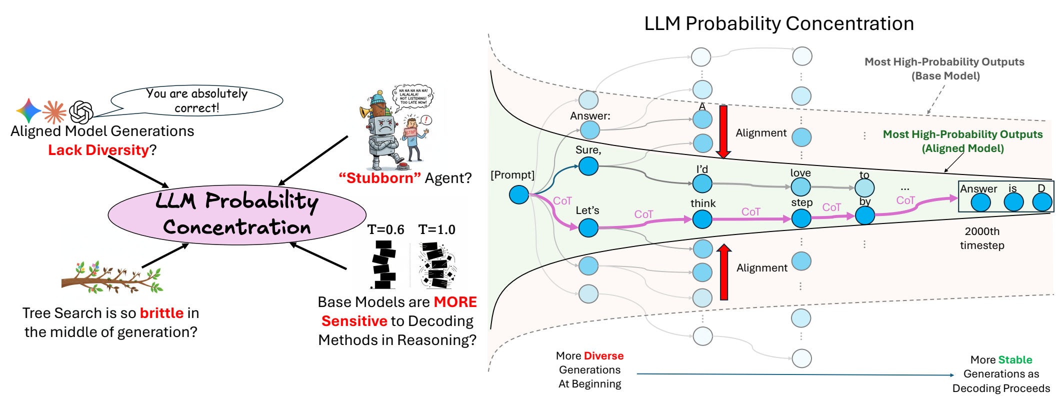
LLM PROBABILITY CONCENTRATION: HOW ALIGNMENT SHRINKS THE GENERATIVE HORIZON

Anonymous authors
Paper under double-blind review

ABSTRACT

Despite their impressive capabilities, aligned large language models (LLMs) often generate outputs that lack diversity. What drives this consistency in the generation? We investigate this phenomenon through the lens of probability concentration in the model's output distribution. To quantify this concentration, we introduce the *Branching Factor* (BF)–a token-invariant measure of the effective number of plausible next steps during generation. Our empirical analysis reveals two key findings: (1) BF often decreases as generation progresses, suggesting that LLMs become more predictable as they generate. (2) alignment tuning substantially sharpens the model's output distribution from the outset, reducing BF by nearly an order of magnitude (e.g., from 12 to 1.2) relative to base models. This stark reduction helps explain why aligned models often appear less sensitive to decoding strategies. Building on this insight, we find this consistency has surprising implications for complex reasoning. Aligned Chain-of-Thought (CoT) models (e.g., DeepSeek-distilled models), for instance, leverage this effect; by generating longer reasoning chains, they push generation into later, more deterministic (lower BF) stages, resulting in more stable outputs. We hypothesize that alignment tuning does not fundamentally change a model's behavior, but instead steers it toward stylistic tokens (e.g., "Sure") that unlock low-entropy trajectories already present in the base model. This view is supported by nudging experiments, which show prompting base models with such tokens can similarly reduce BF. Together, our findings establish BF as a powerful diagnostic for understanding and controlling LLM outputs - clarifying how alignment reduces variability, how CoT promotes stable generations, and how base models can be steered away from diversity.



(a) (b)

Figure 1: (a): LLM probability concentration connects and explains several disparate yet critical phenomena in aligned LLMs. (b): A conceptual illustration of how alignment and CoT influence the generation space of LLMs. While base models begin with high output diversity, alignment tuning sharply concentrates early probability mass, leading to more stable outputs. CoT extends this effect into later positions, flattening output sample variation and reducing sensitivity to decoding.

1 INTRODUCTION

While alignment tuning improves helpfulness and safety in large language models (LLMs), it often introduces a tradeoff: reduced output diversity (Padmakumar & He, 2024; Chakrabarty et al., 2024; Tian et al., 2024; Kirk et al., 2024; Lu et al., 2025) and increased determinism (Saparov & He, 2023; Song et al., 2024; Renze & Guven, 2024; Bigelow et al., 2024; West & Potts, 2025). Our own case study on MMLU (Hendrycks et al., 2021) confirms that aligned models exhibit reduced variance under Chain-of-Thought (CoT) prompting (Wei et al., 2022). These findings suggest a core phenomenon we call *LLM probability concentration* (Figure 1a) – i.e., a tendency to produce semantically and structurally similar outputs.

But how should we conceptualize this concentration? Autoregressive language generation is inherently a traversal through a *branching tree*: at each step, the model selects a token and expands the sequence along a specific branch. This structure, illustrated conceptually in Figure 1b, naturally defines a space of possible continuations, independent of specific token identities. To analyze how concentrated this space is during generation, we introduce the *Branching Factor* (BF) as a local, average-case measure of LLMs' output breadth, offering a microscopic lens on how local branching behavior gives rise to global output concentration.

To quantify BF, we leverage perplexity, an information-theoretic proxy for the effective size of a models output space. Unlike standard perplexity evaluations that are computed over a dataset using teacher-forcing, our goal is to estimate the perplexity of the models full output distribution over continuations–i.e., the number of diverse, high-probability sequences it is likely to generate. However, directly computing this quantity is intractable due to the exponential number of possible outputs, especially for long generations. To address this, we leverage the Asymptotic Equipartition Property (AEP) (Shannon, 1948; Mudireddy et al., 2024), which implies that the average log-probability of sufficiently long samples approximates the length-averaged entropy of the underlying distribution. This lets us estimate BF directly from naturally sampled completions, without requiring teacher-forcing or exhaustive enumeration.

