CE}}^{\star}_{\max,P,M} - \mathrm{CE}^{\star} = \left(\widehat{\mathrm{CE}}^{\star}_{\max,P,M} - \mathrm{CE}^{\star}_{\max}\right) - g(T_{\max})$. Taking squares and expectations yields $\mathbb{E}\left[\left(\widehat{\mathrm{CE}}^{\star}_{\max,P,M} - \mathrm{CE}^{\star}\right)^2\right] = \mathbb{E}\left[\left(\widehat{\mathrm{CE}}^{\star}_{\max,P,M} - \mathrm{CE}^{\star}_{\max}\right)^2\right] + (g(T_{\max}))^2$, because $\mathbb{E}[\widehat{\mathrm{CE}}^{\star}_{\max,P,M} - \mathrm{CE}^{\star}_{\max}] = 0$.

Since pathwise, for any given history g_t and prompt x, we have $0 \le S_{\max}(x) = \sum_{t=1}^{N_{\max}(x)}\widetilde{H}(g_t \mid x) \le H_{\max}N_{\max}(x)$. Thus, conditional on $X = x$, $\operatorname{Var}(S_{\max}(x) \mid X = x) \le \mathbb{E}[S^2_{\max}(x) \mid X = x] \le H^2_{\max}\mathbb{E}[N^2_{\max}(x) \mid X = x]$. Averaging over prompts gives $\mathbb{E}_X[\sigma^2_{\max}(X)] \le H^2_{\max}\mathbb{E}_X\mathbb{E}[N^2_{\max}(X) \mid X] = H^2_{\max}\mathbb{E}[N^2_{\max}]$.

Similarly, $\mu_{\max}(X) = \mathbb{E}[S_{\max}(X) \mid X] \le H_{\max}\mathbb{E}[N_{\max} \mid X]$. Since both quantities are nonnegative, this gives the prompt-level bound $0 \le \mu_{\max}(X) \le H_{\max}\mathbb{E}[N_{\max} \mid X]$. Therefore, $\operatorname{Var}_X(\mu_{\max}(X)) \le H^2_{\max}\operatorname{Var}_X(\mathbb{E}[N_{\max} \mid X])$. Combining these two refined bounds with the exact variance decomposition gives

$$\operatorname{Var}\left(\widehat{\mathrm{CE}}^{\star}_{\max,P,M}\right) \le \frac{H^2_{\max}}{P}\operatorname{Var}_{X\sim p_X}(\mathbb{E}[N_{\max} \mid X]) + \frac{H^2_{\max}}{PM}\mathbb{E}[N^2_{\max}].$$

Adding the squared truncation bias $g^2(T_{\max})$ gives the stated refined MSE upper bound. □

A.6. Estimation and Consistency of GenPPL, BF, and Length–Entropy Correlation

Lemma 1 (Consistency of the truncated length estimator)**.** *Let* $\widehat{N}_{\max,P,M} := \frac{1}{PM}\sum_{p=1}^{P}\sum_{i=1}^{M} N_{\max}^{(p,i)}$*, where* $N_{\max}^{(p,i)} := \min\{N^{(p,i)}, T_{\max}\}$*. Assume* $\mathbb{E}[N] < \infty$*. If* $T_{\max} = T_{\max}(P, M) \to \infty$ *and* $\frac{\sqrt{\mathbb{E}[N_{\max}^2]}}{\sqrt{P}} \to 0$*, then* $\widehat{N}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathbb{E}[N]$*.*

Proof. Let $\eta_{\max}(x) := \mathbb{E}[N_{\max} \mid X = x]$. By the law of total variance, $\mathrm{Var}(\widehat{N}_{\max,P,M}) = \frac{1}{P}\mathrm{Var}_{X\sim p_X}(\eta_{\max}(X)) + \frac{1}{PM}\mathbb{E}_{X\sim p_X}\left[\mathrm{Var}(N_{\max} \mid X)\right]$. Using $\mathrm{Var}(\eta_{\max}(X)) \le \mathbb{E}[\eta_{\max}(X)^2] \le \mathbb{E}[N_{\max}^2]$, and $\mathbb{E}[\mathrm{Var}(N_{\max} \mid X)] \le \mathbb{E}[N_{\max}^2]$, we obtain $\mathrm{Var}(\widehat{N}_{\max,P,M}) \le \mathbb{E}[N_{\max}^2]\left(\frac{1}{P} + \frac{1}{PM}\right)$. Therefore, under the stated condition, $\mathrm{Var}(\widehat{N}_{\max,P,M}) \to 0$.

In addition, $\mathbb{E}[\widehat{N}_{\max,P,M}] = \mathbb{E}[N_{\max}]$. Since $N_{\max} = \min\{N, T_{\max}\} \uparrow N$ as $T_{\max} \to \infty$, the monotone convergence theorem (Rudin, 1976) gives $\mathbb{E}[N_{\max}] \to \mathbb{E}[N]$. Hence,

$$\widehat{N}_{\max,P,M} - \mathbb{E}[N] = \left(\widehat{N}_{\max,P,M} - \mathbb{E}[N_{\max}]\right) + \left(\mathbb{E}[N_{\max}] - \mathbb{E}[N]\right).$$

The first term converges to 0 in probability by Chebyshev's inequality, and the second term converges to 0 deterministically. Therefore, $\widehat{N}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathbb{E}[N]$. □

Corollary 2 (Consistency of GenPPL)**.** *Let* $\widehat{\mathrm{CE}}^{\star}_{\max,P,M} = \frac{1}{PM}\sum_{p=1}^{P}\sum_{i=1}^{M} S_{\max}^{(p,i)}$ *be the Monte Carlo estimator of* $\mathrm{CE}^{\star}$*, and let* $\widehat{N}_{\max,P,M} := \frac{1}{PM}\sum_{p=1}^{P}\sum_{i=1}^{M} N_{\max}^{(p,i)}$ *be the corresponding estimator of* $\mathbb{E}[N]$*. Recall that* $\widehat{\mathrm{CE}}^{\star}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathrm{CE}^{\star}$ *(by Theorem 2) and* $\widehat{N}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathbb{E}[N]$ *(by Lemma 1). Define* $\widehat{\mathrm{GenPPL}}_{\max,P,M} := \exp\left(\frac{\widehat{\mathrm{CE}}^{\star}_{\max,P,M}}{\widehat{N}_{\max,P,M}}\right)$, $\mathrm{GenPPL} := \exp\left(\frac{\mathrm{CE}^{\star}}{\mathbb{E}[N]}\right)$*. Then* $\widehat{\mathrm{GenPPL}}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathrm{GenPPL}$*.*

Proof. Since we have $\widehat{\mathrm{CE}}^{\star}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathrm{CE}^{\star}$, $\widehat{N}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathbb{E}[N]$, and $\mathbb{E}[N] > 0$, the function $f(a, b) = \exp(a/b)$ is continuous at $(\mathrm{CE}^{\star}, \mathbb{E}[N])$. Therefore, by the continuous mapping theorem (Mann & Wald, 1943), $\exp\left(\frac{\widehat{\mathrm{CE}}^{\star}_{\max,P,M}}{\widehat{N}_{\max,P,M}}\right) \xrightarrow{\mathbb{P}} \exp\left(\frac{\mathrm{CE}^{\star}}{\mathbb{E}[N]}\right)$. Hence, $\widehat{\mathrm{GenPPL}}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathrm{GenPPL}$, which completes the proof. □