To see what variables control BF, we investigate how BF varies with output length, model size, prompt complexity, and training paradigms. Our findings reveal: ① BF typically declines over the course of generation, indicating that *model output becomes increasingly constrained, and thus more predictable, with each successive token.* ② Among various factors, alignment tuning (e.g., RLHF) exerts the strongest and most consistent impact, sharply compressing the branching factor by nearly an order of magnitude (e.g., $12 \rightarrow 1.2$). This pronounced narrowing *offers a quantitative basis for the reduced output variance and decoding sensitivity observed in aligned models*, highlighting a key behavioral divergence from their base counterparts.

Is the BF reduction merely a statistical artifact, or does it indicate that the model is actively *honing in* on a narrower target space? To probe this, we conduct resampling experiments, requiring the model to choose an alternative to its top-ranked token at an intermediate generation step. We observe a sharp drop in accuracy, suggesting that aligned models not only concentrate probability mass but also commit to specific generative pathways early in the process. Building on these theoretical and empirical insights, we also hypothesize and empirically validate that aligned CoT models exhibit particularly low BF and reduced output variability under majority voting. This consistency arises because CoT encourages long reasoning chains, shifting key information to later tokens–precisely where BF tends to be lowest. As a result, different completions often converge to similar answers, *making CoT a natural stabilizer in generation.*

Why does alignment tuning exert such a dominant effect on BF? Inspired by the *superficial alignment hypothesis* (Zhou et al., 2024) and recent advances in tuning-free alignment (Lin et al., 2023; Fei et al., 2024), we hypothesize that base models already encode low-entropy conditional distributions, and that alignment tuning primarily steers generation toward certain stylistic tokens (e.g., "Sure"), thereby narrowing the conditional output space. To test this, we replicate nudging experiments (Fei et al., 2024) as a form of tuning-free alignment. We find that when conditioning base models on prefixes typically produced by aligned models, BF drops more rapidly than when conditioning on self-generated prefixes. This supports our hypothesis that base models already contain low-entropy subspaces, which alignment surfaces rather than fundamentally reshaping.

In summary, by measuring LLM probability concentration via BF (§ 4), our contributions are:

① We find that aligned models possess a BF of around 1.2, nearly an order of magnitude lower than their base counterparts. This low BF helps explain reduced output diversity and randomness for aligned models. Also, the BF diminishes as generation progresses, reducing the influence of the decoding method further and suggesting LLMs become more predictable as they generate. (§ 5)

② Using this framework, we uncover an unexpected source of consistency in complex reasoning. We show that aligned CoT models, by generating extended reasoning chains, pushes generation into later, more deterministic (lower BF) regions. This indicates CoT stabilizes generations. (§ 6)

③ Perhaps most surprisingly, we find alignment surfaces low-entropy trajectories already latent in base models. Our evidence suggests when conditioning base models on low-probability prefixes typically produced by aligned models, BF drops more rapidly than under self-generated prefixes. This raises important questions about how to achieve alignment while preserving the rich generative capacity of base models. (§ 7)

2 BACKGROUND

Autoregressive Language Models LLMs are typically trained to predict the next token and the probability of output $P\left(y_{1:N}|x;\theta\right)$ can be decomposed as: $P\left(y_{1:N}|x;\theta\right) = \Pi_{t=1}^{N} P\left(y_t|[x, y_{1:t-1}];\theta\right)$, where $y_{1:t-1}$ is the output up to position $t-1$, θ is the model parameter, and x is the prompt. Each output sample is generated via token-by-token sampling, and the generation of multiple samples naturally forms a search tree (Yao et al., 2023; Hao et al., 2023; Wan et al., 2024). Modern LLMs would go through multiple training stages. In this paper, we would use *base models* to refer to the models trained without *alignment tuning* techniques (Touvron et al., 2023), including instruction tuning and Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022; Bai et al., 2022) (e.g., "Llama-2-13B" (Touvron et al., 2023)) and *aligned models* to refer to models undergoing these additional fine-tuning stages (e.g., "Llama-2-13B-Chat").

LLM Decoding and Entropy Though LLMs are trained with a large vocabulary size $|V|$, the desired tokens often concentrate on a much smaller set of tokens under distribution $P(y_t|x, y_{1:t-1};\theta)$. Common decoding methods (Holtzman et al., 2020; Hewitt et al., 2022) utilize this observation and propose various heuristics to truncate vocabulary V as V_t at each step t. The next token is then sampled from the renormalized distribution $\tilde{P}\left(y_t|[x, y_{1:t-1}];\theta\right) = \mathbb{1}(y_t \in V_t)\frac{P(y_t|x,y_{1:t-1};\theta)}{\sum_{y_t\in V_t} P(y_t|x,y_{1:t-1};\theta)}$. Since tokens are sampled from the truncated distribution $\tilde{P}$,[1] we use $\tilde{P}$ to compute the token-level conditional entropy $\tilde{H}$ for a given prefix instance $y_{1:t-1}$:[2]

$$\tilde{H}\left(\mathrm{Y}_t|[x, y_{1:t-1}];\theta\right) = -\sum_{y_t} \tilde{P}\left(y_t|[x, y_{1:t-1}];\theta\right) \log \tilde{P}\left(y_t|[x, y_{1:t-1}];\theta\right) \tag{1}$$

To generalize, we can compute the *expected conditional entropy* over the distribution of prefix sequences $\mathrm{Y}_{1:t-1}$: $\tilde{H}\left(\mathrm{Y}_t|[x, \mathrm{Y}_{1:t-1}];\theta\right) = \mathbb{E}_{y_{1:t-1}}\tilde{H}\left(\mathrm{Y}_t|[x, y_{1:t-1}];\theta\right)$. Conventionally, we use uppercase Y to denote the *random variable* for an output and lowercase y for its specific realization.

3 CASE STUDY: IS DECODING METHOD CRUCIAL FOR MODERN LLMS?

Many prevalent decoding methods were introduced before LLMs scaled to billions of parameters and underwent multiple training stages. Additionally, model developers adopt different decoding strategies when reporting LLM capabilities (Touvron et al., 2023; Dubey et al., 2024; Yang et al., 2024; Guo et al., 2025), raising questions about the significance of decoding choices for modern LLMs. To explore this, we benchmark various decoding methods on standard LLM reasoning tasks, extending prior work (Song et al., 2024; Renze & Guven, 2024) to the latest models including DeepSeek-distlled models (Guo et al., 2025), which would generate long CoT before the final answer.[3] Specifically, we evaluate model performance on MMLU-STEM (Hendrycks et al., 2021)

[1] Our main experiments employ mild decoding settings (T=1.0, p=0.9). These settings approximate the full distribution, align with standard evaluation practices, and ensure coherent generation from base models. Stronger truncation settings are explicitly noted where applied.

[2] The common convention setting $0 \log 0 = 0$ for entropy computation is followed.

[3] For Llama-3 series models, in our prior study, we find there is only a minor performance difference between Llama-3 and Llama-3.x. We mainly use Llama-3 in this paper as it includes the most diverse model collection.

under CoT prompting across different temperatures (T= 0.6/1.0) in temperature sampling and truncation thresholds (p=0.9/1.0) in nucleus sampling (Holtzman et al., 2020). Further implementation details can be found in Appendix A.

Models	Default (T=0.6, p=0.9)	T=0.6, p=1.0	T=1.0, p=0.9	Min (T=1.0, p=1.0)	$\frac{\text{Default}-\text{Min}}{\text{Default}}\%$
Llama-3-70B-Instruct	78.50 (± 2.09)	77.60 (± 2.23)	77.50 (± 2.60)	75.90 (± 2.85)	3.31
Llama-3-70B	78.00 (± 3.52)	74.00 (± 3.80)	72.00 (± 4.38)	63.50 (± 5.02)	18.59
DeepSeek-R1-Distill-Llama-8B	66.30 (± 3.51)	65.70 (± 3.84)	62.70 (± 4.14)	59.70 (± 4.65)	9.95
Llama-3.1-8B-Instruct	63.00 (± 4.01)	61.50 (± 4.37)	57.50 (± 4.92)	50.50 (± 5.34)	19.84
Llama-3.1-8B	54.00 (± 4.61)	53.50 (± 4.92)	47.00 (± 5.21)	37.00 (± 5.48)	**31.48**

Table 1: **Experiment Results across decoding methods on STEM subset of MMLU.** We follow the common practice of using 5-shot CoT prompting. $\frac{\text{Default}-\text{Min}}{\text{Default}}\%$ indicates the maximum relative performance drop when deviating from the default decoding configuration.

The results in Table 1 reveal that for aligned models, decoding configurations have a limited impact – typically around 10% (up to 20%) relative performance changes. Among Llama-3.1-8B models, DeepSeek-distilled Llama-8B (based on Llama-3.1-8B), which is trained to generate long CoT, exhibits the smallest relative performance changes. In contrast, base models exhibit greater sensitivity, with performance varying by up to 31%. Additionally, lowering the temperature (T) generally improves performance across all models more than adjusting truncation threshold (p), though excessive reduction (e.g., greedy decoding when T=0) may lead to repetition issues (Guo et al., 2025). Based on these observations and findings in existing literature, we propose the following hypotheses:

Hypo 1 *Aligned models produce tokens with a more concentrated distribution than base models (Padmakumar & He, 2024; Bigelow et al., 2024; Lu et al., 2025; West & Potts, 2025).*

Hypo 2 *Larger models have more concentrated distributions compared with smaller models (Ye et al., 2024; Xiong et al., 2024), though may varied by tasks (Lu et al., 2025; West & Potts, 2025).*

Hypo 3 *As LLMs generate more tokens, its next-word prediction probability distribution becomes increasingly concentrated (Tian et al., 2024; Chakrabarty et al., 2024).*

Researchers often assess probability concentration using token-level metrics such as entropy or log-likelihood. However, these offer only a narrow lens on model behavior: they capture local properties but miss the global structure of the output space–how probability mass is distributed across plausible sequences. This motivates our proposal of the BF as a structural measure of generative breadth.