Corollary 3 (Consistency of the BF estimator)**.** *Let* $\widehat{L}_{\max,P,M} := \frac{1}{PM}\sum_{p=1}^{P}\sum_{i=1}^{M}\frac{S_{\max}^{(p,i)}}{N_{\max}^{(p,i)}}$, $\widehat{\mathrm{BF}}_{\max,P,M} := \exp\left(\widehat{L}_{\max,P,M}\right)$*, where* $S_{\max}^{(p,i)} = \sum_{t=1}^{N_{\max}^{(p,i)}} \widetilde{H}(G_t^{(p,i)} \mid x_p)$, $N_{\max}^{(p,i)} = \min\{N^{(p,i)}, T_{\max}\}$*. Assume that* $N \ge 1$ *almost surely and that the normalized truncated entropy variables* $r_{\max} := \frac{S_{\max}}{N_{\max}}$ *are integrable. Then, for fixed* $T_{\max}$*, as* $P, M \to \infty$*,* $\widehat{L}_{\max,P,M} \xrightarrow{\mathbb{P}} L_{\max} := \mathbb{E}\left[\frac{S_{\max}}{N_{\max}}\right]$*, and therefore, by the continuous mapping theorem,* $\widehat{\mathrm{BF}}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathrm{BF}_{\max} := \exp(L_{\max})$*. Moreover, if* $\frac{S_{\max}}{N_{\max}} \xrightarrow[T_{\max}\to\infty]{a.s.} \frac{S}{N}$ *and the sequence* $\{S_{\max}/N_{\max}\}_{T_{\max}\ge 1}$ *is uniformly integrable, then* $L_{\max} \to L := \mathbb{E}\left[\frac{1}{N}\sum_{t=1}^{N}\widetilde{H}(G_t \mid X)\right]$*, and consequently* $\mathrm{BF}_{\max} \to \mathrm{BF} := \exp(L)$*. Thus, under these conditions,* $\widehat{\mathrm{BF}}_{\max,P,M} \xrightarrow{\mathbb{P}} \mathrm{BF}$ *as* $P, M \to \infty$ *and* $T_{\max} \to \infty$*.*

Proof. For fixed $T_{\max}$, the variables $\hat{r}_{\max}^{(p,i)} = \frac{S_{\max}^{(p,i)}}{N_{\max}^{(p,i)}}$ are integrable rollout-level observations. Since $N \ge 1$ almost surely, the denominator is bounded away from zero. Applying the law of large numbers to the empirical average over prompts and rollouts gives $\widehat{L}_{\max,P,M} = \frac{1}{PM}\sum_{p=1}^{P}\sum_{i=1}^{M}\hat{r}_{\max}^{(p,i)} \xrightarrow{\mathbb{P}} \mathbb{E}[r_{\max}] = L_{\max}$. Since the exponential map is continuous, $\widehat{\mathrm{BF}}_{\max,P,M} = \exp(\widehat{L}_{\max,P,M}) \xrightarrow{\mathbb{P}} \exp(L_{\max}) = \mathrm{BF}_{\max}$. Finally, if $S_{\max}/N_{\max} \to S/N$ almost surely and the normalized entropy ratios are uniformly integrable, then convergence of expectations follows, yielding $L_{\max} \to L$. Another application of continuity of the exponential function gives $\mathrm{BF}_{\max} \to \mathrm{BF}$. □

Empirical Support for the Truncation Condition. The consistency result above requires the truncation bias induced by $T_{\max}$ to vanish, or at least to be negligible in finite-sample estimation. While this is a population-level condition and cannot be proven from data alone, our diagnostics provide empirical support for it in our experimental setting. With $T_{\max} = 4000$, the observed truncation rates are uniformly small across tasks, model families, and variants: most are below 1%, and the

largest observed rate is 4.32% (see subsection C.1). In addition, the empirical length distributions concentrate well below the truncation threshold (see Figure 3), and the number of active rollouts decays rapidly with token position (see Figure 2). Together with the bounded observed entropy-rate trajectories (see Figure 2), these results suggest that the normalized entropy ratios are well behaved and that the finite-$T_{\max}$ truncation bias is small in practice.

Estimating Entropy Rate. For each rollout (p, i), Algorithm 1 returns the truncated stopping time $N_{\max}^{(p,i)} = \min\{N^{(p,i)}, T_{\max}\}$ and the accumulated truncated canopy entropy $S_{\max}^{(p,i)} = \sum_{t=1}^{N_{\max}^{(p,i)}} \widetilde{H}(G_t^{(p,i)} \mid x_p)$. We estimate the rollout-level entropy rate by $\widehat{r}_{\max}^{(p,i)} := \frac{S_{\max}^{(p,i)}}{N_{\max}^{(p,i)}}$. Thus, for each prompt x_p, the prompt-specific length–entropy relationship is estimated from the rollout-level pairs $\left\{\left(N_{\max}^{(p,i)}, \widehat{r}_{\max}^{(p,i)}\right)\right\}_{i=1}^{M}$.

Length–Entropy Correlation. Definition 4 introduced the length–entropy correlation $\rho(N, r_N)$ at a conceptual level. We now provide its statistically rigorous formulation together with the corresponding aggregation procedure.

Definition 5 (Length–Entropy Correlation)**.** *For a fixed prompt $X = x$, define the prompt-conditional Pearson correlation $\rho_X := \rho(N, r_N \mid X = x) = \frac{\mathrm{Cov}(N, r_N \mid X=x)}{\sqrt{\mathrm{Var}(N \mid X=x)\,\mathrm{Var}(r_N \mid X=x)}}$. Since the prompt-conditional correlation ρ_X is bounded and nonlinear, directly averaging correlations across prompts may introduce bias. Therefore, we aggregate the prompt-conditional Pearson correlations on the Fisher-z[5] scale to obtain a prompt-averaged correlation measure. Then the (prompt-averaged) length–entropy correlation $\rho(N, r_N) := \tanh\left(\frac{\mathbb{E}_{X\sim p_X}[w_X \,\mathrm{atanh}(\rho_X)]}{\mathbb{E}_{X\sim p_X}[w_X]}\right)$, where $w_X \geq 0$ is a prompt-specific reliability weight.*

Rank-Based Length–Entropy Correlations. In addition to Pearson correlation, which measures linear association, we also report two rank-based measures. Spearman correlation (Spearman, 1904) is Pearson correlation applied to the ranked variables and therefore captures monotone dependence. Kendall's τ_b (Kendall, 1938) measures ordinal association by comparing concordant and discordant rollout pairs, with tie adjustment. Since Kendall's τ_b is defined through pairwise comparisons, we aggregate it using pair-count weights rather than Fisher-z averaging. We estimate and report rank-based length–entropy correlations to assess the robustness of our findings. The consistently positive relationship across different correlation measures suggests that the observed trend is not an artifact of any particular notion of correlation.