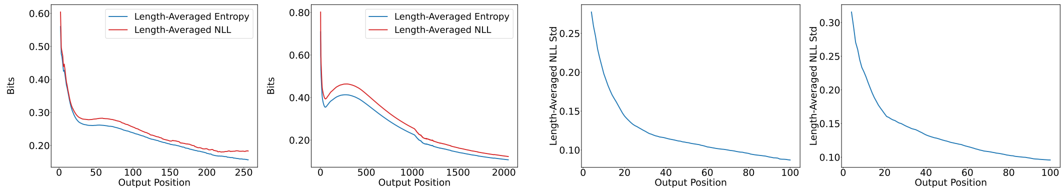
4 Measuring Branching Factor

The generative process of language models can be viewed as moving down a branching tree, with each token choice selecting a path forward. While the full tree spans $O(|V|^N)$ sequences for vocabulary size $|V|$ and sequence length N, LLMs concentrate probability mass on a far smaller subset. This high-probability subset forms a complex, sparse "effective tree" $\mathcal{T}$. To quantify how this tree expands—that is, how many options the model seriously considers at each step—we introduce the *Branching Factor* (BF). Since calculating the precise structure of $\mathcal{T}$ is intractable, we *model* it as an equivalent **balanced B-ary tree** of the same depth N. A balanced tree with depth N has B^N leaf nodes. By equating this to the number of high-probability sequences $|\mathcal{T}|$ in the effective tree, we define the Branching Factor as $B \stackrel{\text{def}}{=} |\mathcal{T}|^{1/N}$. This section describes how to compute $|\mathcal{T}|$ and B.

4.1 Intuitive Perspective: Exponentiated Entropy (Perplexity) as Branches

We propose to use the exponentiated entropy (perplexity) to quantify $|\mathcal{T}|$: $|\mathcal{T}| \stackrel{\text{def}}{=} \exp\left(H(\mathrm{Y}_{1:N}|x;\theta)\right)$. This reflects the effective number of equally probable outcomes with the same total uncertainty (O'Connor, 2013). Analogously, it is like sampling from a fair $|\mathcal{T}|$-sided die, where entropy equals $-\sum \frac{1}{|\mathcal{T}|}\log\frac{1}{|\mathcal{T}|} = H(\mathrm{Y}_{1:N}|x;\theta)$. Thus, $B(x;\theta) = \exp\left(\bar{H}(\mathrm{Y}_{1:N}|x;\theta)\right)$ where $\bar{H}(\mathrm{Y}_{1:N}|x;\theta) = \frac{1}{N}\tilde{H}\left(\mathrm{Y}_{1:N}|x;\theta\right)$ is the averaged entropy per output token up to position N. A larger $B(x;\theta)$ indicates a greater potential for diverse outputs.

For short outputs, where it's tractable to sample sufficiently many sequences to closely estimate the conditional entropy at each position, we can estimate the BF by computing the conditional entropy



(a) MMLU (b) BBCNewsLatest (c) MMLU (Std) (d) BBCNewsLatest (Std)

Figure 2: **AEP Empirical verification for Llama-3-8B-Instruct.** (**a, b**): length-averaged NLL closely tracks length-averaged Entropy. (**c, d**): Standard deviation of length-averaged NLL diminishes with output length.

at each position and then aggregating as:

$$B(x;\theta) \approx \exp\left(\frac{1}{M}\sum_{i=1}^{M}\frac{\sum_{t=1}^{|y^{(i)}|}\tilde{H}(\mathrm{Y}_t|[x, y_{1:t-1}^{(i)}];\theta)}{|y^{(i)}|}\right) \tag{2}$$

where $\tilde{H}(\mathrm{Y}_t|[x, y_{1:t-1}^{(i)}];\theta)$ is the entropy of the distribution at position t for sample i.

4.2 Practical BF Estimator via Asymptotic Equipartition Property

While the above approach works for short outputs, it is challenging for longer sequences. Sampling a tiny fraction of the exponentially large output space leads to a systematic underestimation of the true entropy, an issue we demonstrate empirically in Appendix C. In such cases, we show that when LLMs generate sufficiently long outputs, the average log-probability of each output sequence will be roughly the same, and can approximate average output entropy well, following the Asymptotic Equipartition Property (AEP) (Shannon, 1948; Breiman, 1957; Cover, 1999). The original AEP proof requires additional assumptions about the generation process, such as that it needs to be stationary and ergodic, often violated by LLMs. But as noted by Mudireddy et al. (2024), these assumptions are unnecessary if we do not require $\bar{H}\left(\mathrm{Y}_{1:N}|x;\theta\right)$ to converge to a constant:

Theorem 4.1 (AEP for LLMs) *Given* $0 < \epsilon < 1$*, we have:*

$$\lim_{N\to\infty} P\left(\left|-\frac{1}{N}\log\tilde{P}\left(y_{1:N}|x;\theta\right) - \bar{H}\left(\mathrm{Y}_{1:N}|x;\theta\right)\right| < \epsilon\right) = 1 \tag{3}$$

Theorem 4.1 is equivalent to the statement: for sufficiently large N, the probability of any length-N high-probability output $y_{1:N}$ under $\tilde{P}$ can be approximated as $\exp\left(-N\bar{H}(\mathrm{Y}_{1:N}|x;\theta)\right)$, rendering log-probability asymptotically ineffective for distinguishing among them.

The proof can be found in Appendix H.[4] This result holds under the mild and practical assumptions that the model is autoregressive and runs with finite precision (e.g., `bfloat16`)—conditions met by most popular LLMs. As an empirical demonstration, we plot the standard deviation of the average negative log-likelihood of Llama-3-8B-Instruct over multiple datasets[5] in Figure 2, where we can see that with the increased output length, the difference between length-averaged entropy and negative log-likelihood (NLL) is reduced, and the standard deviation of average NLL also quickly reduces within the first 50 output tokens. A key additional benefit of our AEP-based estimator is computational efficiency, as it only requires the log-probability of sampled tokens, avoiding potential GPU memory surge for full-distribution computation, enabling long-form generation analysis.