Proposition 1 (Consistency of prompt-averaged length–entropy correlation estimators)**.** *For each prompt $X = x$, define $r_{\max}(x) := \frac{S_{\max}(x)}{N_{\max}(x)},\ S_{\max}(x) = \sum_{t=1}^{N_{\max}(x)} \widetilde{H}(G_t \mid x),\ N_{\max}(x) = \min\{N(x), T_{\max}\}$. Note that in this case, $S_{\max}(x)$ and $N_{\max}(x)$ are random variables which depend on the generated history G_t. Let $\rho_{\max}^{\mathrm{P}}(x) := \mathrm{Corr}(N_{\max}(x), r_{\max}(x) \mid X = x)$ denote the Pearson correlation, let $\rho_{\max}^{\mathrm{S}}(x) := \mathrm{Corr}\left(F_{N_{\max}|x}(N_{\max}), F_{r_{\max}|x}(r_{\max}) \mid X = x\right)$ denote the population Spearman rank correlation, and let $\tau_{\max}^{\mathrm{K}}(x)$ denote Kendall's τ_b between $N_{\max}$ and $r_{\max}$ conditional on $X = x$. For sampled prompts $x_1, \ldots, x_P$, suppose that for each prompt x_p we observe n_p conditionally i.i.d. rollouts and compute the prompt-specific estimates $\widehat{\rho}_p^{\mathrm{P}},\ \widehat{\rho}_p^{\mathrm{S}},\ \widehat{\tau}_p^{\mathrm{K}},\ \widehat{\mathrm{Cov}}_p$ as in Algorithm 2. Assume that $N_{\max} \geq 1$ almost surely. For Pearson and covariance, assume that the conditional fourth moments of $N_{\max}$ and $r_{\max}$ are uniformly bounded and that $0 < c \leq \mathrm{Var}(N_{\max} \mid X = x),\ 0 < c \leq \mathrm{Var}(r_{\max} \mid X = x)$ uniformly over x. For Spearman and Kendall, assume the corresponding rank-correlation functionals are continuous at the conditional law of $(N_{\max}, r_{\max}) \mid X = x$, uniformly over x; in particular, ties are either absent or handled by the mid-rank and τ_b conventions used in Algorithm 2. Finally, assume $|\rho_{\max}^{\mathrm{P}}(x)| \leq 1 - \epsilon_0,\ |\rho_{\max}^{\mathrm{S}}(x)| \leq 1 - \epsilon_0$ for some $\epsilon_0 > 0$, and that $\min_p n_p \to \infty$ as $P \to \infty$. Then, as $P \to \infty$ and $\min_p n_p \to \infty$, $\widehat{\rho}_{\mathrm{pc,max}}^{\mathrm{P}} \xrightarrow{\mathbb{P}} \rho_{\mathrm{pc,max}}^{\mathrm{P}},\ \widehat{\rho}_{\mathrm{pc,max}}^{\mathrm{S}} \xrightarrow{\mathbb{P}} \rho_{\mathrm{pc,max}}^{\mathrm{S}}$, where $\rho_{\mathrm{pc,max}}^{\mathrm{P}} = \tanh\left(\frac{\mathbb{E}_X[w^{\mathrm{F}}(X)\,\mathrm{atanh}\{\rho_{\max}^{\mathrm{P}}(X)\}]}{\mathbb{E}_X[w^{\mathrm{F}}(X)]}\right)$, $\rho_{\mathrm{pc,max}}^{\mathrm{S}} = \tanh\left(\frac{\mathbb{E}_X[w^{\mathrm{F}}(X)\,\mathrm{atanh}\{\rho_{\max}^{\mathrm{S}}(X)\}]}{\mathbb{E}_X[w^{\mathrm{F}}(X)]}\right)$. Also, $\widehat{\tau}_{\mathrm{pc,max}}^{\mathrm{K}} \xrightarrow{\mathbb{P}} \tau_{\mathrm{pc,max}}^{\mathrm{K}} := \frac{\mathbb{E}_X[w^{\mathrm{K}}(X)\tau_{\max}^{\mathrm{K}}(X)]}{\mathbb{E}_X[w^{\mathrm{K}}(X)]}$, and $\widehat{\mathrm{Cov}}_{\mathrm{pc,max}} \xrightarrow{\mathbb{P}} \mathrm{Cov}_{\mathrm{pc,max}} := \frac{\mathbb{E}_X[w^{\mathrm{Cov}}(X)\,\mathrm{Cov}(N_{\max}, r_{\max} \mid X)]}{\mathbb{E}_X[w^{\mathrm{Cov}}(X)]}$. If additionally $T_{\max} \to \infty$ and the corresponding truncated targets converge to their untruncated counterparts, then the same estimators are consistent for the corresponding prompt-averaged untruncated Pearson, Spearman, Kendall, and covariance targets.*

Proof. We prove the result in two steps: first within each prompt, and then across prompts. For a fixed prompt x_p, the rollout-

[5]This transformation maps correlation coefficients from $[-1, 1]$ to $\mathbb{R}$ and stabilizes their variance, making them approximately normally distributed with variance $1/(n_p - 3)$. n_p is the number of rollouts associated with each prompt p. Directly averaging correlations would lead to biased estimates due to the nonlinear and bounded nature of correlation coefficients. See Algorithm 2 for details.

Algorithm 2 Estimating prompt-controlled length–entropy correlations

Require: Prompt distribution p_X; number of prompts P; rollout counts $\{n_p\}_{p=1}^{P}$; maximum length $T_{\max}$; sampling policy π; clipping constant $\epsilon > 0$.
for $p = 1$ to P **do**
  Sample prompt $X_p \sim p_X$;
  for $i = 1$ to n_p **do**
    Generate one rollout using Algorithm 1;
    Record $N_{\max}^{(p,i)} = \min\{N^{(p,i)}, T_{\max}\}$, $S_{\max}^{(p,i)} = \sum_{t=1}^{N_{\max}^{(p,i)}} \widetilde{H}(g_t^{(p,i)} \mid X_p)$.
    Compute the rollout-level entropy rate $r_{\max}^{(p,i)} = \frac{S_{\max}^{(p,i)}}{N_{\max}^{(p,i)}}$.
  end for
  Compute the prompt-specific Pearson correlation $\widehat{\rho}_p^{\mathrm{P}} = \mathrm{Corr}\left(\{N_{\max}^{(p,i)}\}_{i=1}^{n_p}, \{r_{\max}^{(p,i)}\}_{i=1}^{n_p}\right)$.
  Compute the prompt-specific Spearman correlation
    $\widehat{\rho}_p^{\mathrm{S}} = \mathrm{Corr}\left(\{\mathrm{rank}(N_{\max}^{(p,i)})\}_{i=1}^{n_p}, \{\mathrm{rank}(r_{\max}^{(p,i)})\}_{i=1}^{n_p}\right)$, using mid-ranks for ties.
  Compute the prompt-specific Kendall's τ_b $\widehat{\tau}_p^{\mathrm{K}} = \mathrm{KendallTauB}\left(\{N_{\max}^{(p,i)}\}_{i=1}^{n_p}, \{r_{\max}^{(p,i)}\}_{i=1}^{n_p}\right)$.
  Compute the prompt-specific covariance $\widehat{\mathrm{Cov}}_p = \widehat{\mathrm{Cov}}\left(\{N_{\max}^{(p,i)}\}_{i=1}^{n_p}, \{r_{\max}^{(p,i)}\}_{i=1}^{n_p}\right)$.
end for
Aggregate Pearson and Spearman using Fisher-z weighted averages: $w_p^{\mathrm{F}} = \max\{n_p - 3, 0\}$,
  $\widehat{\rho}_{\mathrm{pc}}^{\mathrm{P}} = \tanh\left(\frac{\sum_{p=1}^{P} w_p^{\mathrm{F}} \operatorname{atanh}\left(\mathrm{clip}(\widehat{\rho}_p^{\mathrm{P}}, -1+\epsilon, 1-\epsilon)\right)}{\sum_{p=1}^{P} w_p^{\mathrm{F}}}\right)$, and
  $\widehat{\rho}_{\mathrm{pc}}^{\mathrm{S}} = \tanh\left(\frac{\sum_{p=1}^{P} w_p^{\mathrm{F}} \operatorname{atanh}\left(\mathrm{clip}(\widehat{\rho}_p^{\mathrm{S}}, -1+\epsilon, 1-\epsilon)\right)}{\sum_{p=1}^{P} w_p^{\mathrm{F}}}\right)$.
Aggregate Kendall's τ_b using pair-count weights:$w_p^{\mathrm{K}} = \max\left\{\frac{n_p(n_p-1)}{2}, 0\right\}$,$\widehat{\tau}_{\mathrm{pc}}^{\mathrm{K}} = \frac{\sum_{p=1}^{P} w_p^{\mathrm{K}} \widehat{\tau}_p^{\mathrm{K}}}{\sum_{p=1}^{P} w_p^{\mathrm{K}}}$.
Aggregate covariance using degrees-of-freedom weights: $w_p^{\mathrm{Cov}} = \max\{n_p - 1, 0\}$, and $\widehat{\mathrm{Cov}}_{\mathrm{pc}} = \frac{\sum_{p=1}^{P} w_p^{\mathrm{Cov}} \widehat{\mathrm{Cov}}_p}{\sum_{p=1}^{P} w_p^{\mathrm{Cov}}}$.
return $\widehat{\rho}_{\mathrm{pc}}^{\mathrm{P}}$, $\widehat{\rho}_{\mathrm{pc}}^{\mathrm{S}}$, $\widehat{\tau}_{\mathrm{pc}}^{\mathrm{K}}$, $\widehat{\mathrm{Cov}}_{\mathrm{pc}}$.

level pairs $\left(N_{\max}^{(p,i)}, r_{\max}^{(p,i)}\right)$, $i = 1, \ldots, n_p$, are conditionally i.i.d. By Algorithm 2, both coordinates are observed from the same rollout, with $r_{\max}^{(p,i)} = \frac{S_{\max}^{(p,i)}}{N_{\max}^{(p,i)}}$. The moment and nondegeneracy assumptions imply consistency of the sample covariance and variances. Hence, $\widehat{\rho}_p^{\mathrm{P}} \xrightarrow{\mathbb{P}} \rho_{\max}^{\mathrm{P}}(x_p)$, $\widehat{\mathrm{Cov}}_p \xrightarrow{\mathbb{P}} \mathrm{Cov}(N_{\max}, r_{\max} \mid x_p)$. Similarly, by consistency of empirical ranks and the assumed continuity of the rank-correlation functionals, $\widehat{\rho}_p^{\mathrm{S}} \xrightarrow{\mathbb{P}} \rho_{\max}^{\mathrm{S}}(x_p)$, $\widehat{\tau}_p^{\mathrm{K}} \xrightarrow{\mathbb{P}} \tau_{\max}^{\mathrm{K}}(x_p)$. Since $|\rho_{\max}^{\mathrm{P}}(x)|$ and $|\rho_{\max}^{\mathrm{S}}(x)|$ are bounded away from one, and the clipping constant is chosen smaller than this margin, the Fisher transform is continuous in a neighborhood of the limiting values. Therefore, $\operatorname{atanh}\left(\mathrm{clip}(\widehat{\rho}_p^{\mathrm{P}}, -1+\epsilon, 1-\epsilon)\right) \xrightarrow{\mathbb{P}} \operatorname{atanh}\{\rho_{\max}^{\mathrm{P}}(x_p)\}$, and $\operatorname{atanh}\left(\mathrm{clip}(\widehat{\rho}_p^{\mathrm{S}}, -1+\epsilon, 1-\epsilon)\right) \xrightarrow{\mathbb{P}} \operatorname{atanh}\{\rho_{\max}^{\mathrm{S}}(x_p)\}$. Now consider the aggregation across prompts. For Pearson, define $Z_p^{\mathrm{P}} = \operatorname{atanh}\{\rho_{\max}^{\mathrm{P}}(x_p)\}$. By the previous step and the uniform assumptions, $\frac{\sum_{p=1}^{P} w_p^{\mathrm{F}}\left[\operatorname{atanh}\left(\mathrm{clip}(\widehat{\rho}_p^{\mathrm{P}}, -1+\epsilon, 1-\epsilon)\right) - Z_p^{\mathrm{P}}\right]}{\sum_{p=1}^{P} w_p^{\mathrm{F}}} \xrightarrow{\mathbb{P}} 0$. Since prompts are sampled i.i.d., the weighted law of large numbers gives $\frac{\sum_{p=1}^{P} w_p^{\mathrm{F}} Z_p^{\mathrm{P}}}{\sum_{p=1}^{P} w_p^{\mathrm{F}}} \xrightarrow{\mathbb{P}} \frac{\mathbb{E}_X[w^{\mathrm{F}}(X) \operatorname{atanh}\{\rho_{\max}^{\mathrm{P}}(X)\}]}{\mathbb{E}_X[w^{\mathrm{F}}(X)]}$. Applying the continuous mapping theorem with $\tanh(\cdot)$ yields $\widehat{\rho}_{\mathrm{pc,max}}^{\mathrm{P}} \xrightarrow{\mathbb{P}} \rho_{\mathrm{pc,max}}^{\mathrm{P}}$. The Spearman result is identical after replacing $\rho_{\max}^{\mathrm{P}}$ by $\rho_{\max}^{\mathrm{S}}$. For Kendall's τ_b, Algorithm 2 aggregates directly on the τ scale using pair-count weights $w_p^{\mathrm{K}} = \frac{n_p(n_p-1)}{2}$. Thus, by the prompt-level consistency of $\widehat{\tau}_p^{\mathrm{K}}$ and the weighted law of large numbers, $\widehat{\tau}_{\mathrm{pc,max}}^{\mathrm{K}} = \frac{\sum_{p=1}^{P} w_p^{\mathrm{K}} \widehat{\tau}_p^{\mathrm{K}}}{\sum_{p=1}^{P} w_p^{\mathrm{K}}} \xrightarrow{\mathbb{P}} \frac{\mathbb{E}_X[w^{\mathrm{K}}(X) \tau_{\max}^{\mathrm{K}}(X)]}{\mathbb{E}_X[w^{\mathrm{K}}(X)]}$. The covariance result follows in the same way, using weights $w_p^{\mathrm{Cov}} = (n_p - 1)_+$ and the prompt-level consistency of $\widehat{\mathrm{Cov}}_p$. Finally, if $T_{\max} \to \infty$ and the truncated conditional covariance, variance, and rank-association functionals converge to their untruncated counterparts, then the truncated prompt-averaged targets converge to the corresponding untruncated targets. Combining this approximation step with the fixed-$T_{\max}$ consistency above completes the proof. □