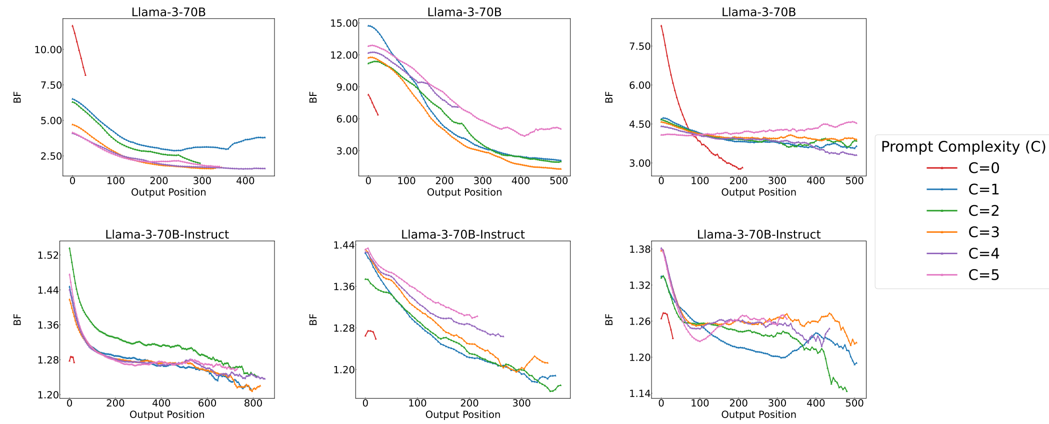
Therefore, for long sequences,[6] we can estimate BF using the NLL of sampled sequences as:

$$B(x;\theta) \approx \exp\left(-\frac{1}{M}\sum_{i=1}^{M}\frac{1}{|y^{(i)}|}\log\tilde{P}\left(y_{1:|y^{(i)}|}|x;\theta\right)\right) \tag{4}$$

[4]We provide a simplified proof with minor changes to the original proof of Mudireddy et al. (2024). Notably, our goal is only to show the approximation between length-averaged log-likelihood and entropy for a typical sequence, so we do not require stricter assumptions like ergodicity or stationarity.

[5]For dataset-specific details, we refer readers to Appendix B.

[6]For short sequences, direct computation (§ 4.1) is preferred, as AEP estimation is unreliable.



(a) Creative StoryGen (b) Random Strings (c) BBCNewsLatest

Figure 3: **Shrinking BF with output length over various tasks for Llama-3-70B and Llama-3-70B-Instruct.** For better visualization, we compute the exponential moving averaged values of BF with the smoothing factor set as 0.1.

This approach allows us to compute BF in a sample-efficient way. For task-wise BF, we simply compute it via averaging all instance-wise BF: $B(\mathrm{X};\theta) = \sum_x p(x)B(x;\theta)$.

5 Benchmarking and Attributing Branch Factors

Models and Sampling. We run experiments on models from Llama-2 (Touvron et al., 2023) and Llama-3 (Dubey et al., 2024) families as they are widely-used open-weight model families. For each model family, we include both base and aligned models to investigate how alignment tuning affects BF. We set p=0.9 and T=1.0 to sample outputs to conform with the setting for most datasets.

We set M=50 sequences to estimate BF, which yields a reliable estimation across datasets in prior studies. For aligned models, we apply the official chat templates to prompts. In addition, we carefully control the lengths of all inputs plus outputs to be within the context window of the models.

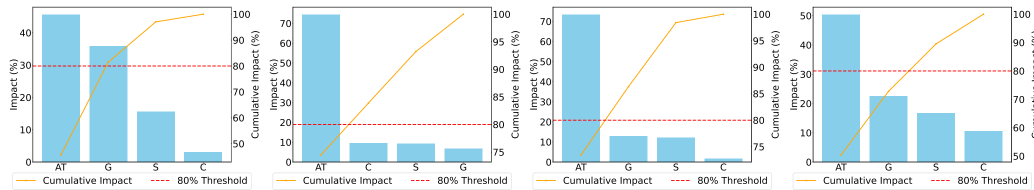
Tasks. We consider a variety of tasks covering common application scenarios of LLM generation, including reasoning and open-ended generation: MMLU (Hendrycks et al., 2021) (Reasoning), Cognac (Chen et al., 2022) (Controlled Generation), BBCLatestNews (Li et al., 2024b) (News Generation), and Creative Story Generation (Chakrabarty et al., 2024) (Creative Generation). To test subjective randomness bias (Bigelow et al., 2024), we also prepare a synthetic task Random Strings where the prompt is generated via random characters. See Appendix B for dataset details.

Impact Factors (IFs). We consider modulating these factors that may impact BF computations: Prompt Complexity (C), Alignment Tuning ($AT \in \{\text{Instruct}, \text{Base}\}$), Model Size ($S \in \{8\text{B}/13\text{B}, 70\text{B}\}$), and Model Generation ($G \in \{2, 3\}$). C controls the informativeness of the input prompt x (e.g., the number of banned words in Cognac, the number of in-context samples in MMLU). Intuitively, providing more information in x should make the model more confident in its outputs, resulting in a lower BF. Dataset-specific setups for C are detailed in Appendix B. AT, S, G represent model-wise variations to explore how different configurations of θ affect $B(\mathrm{X};\theta)$.