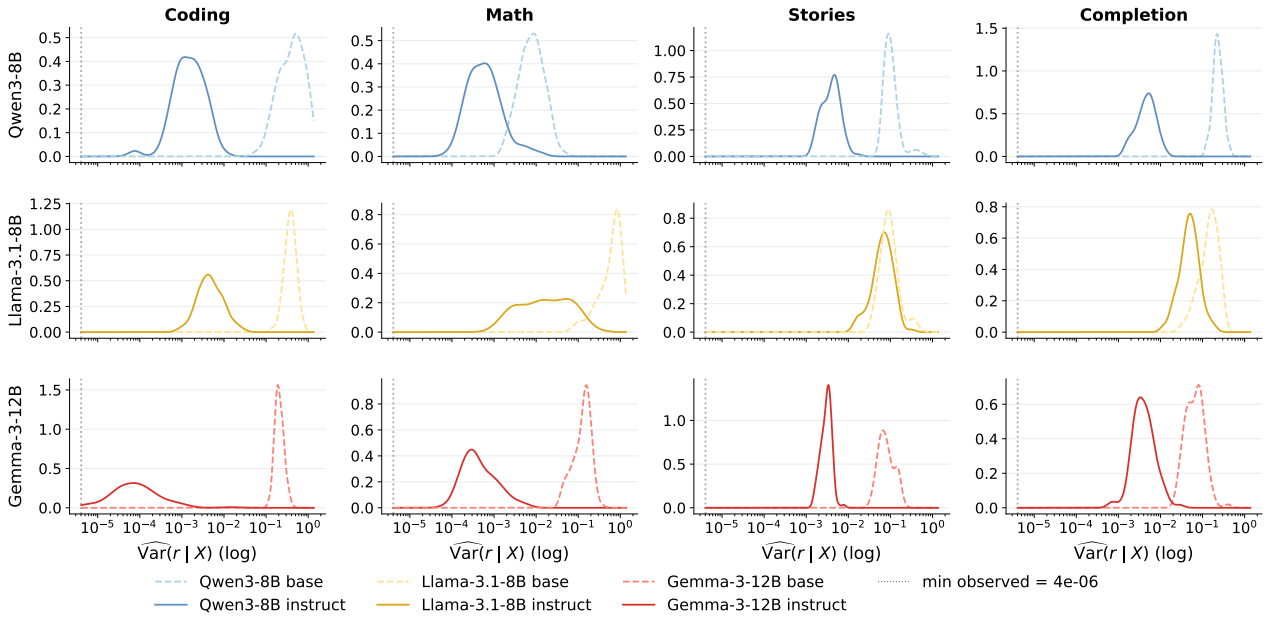

Figure 4. Gaussian KDEs of $\log \widehat{\mathrm{Var}}(r \mid x_p)$ over $P{=}100$ prompts. The dotted vertical marks $\min_p \widehat{\mathrm{Var}}(r \mid x_p) \approx 4{\times}10^{-6}$. All densities sit well to the right of zero, providing empirical support for the bounded-away-from-zero assumption.

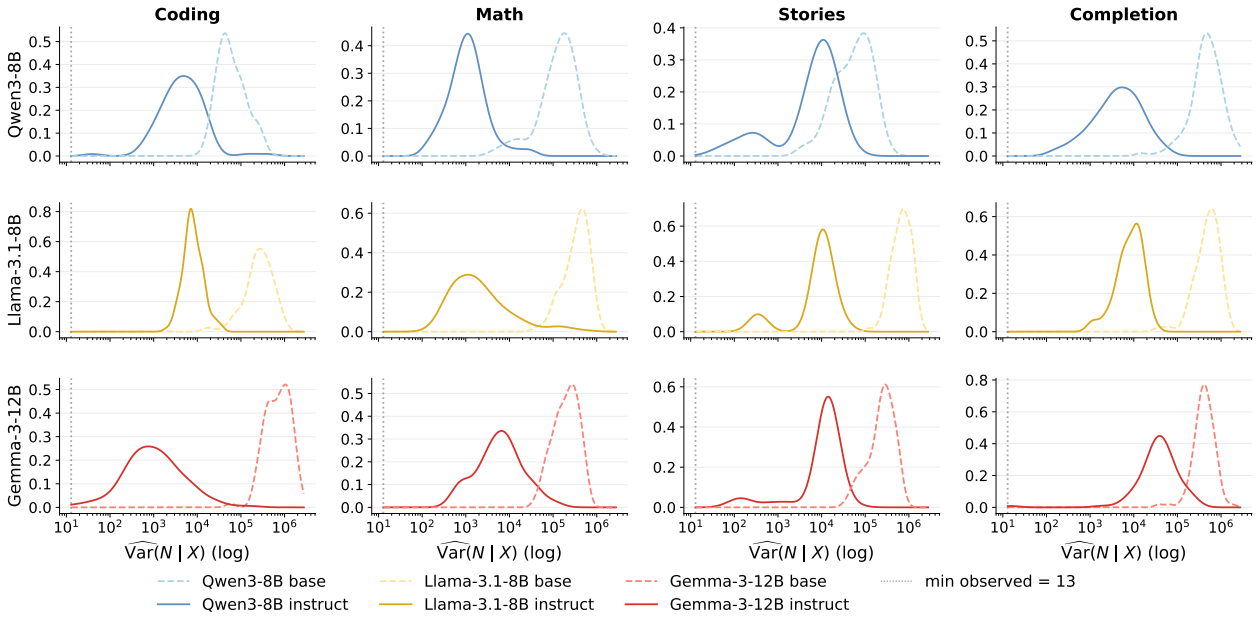

Figure 5. Gaussian KDEs of $\log \widehat{\mathrm{Var}}(N \mid x_p)$ over $P{=}100$ prompts. The dotted vertical line marks $\min_p \widehat{\mathrm{Var}}(N \mid x_p) \approx 13$, the smallest per-prompt variance observed across all model-dataset combinations. All densities concentrate at $\widehat{\mathrm{Var}}(N \mid x_p) \gg 0$, empirically supporting the bounded-away-from-zero assumption. Instruct variants consistently shift to smaller values, indicating that fine-tuning produces tighter length distributions.

Empirical Support for the Consistency Assumptions. The assumptions underlying Proposition 1 are supported by several empirical diagnostics observed in our experiments. First, the truncation rates under $T_{\max} = 4000$ are uniformly small across model families, tasks, and variants, with most rates below 1% and the maximum observed rate equal to 4.32% (see Table 5). This suggests that the truncated quantities $(N_{\max}, r_{\max})$ closely approximate their untruncated counterparts (N, r_N) in practice. Second, the empirical sequence-length distributions exhibit rapidly decaying tails, and the number of active rollouts decreases quickly as generation proceeds, providing evidence that the stopping-time distribution is sufficiently light-tailed and that truncation bias is negligible for the chosen $T_{\max}$ (see Figure 3).