5.1 BF Dynamic in Generation Process

Both BF and the output length N are functions of the output Y, and BF computation relies on N. To avoid confounding effects, we first analyze how BF varies with N before intervening IFs In Figure 3, we demonstrate BF trajectories over different output positions by running Llama-3-70B and Llama-3-70B-Instruct on three representative tasks. Specifically, we compute BF over every five output tokens, conditioning on the prompt and all previously generated output tokens.[7]

[7] See Appendix D for full results across all models and tasks. Our findings also generalizes to summarization, multilingual tasks, and Qwen family (Team, 2025) (Appendix G).



(a) Cognac (b) MMLU (c) BBCNews (d) Creative StoryGen

Figure 4: **Pareto Analysis of BF across various IFs.** AT indicates whether the model is aligned. C denotes the prompt complexity. S refers to model size, and G refers to model generation (Llama-2 vs. Llama-3). Across all settings, alignment tuning has the most pronounced impact on BF.

As we can see, first, **the average BF for the base model** ($\approx$ 12) **is roughly ten times higher than the aligned model** (≈ 1.2). Therefore, there are actually very few candidate next-token to be truncated in decoding for the aligned models. This explains why the decoding method would assert weaker effects for aligned models, as we see in § 3. Also, in most cases, **BF would often drop smoothly as more output tokens are generated**. Under the same task, when $C > 0$, different C mainly controls the starting point and the rate of decreasing, while in the end, they would converge to roughly the same point. When almost zero knowledge is provided ($C = 0$), the output will end much earlier compared to $C > 0$ cases. These findings also provide support that the future token generation is gradually becoming predictable and the model may have a certain generation plan to follow, resonating with recent observation in interpretability (Pal et al., 2023; Wu et al., 2024; Li et al., 2024a) and inference acceleration (Cai et al., 2024; Welleck et al., 2024).

We further examine potential confounds such as prompt likelihood and data contamination in Appendix L, and find they do not fully account for the observed BF reductions.

5.2 Pareto Analysis of BF

We perform a Pareto analysis to identify the relative influence of all IFs of BF. For each factor D_i, we define the unnormalized *Impact* $\tilde{I}(D_i)$ as the average absolute pairwise difference in BF when varying D_i while holding other dimensions constant:

$$\tilde{I}(D_i) = \frac{\sum_{d_i, d_j \in \text{Domain}(D_i), d_i \neq d_j} |\text{Avg}(\text{B}(\cdot|D_i = d_i)) - \text{Avg}(\text{B}(\cdot|D_i = d_j))|}{|\text{Domain}(D_i)| * |\text{Domain}(D_i) - 1|}. \tag{5}$$

Then we normalize it as $I(D_i) = \frac{\tilde{I}(D_i)}{\sum \tilde{I}(D_i)}$. The results, shown in Figure 4, indicate that **alignment tuning is the most influential factor affecting BF**, surpassing model size, model generation, and prompt complexity by a large margin. For tasks with richer inputs–such as MMLU (with more in-context examples) and BBCLatestNews (with more headlines)–prompt complexity C and model size M emerge as the next most impactful factors. In contrast, for open-ended tasks like Cognac and Story Generation, model generation G–particularly improvements from Llama-2 to Llama-3–plays a more dominant role. This shift likely reflects gains the use of larger, more diverse datasets in training (Dubey et al., 2024).

Among these secondary factors, prompt complexity C has a noteworthy effect: contrary to intuition, more context provided in the prompt does not always reduce BF but can in fact increase it, potentially due to the cognitive burden of processing complex linguistic structures. A detailed case study and comprehensive task-wise BF results are presented in Appendices E and F.

6 Reduced BF Neglects Alternative Generation and Forking

Building on our findings that BF declines over the generation process (§ 5.1) and is lower in aligned models (§ 5.2), we derive a practical implication: aligned CoT models, by starting with low BF and delaying decisive tokens, shrink the output space more aggressively and produce fewer high-probability variants. To test this, we evaluate output variability on MMLU-STEM using 200 samples per model, measuring the standard deviation of Majority@K accuracy for $K = 1, 3, 8, 16$ under temperature $T = 0.6$ and truncation threshold $p = 0.9$. As shown in Table 2, among models with similar capacity, those with lower BF–especially the aligned CoT model–exhibit markedly lower variance. This confirms that BF is a reliable predictor of sampling consistency.

Model	Maj@1 Std	Maj@3 Std	Maj@8 Std	Maj@16 Std	BF
DeepSeek-R1-Distill-Llama-70B	**14.34**	**8.29**	**4.99**	**3.21**	**1.23**
Llama-3-70B-Instruct	16.37	11.40	7.50	5.12	1.28
Llama-3-70B	27.78	19.53	13.22	9.23	1.31
DeepSeek-R1-Distill-Llama-8B	**27.10**	**20.91**	**13.93**	**9.14**	**1.23**
Llama-3.1-8B-Instruct	31.54	24.64	17.30	12.90	1.31
Llama-3.1-8B	36.41	29.78	20.43	14.05	1.35

Table 2: **Majority Voting@K standard deviation on MMLU-STEM with** 200 **samples.** We compute the standard deviation over 100 bootstrapping trials, each using 64 samples per instance. We set $T = 0.6, p = 0.9$ to match standard benchmarking settings, differing from $T = 1.0, p = 0.9$ setup in § 5. Lower temperature concentrates probability mass on fewer tokens, reducing BF and making direct comparisons more difficult. Still, BF remains a strong predictor of standard deviation.