Third, the rollout-level entropy rates remain bounded and stable throughout generation trajectories, supporting the bounded-moment and continuity assumptions required for consistency of the covariance and correlation estimators (see Figure 2). Finally, substantial variability is consistently observed in both generation length and entropy rate across prompts and rollouts, indicating that the conditional variances of $N_{\max}$ and $r_{\max}$ are bounded away from zero with high probability (see Figure 4 and Figure 5). Together, these empirical observations suggest that the regularity conditions required for the consistency of the prompt-averaged Pearson, Spearman, Kendall, and covariance estimators are well approximated in our experimental regime.

B. Discussions

B.1. How GenPPL and BF Capture Different Notions of Diversity: An Essay Analogy Case Study

We illustrate the difference between GenPPL and BF through an essay–word analogy, where trajectories correspond to essays and tokens correspond to individual words. For a fixed prompt x, suppose a model generates two types of essays. With probability $1/2$, it produces a short essay of length $N = 2$, where each word is selected uniformly from 100 plausible words. With probability $1/2$, it produces a long essay of length $N = 10$, where each word is selected uniformly from only 2 plausible words. For the short essays, $H(Y_{1:2} \mid N = 2, x) = 2\log 100, r_2 = \log 100$. For the long essays, $H(Y_{1:10} \mid N = 10, x) = 10\log 2, r_{10} = \log 2$.

BF treats each *trajectory* equally: $\log \mathrm{BF} = \mathbb{E}_N[r_N] = \sum_n P(N = n)\frac{H(Y_{1:n}|N=n,x)}{n} = \frac{1}{2}\log 100 + \frac{1}{2}\log 2$, which yields $\mathrm{BF} \approx 14$. This means that a typical generated trajectory behaves as if each token has roughly 14 plausible continuations. Because BF weights trajectories equally, short but highly diverse generations contribute substantially even if they contain relatively few tokens overall. In contrast, GenPPL treats each *token* equally. $\log \mathrm{GenPPL} = \frac{H(N,Y_{1:N}|x)}{\mathbb{E}[N]} = \frac{H(N|x)+\sum_n P(N=n)\,H(Y_{1:n}|N=n,x)}{\sum_n P(N=n)\,n} = \frac{6\log 2+\log 100}{6}$. Since most generated tokens come from the long low-diversity essays, we obtain $\mathrm{GenPPL} \approx 4$. Thus, a token on average has only about 4 plausible continuations.

The distinction arises from the averaging mechanism. BF averages uncertainty at the trajectory level, while GenPPL averages at the token level and therefore assigns greater weight to longer generations. Consequently, BF reflects the diversity of a *typical trajectory*, whereas GenPPL reflects the uncertainty of a *typical token*. These two measures therefore provide complementary views of generation space that cannot be captured by a single metric alone.

B.2. When To Use BF vs. GenPPL Across Different Tasks.

We refer readers to Table 4 for a concise practical comparison of when to use BF versus GenPPL across different tasks.

B.3. Comparison with Perplexity.

Perplexity (Jelinek et al., 1977) is a widely used evaluation metric in the LLM literature. To avoid confusion, we briefly contrast it with GenPPL and BF. Perplexity is defined on a fixed dataset $(y_1, \ldots, y_T)$, where $y_t \in \mathcal{V}$ is the observed token and $y_{<t}$ denotes its context. Given the model distribution $q(\cdot \mid y_{<t})$, it is defined as $\mathrm{PPL} = \exp\left(\frac{1}{T}\sum_{t=1}^{T} -\log q(y_t \mid y_{<t})\right)$, where $-\log q(y_t \mid y_{<t})$ measures how surprising the true token is under the model. Thus, perplexity evaluates predictive accuracy, with lower values indicating better fit to the data. In contrast, GenPPL and BF are defined with respect to the model's generative distribution and measure generation capacity. Namely, the size and diversity of the space of possible outputs rather than accuracy on the observed data.

C. Experiment Details

In this section, we provide additional experimental and implementation details underlying our empirical analysis. In particular, we discuss the stopping behavior of autoregressive generation and the resulting truncation bias, as well as the statistical modeling procedures used throughout the paper. We further present detailed outputs from the Beta mixed-effects regression analysis, including coefficient estimates, interaction effects, and dispersion modeling results, together with additional discussions on uncertainty allocation, semantic diversity, and length–entropy dynamics across model families and tasks.

Table 4. When to use BF vs. GenPPL across different tasks.

Task	Preferred Metric	Level	Reason
MMLU (QA)	GenPPL	Token	Requires selecting a single correct answer. Performance depends on calibrated token probabilities. Diversity across full trajectories is unnecessary.
Story Generation	BF	Trajectory	Outputs are evaluated as complete units. Diversity across stories reflects creativity and coverage of different narrative possibilities.
Math	GenPPL BF	Token Trajectory	GenPPL is primary because correctness depends on precise step-by-step reasoning. BF is secondary and useful for analyzing diversity across alternative solution strategies.
Coding	GenPPL BF	Token Trajectory	GenPPL is primary because code correctness requires token-level precision. BF is secondary and useful for comparing multiple valid implementations.
Sentence Completion	GenPPL	Token	This is a next-token prediction task, so GenPPL aligns directly with likelihood and perplexity-based evaluation.

C.1. Stopping Criteria and Empirical Truncation Bias

To ensure the generative process remains consistent with the theoretical framework, we include the `EOS` token in the sampling vocabulary $\mathcal{V} \cup \{\texttt{EOS}\}$. Crucially, during top-$k$ sampling, a rollout only terminates if `EOS` is present within the top k most probable tokens and is subsequently sampled.

While our theoretical framework allows for infinite generation length, practical estimation requires a finite truncation point $T_{\max}$. To ensure that our estimator $\widehat{\mathrm{CE}}^{\star}_{\max,P,M}$ remains a consistent proxy for the true Canopy Entropy $\mathrm{CE}^{\star}$, we set a generous token budget of $T_{\max} = 4000$. As shown in Table 5, the empirical truncation rate remains low across all model families and tasks. Notably, fine-tuned models exhibit a 0.00% truncation rate in almost all settings, confirming that aligned models naturally terminate well within the provided budget. While truncation rates are higher for base models, they remain sufficiently low (≤ 0.0432). Overall, these results demonstrate that the truncation bias term $g(T_{\max})$ defined in Theorem 2 is sufficiently small, ensuring that the observed differences in generation space are not artifacts of the maximum length constraint.

Table 5. Empirical Truncation Rates. We report the proportion of Monte Carlo rollouts that reached the maximum generation budget ($T_{\max} = 4000$ tokens) without sampling an `EOS` token. Rates are calculated over $P = 100$ prompts with $M = 100$ rollouts per prompt.

Model	MATH		STORY		CODING		COMPLETION	
	Base	Instruct	Base	Instruct	Base	Instruct	Base	Instruct
Qwen3-8B	0.0073	0.0000	0.0005	0.0000	0.0008	0.0004	0.0313	0.0000
Llama-3.1-8B	0.0105	0.0006	0.0325	0.0000	0.0095	0.0000	0.0174	0.0000
Gemma-3-12B	0.0036	0.0000	0.0066	0.0000	0.0432	0.0000	0.0119	0.0000

C.2. Completion Traps

Instead of answering the instruction, base models sometimes treat the prompt as an unfinished text prefix and continue it with a short completion fragment. Here we present some empirical examples where base models fall into such "completion

Table 6. Base–instruct comparison of length–entropy correlation measured by Spearman's ρ and Kendall's τ. Each entry reports estimate $\pm$ standard error, and absolute change is computed as instruct minus base. All numbers are rounded to two significant digits. Warm colors indicate increases, while cold colors indicate decreases; darker colors correspond to larger magnitude changes.