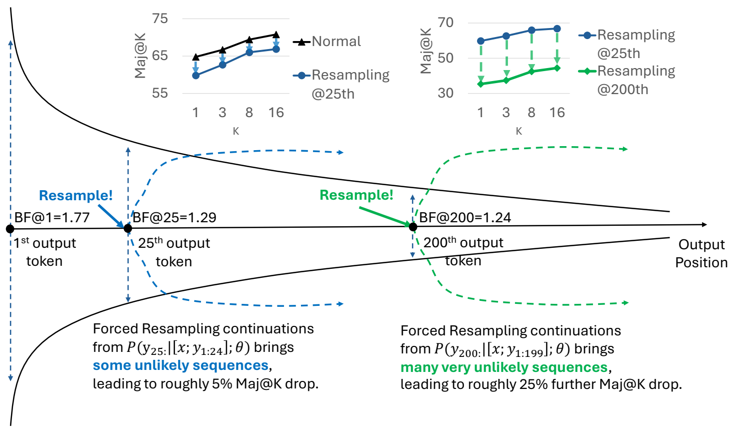


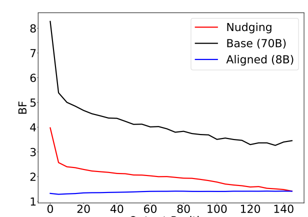
Figure 5: **Resampling from different output positions to assess the effect of interrupting BF reduction**. We resample new continuations at the 25th and 200th output token of DeepSeek-Distilled Llama-8B MMLU outputs. Results show substantial performance drops at both positions.

But is this narrowing of BF merely a reflection of concentrated token probabilities, or does it reflect a deeper commitment to specific generative paths? To examine this, we conduct a resampling experiment: at various points in the generation process, we force the model to take a different path by replacing the remainder of the output with a newly sampled continuation. As shown in Figure 5, performance drops sharply when resampling occurs at a later, lower-BF position in the sequence. This suggests that aligned models arent just concentrating probability mass locally, but are actively locking into trajectories, making late-stage deviations more error-prone. In practice, this highlights a key application of BF: **parallel sampling should be applied early, while BF remains high**, to ensure meaningful diversity and avoid quality degradation.

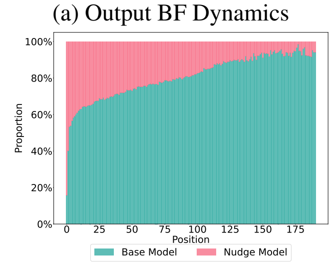
7 How does Alignment Tuning Impact BF?

Why does alignment tuning exert such a pronounced effect on BF? Building on the superficial alignment hypothesis (Zhou et al., 2024) ("*Alignment tuning might simply teach base LLMs to select a subdistribution of data formats for interacting with users.*") and recent tuning-free alignment work (Lin et al., 2023; Fei et al., 2024), we hypothesize base models already encode low-entropy conditional distributions. In this view, alignment tuning doesn't reshape generation from scratch, but instead nudge the model toward stylistic tokens (e.g., "Sure"), thereby narrowing the conditional distribution.

To test this hypothesis, we reproduce the nudging experiments (Fei et al., 2024), over Just-Eval-Instruct (Lin et al., 2023) and MMLU datasets. We employ Llama-3-70B for drafting most outputs. However, when the base model's Top-1 probability is low, we apply nudging by switching to Llama-3-8B-Instruct to generate a single word. BF was computed as in prior experiments. The results, shown



(a) Output BF Dynamics



(b) Nudging Ratio Histogram

Figure 6: Nudging Experiments over Just-Eval-Instruct.

in Figure 6,[8] indicate that after most nudging occurs early in the generation process – indicating the prefix generated by the nudging model is of low probability. These observations collectively support our hypothesis. Considering that nudging not only reduces BF but also improves aligned model performance on these tasks (Fei et al., 2024), our results highlight the dual effect of alignment training: reducing BF while preserving or even enhancing task performance.

8 RELATED WORKS

Uncertainty Quantification for LLM. Uncertainty quantification (UQ) for LLMs has gained significant attention due to its importance in real-world applications, particularly in high-stakes domains (Desai & Durrett, 2020; Jiang et al., 2021; Wang et al., 2022; Kadavath et al., 2022; Xiong et al., 2024; Ye et al., 2024; Gupta et al., 2024). Existing methods typically address closed-domain tasks such as classification and question-answering, where outputs are discrete and easier to assess. However, as Kuhn et al. (2023) note, these approaches often overlook challenges specific to open-ended generation, such as semantic equivalence across outputs. They introduce "semantic entropy" to quantify uncertainty in LLM output space by first clustering the sampled output and then quantifying uncertainty over cluster distribution. This method empirically works well in hallucination detection (Farquhar et al., 2024). In this paper, we focus on investigating the probability concentration phenomenon for LLMs. We introduce BF to quantify this concentration, whichapplies broadly across tasks without imposing strong assumptions on output categories.