Task	Type	Spearman's ρ correlation			Kendall's τ correlation		
		Qwen3-8B	Llama-3.1-8B	Gemma-3-12B	Qwen3-8B	Llama-3.1-8B	Gemma-3-12B
MATH	Base	-0.19 ± 0.023	-0.56 ± 0.031	-0.27 ± 0.023	-0.12 ± 0.015	-0.36 ± 0.024	-0.17 ± 0.015
	Instruct	0.039 ± 0.032	0.13 ± 0.029	0.096 ± 0.034	0.026 ± 0.020	0.082 ± 0.019	0.056 ± 0.021
	Abs. Change	0.23 ± 0.037	0.68 ± 0.046	0.36 ± 0.045	0.15 ± 0.024	0.44 ± 0.033	0.22 ± 0.028
CODING	Base	-0.28 ± 0.018	-0.66 ± 0.014	-0.40 ± 0.011	-0.18 ± 0.013	-0.47 ± 0.012	-0.26 ± 0.0079
	Instruct	0.49 ± 0.030	0.30 ± 0.023	0.38 ± 0.032	0.32 ± 0.021	0.20 ± 0.015	0.25 ± 0.022
	Abs. Change	0.76 ± 0.035	0.97 ± 0.022	0.78 ± 0.033	0.50 ± 0.024	0.67 ± 0.016	0.51 ± 0.023
SENTENCE COMPLETION	Base	-0.0049 ± 0.025	-0.39 ± 0.019	-0.32 ± 0.016	0.0026 ± 0.017	-0.26 ± 0.013	-0.22 ± 0.011
	Instruct	0.52 ± 0.024	-0.22 ± 0.025	0.13 ± 0.035	0.36 ± 0.018	-0.14 ± 0.017	0.086 ± 0.021
	Abs. Change	0.53 ± 0.033	0.17 ± 0.029	0.46 ± 0.037	0.35 ± 0.023	0.12 ± 0.019	0.30 ± 0.023
STORY	Base	-0.36 ± 0.015	-0.32 ± 0.015	-0.30 ± 0.014	-0.25 ± 0.011	-0.21 ± 0.011	-0.20 ± 0.0099
	Instruct	0.16 ± 0.021	-0.12 ± 0.019	-0.049 ± 0.027	0.10 ± 0.014	-0.081 ± 0.012	-0.032 ± 0.017
	Abs. Change	0.52 ± 0.025	0.20 ± 0.024	0.25 ± 0.032	0.35 ± 0.017	0.13 ± 0.016	0.16 ± 0.021

traps". Some examples are listed below.

Prompt:

> Come up with an original and creative solution for the following real-world problem: Clara, a junior pre-med student, is working part-time and taking a 15 hour credit load at school *[...]* Clara is not sure how to solve her problem.

Qwen3-8B Base:

```
"In order to entice donors, you would like to take the following
approach:"
```

Llama-3.1-8B Base:

```
"How can she decide what steps she should take to solve her problem?"
```

Gemma-3-12B Base:

```
"There will be partial credit for good ideas that might just work (even if
they're not perfect).  Your solution will be judged on its creativity and
how well it works."
```

Coding prompts were especially vulnerable to premature termination caused by structural delimiters. Prompts ending with a closing code block delimiter (```) frequently produced empty responses from base models, as they likely interpret the closing backticks as an end-of-document signal from the training distribution. To ensure valid evaluations, we manually stripped these trailing delimiters to force the model to generate the intended logic rather than treating the task as already finalized.

C.3. Bootstrap Procedure

All standard errors and confidence intervals reported in Table 1, Table 2, and Table 6 are estimated using a prompt-level cluster bootstrap with $B = 2000$ replicates. In each replicate, we sample $P = 100$ prompt indices uniformly with replacement from the original set of prompts. All $M = 100$ rollouts associated with a sampled prompt are resampled jointly as a single cluster, preserving the within-prompt rollout structure.

Point estimates are always computed on the original, non-resampled data; bootstrap replicates are used only for uncertainty quantification. Reported standard errors correspond to the sample standard deviation of the bootstrap replicates with degrees of freedom = $n - 1$.

C.4. Beta Mixed-Effects Model for Semantic Diversity

The analysis is based on 2400 observations, corresponding to 4 tasks, 3 model families, and 2 model variants (base and instruct), with 100 prompts evaluated for each configuration.

Table 7. Full table of conditional mean model estimates for the Beta mixed-effects regression.

Term	Estimate	Std. Error	p-value
Intercept	-0.87953	0.07007	$< 2\text{e}{-16}$
R	0.70803	0.03340	$< 2\text{e}{-16}$
Instruct	-0.37650	0.08609	$1.23\text{e}{-05}$
$\text{ns}(\text{invN}, 4)_1$	0.15197	0.04085	0.000199
$\text{ns}(\text{invN}, 4)_2$	0.39690	0.23890	0.096641
$\text{ns}(\text{invN}, 4)_3$	0.62065	1.47024	0.672924
$\text{ns}(\text{invN}, 4)_4$	0.61022	3.10032	0.843966
Coding	-0.03721	0.03640	0.306726
Math	0.16948	0.03448	$8.83\text{e}{-07}$
Stories	-0.27563	0.02864	$< 2\text{e}{-16}$
$R \times \text{Instruct}$	1.18254	0.04642	$< 2\text{e}{-16}$
Instruct $\times \text{ns}(\text{invN}, 4)_1$	-0.13029	0.07868	0.097734
Instruct $\times \text{ns}(\text{invN}, 4)_2$	1.35321	0.31701	$1.97\text{e}{-05}$
Instruct $\times \text{ns}(\text{invN}, 4)_3$	1.00650	1.48303	0.497340
Instruct $\times \text{ns}(\text{invN}, 4)_4$	0.81421	3.10203	0.792954

Conditional Mean Modeling. Formally, we model $D_{tpm} \sim \text{Beta}(\mu_{tpm}\phi_{tpm}, (1-\mu_{tpm})\phi_{tpm})$, where $\mu_{tpm} \in (0, 1)$ denotes the conditional mean and $\phi_{tpm} > 0$ is the precision parameter. The conditional mean model is specified as $\text{logit}(\mu_{tpm}) = \alpha + \beta_1 R_{tpm} + \beta_2 \mathbf{1}\{m = \text{FT}\} + f(\text{invN}_{tpm}) + \sum_k \tau_k \mathbf{1}\{t = k\} + \beta_3 (R_{tpm} \cdot \mathbf{1}\{m = \text{FT}\}) + g(\text{invN}_{tpm})\, \mathbf{1}\{m = \text{FT}\} + u_p + v_m$ where R_{tpm} is the entropy rate, invN_{tpm} is the inverse sequence length, and $f(\cdot)$, $g(\cdot)$ are natural spline functions corresponding to nonlinear length effects and their interaction with instruction tuning. The coefficients τ_k represent fixed task effects. To account for hierarchical dependence, we include random intercepts $u_p \sim \mathcal{N}(0, \sigma_p^2), v_m \sim \mathcal{N}(0, \sigma_m^2)$, for prompt identity and model family, respectively.

In addition to the finding presented in section 5, we also observe strong nonlinear dependence on inverse sequence length. Several spline terms and their interactions with fine-tuning are significant, confirming that the relationship between diversity and length is highly nonlinear and differs between base and fine-tuned models. This reinforces that simple linear controls for length are insufficient.

Task-level effects are also pronounced. Relative to completion, mathematical reasoning exhibits significantly higher semantic diversity ($+0.169$, $p < 10^{-6}$), while story generation shows substantially lower embedding-based diversity (-0.276, $p < 2 \times 10^{-16}$), despite appearing lexically diverse. This suggests that open-ended generation can remain semantically concentrated even when surface variation is high (see Table 7).

Finally, random effects indicate that prompt-level variability remains larger than model-level variability, highlighting the strong influence of prompt semantics on generation diversity (see Table 8).

Table 8. Estimated random intercept variances for prompt identity and model family in the Beta mixed-effects regression model.

Random effect	Group	Variance	Std. Dev.
Intercept	prompt_uid	0.013937	0.11806
Intercept	model_name	0.006817	0.08256

Dispersion Modeling. The dispersion model specifies how the precision parameter ϕ_{tpm} varies across model variants,

Table 9. Dispersion model estimates for the Beta mixed-effects regression. The dispersion submodel uses a log link for the precision parameter ϕ_{tpm}.

Term	Estimate	Std. Error	p-value
Intercept	2.7544	0.2328	$< 2\text{e}{-16}$
Instruct	1.4153	0.2452	$7.83\text{e}{-09}$
Coding	2.5511	0.2186	$< 2\text{e}{-16}$
Math	2.1418	0.2149	$< 2\text{e}{-16}$
Stories	1.6534	0.1479	$< 2\text{e}{-16}$
$\text{ns}(\text{invN}, 3)_1$	-3.3780	0.5323	$2.22\text{e}{-10}$
$\text{ns}(\text{invN}, 3)_2$	-2.0039	0.8870	0.0239
$\text{ns}(\text{invN}, 3)_3$	2.6200	1.8507	0.1569
R	0.9751	0.1290	$4.12\text{e}{-14}$
Llama-3.1-8B	1.0091	0.1162	$< 2\text{e}{-16}$
Qwen3-8B	0.7270	0.1151	$2.71\text{e}{-10}$
Instruct $\times$ Coding	-2.1969	0.2852	$1.32\text{e}{-14}$
Instruct $\times$ Math	-1.5440	0.2861	$6.76\text{e}{-08}$
Instruct $\times$ Stories	-2.9273	0.2085	$< 2\text{e}{-16}$
Instruct $\times R$	-1.6085	0.2047	$3.96\text{e}{-15}$

tasks, sequence length, entropy rate, and model family. Specifically, we use the log-link specification $\log(\phi_{tpm}) = \gamma_0 + \gamma_1 \mathbf{1}\{m = \text{FT}\} + \sum_k \delta_k \mathbf{1}\{t = k\} + h(\text{invN}_{tpm}) + \gamma_2 R_{tpm} + \sum_j \eta_j \mathbf{1}\{\text{model} = j\} + \sum_k \zeta_k \mathbf{1}\{m = \text{FT}\}\mathbf{1}\{t = k\} + \gamma_3 \left(R_{tpm} \cdot \mathbf{1}\{m = \text{FT}\}\right)$, where $h(\cdot)$ is a natural spline function capturing nonlinear length-dependent dispersion effects. The task indicators δ_k allow the residual precision to vary across tasks, while the model-family indicators η_j account for systematic differences in dispersion across model families. The interaction terms between model variant and task allow fine-tuning to affect dispersion differently across tasks, and the interaction between entropy rate and model variant tests whether the relationship between entropy rate and precision changes after instruction tuning.

Empirically, the dispersion model reveals substantial heterogeneity in residual precision. Instruction-tuned models have significantly higher baseline precision than base models ($\hat{\gamma}_1 = 1.415$, $p < 0.001$), indicating lower conditional variability after accounting for the mean structure. Precision also differs strongly by task, with coding, math, and story generation all showing significantly higher precision relative to the reference task. The spline terms for inverse length indicate that dispersion varies nonlinearly with output length, while the positive main effect of entropy rate ($\hat{\gamma}_2 = 0.975$, $p < 0.001$) suggests that higher entropy rate is associated with greater precision among base models. However, the negative interaction between instruction tuning and entropy rate ($\hat{\gamma}_3 = -1.609$, $p < 0.001$) shows that this relationship is substantially weakened or reversed for fine-tuned models. The significant model-variant-by-task interactions further indicate that fine-tuning changes residual variability in a task-dependent manner (see Table 9).

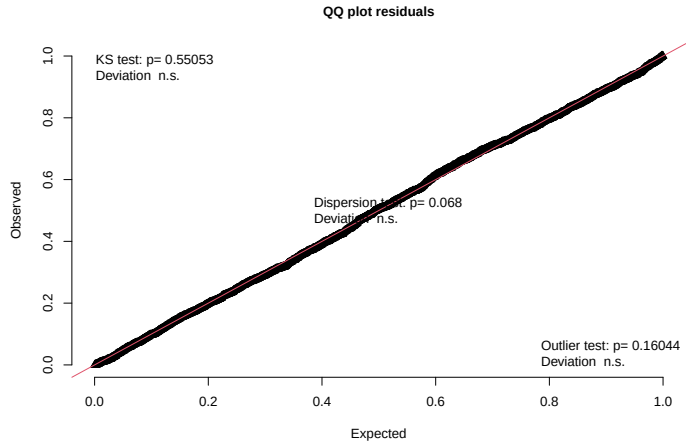

Figure 6. DHARMa residual diagnostics for the fitted Beta mixed-effects regression model. The QQ plot shows no significant deviation from the expected uniform residual distribution.

Residual Diagnosis. The figure (see Figure 6) presents a DHARMa residual diagnostic QQ plot (Hartig, 2016) for the fitted Beta mixed-effects regression model, comparing the empirical residual distribution against the expected uniform distribution. The residuals closely follow the diagonal reference line, indicating good overall model calibration and no substantial systematic deviation from the assumed distribution. The associated diagnostic tests further support model adequacy: the Kolmogorov–Smirnov test ($p = 0.551$) suggests no significant deviation from uniformity, the dispersion test ($p = 0.068$) indicates no significant over- or under-dispersion, and the outlier test ($p = 0.160$) shows no evidence of excessive outliers. Overall, these diagnostics suggest that the fitted model provides an adequate characterization of the observed data and that the regression results are statistically reliable.

C.5. Compute Resources and Runtime

All experiments were run on a single internal SLURM cluster. Each generation job uses one NVIDIA GPU (A100, H100, or H200), 8 CPU cores, and 100 GB host RAM, with a 6-hour wall-clock timeout per job. Overall, we estimate total project compute at approximately 30–40 GPU-hours, overwhelmingly dominated by rollout generation; metric computation and statistical analysis were negligible by comparison and can be completed on CPU.