Reduced Diversity in Aligned Models. Recent studies have consistently shown that alignment tuning reduces output diversity in language models (Perez et al., 2022; Padmakumar & He, 2024; Chakrabarty et al., 2024; Tian et al., 2024; Kirk et al., 2024; Lu et al., 2025; West & Potts, 2025). Our work aims at connecting reduced diversity with related observations on diminished randomness and robustness in aligned models (Saparov & He, 2023; Song et al., 2024; Renze & Guven, 2024; Bigelow et al., 2024), and proposes a unifying explanation: increased probability concentration. Traditional diversity metrics such as n-gram lexical diversity (Li et al., 2016) are sensitive to vocabulary size and output length (Liu et al., 2022; Tevet & Berant, 2021; Guo et al., 2024; Kirk et al., 2024) and cannot work well with most recent long CoT models. In Appendix K, we demonstrate that lexical diversity poorly correlates with BF and fails to robustly measure generation concentration.

Our work also resonates with information density research in cognitive science and linguistic theories, and we present a short discussion in Appendix J.

9 DISCUSSION

Practical Implications. A key practical implication of our findings is that reduced BF neglects alternative generations and forking. Consequently, simply tweaking decoding parameters (e.g., temperature), is unlikely to restore diversity without severely degrading quality (Renze & Guven, 2024). Our work suggests that efforts to mitigate diversity loss should target the training process itself. This represents a more promising, albeit more challenging, direction. Future work could involve curating more diverse alignment data or designing novel training objectives that balance instruction-following with distributional diversity (Wang et al., 2024; Kwon et al., 2024; Lanchantin et al., 2025; Chung et al., 2025). System-level interventions (e.g., model collaboration) also present a viable path forward (Fei et al., 2024; Lu et al., 2024; Venkatraman et al., 2025; Ismayilzada et al., 2025). While our paper's primary contribution is diagnostic, we believe this foundational understanding is a necessary prerequisite for developing such effective countermeasures.

Which Training Stage Reduces BF Most? While our analysis does not disentangle which stage of alignment (supervised fine-tuning, reward modeling, or RL) contributes most to BF reduction, we hypothesize that the RL stage is the primary driver. This hypothesis is informed by concurrent work investigating entropy dynamics during RL for language models. These studies suggest that the nature of RL training, particularly with sparse reward signals, is the main cause of probability distribution shrinkage (Cui et al., 2025; Wu et al., 2025b). A fine-grained, checkpoint-by-checkpoint analysis of BF throughout the entire alignment pipeline remains an important direction for future investigation.

[8]We present results on Just-Eval-Instruct in short of space. MMLU results are included in Appendix I.

10 Ethics Statement

Our work identifies a key dynamic in modern LLMs: alignment tuning significantly reduces the Branching Factor (BF), leading to more homogenized and predictable outputs. While this can be beneficial, it also carries potential negative societal impacts. In applications such as automated content generation, creative writing, or decision-support systems (Padmakumar & He, 2024; Sorensen et al., 2024; Wu et al., 2025a; Murthy et al., 2025; Ashkinaze et al., 2025; Lake et al., 2025), this reduction in diversity could inadvertently reinforce social biases, stifle creativity, and limit the exploration of novel ideas. We believe that formally understanding and quantifying the mechanisms of probability concentration, as we do in this paper, is a critical and necessary first step toward developing alignment techniques that mitigate these risks and foster models that are not only helpful and harmless but also diverse and robust. All experiments in this work were conducted on publicly available datasets.

11 Reproducibility Statement

To ensure the reproducibility of our findings, we will release all code used for our experiments upon publication. All datasets used in this paper are publicly available, with detailed descriptions, sources, and processing steps provided in Appendix B. Our experimental setup, including hyperparameters and model configurations, is detailed in § 5. The formal proof for the Asymptotic Equipartition Property for LLMs, central to our BF estimation, is provided in its entirety in Appendix H.

12 Usage of Large Language Models

In this work, besides running LLMs in experiments, we use LLMs for the following purposes:

1. Aid or Polish Writing (Gemini 2.5 Pro, ChatGPT 4/5)
2. Literature Retrieval and Discovery (e.g., finding related work) (Gemini 2.5 Pro Deep Research, ChatGPT Deep Research)
3. Assisting Code Writing and Debugging (Claude 3.5 Sonnet)

We fully understand the responsibility of using LLMs in academic research. We carefully monitor any potential problems, such as plagiarism or scientific misconduct (e.g., fabrication of facts) when using LLMs. We make sure these problems do not occur in the paper.

References

Joshua Ashkinaze, Julia Mendelsohn, Li Qiwei, Ceren Budak, and Eric Gilbert. How ai ideas affect the creativity, diversity, and evolution of human ideas: evidence from a large, dynamic experiment. In *Proceedings of the ACM Collective Intelligence Conference*, pp. 198–213, 2025.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Eric J Bigelow, Ekdeep Singh Lubana, Robert P. Dick, Hidenori Tanaka, and Tomer Ullman. In-context learning dynamics with random binary sequences. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=62K7mALO2q`.

Leo Breiman. The individual ergodic theorem of information theory. *The Annals of Mathematical Statistics*, 28(3):809–811, 1957.

Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D. Lee, Deming Chen, and Tri Dao. Medusa: Simple LLM inference acceleration framework with multiple decoding heads. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=PEpbUobfJv`.

Tuhin Chakrabarty, Philippe Laban, Divyansh Agarwal, Smaranda Muresan, and Chien-Sheng Wu. Art or artifice? large language models and the false promise of creativity. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pp. 1–34, 2024.

Howard Chen, Huihan Li, Danqi Chen, and Karthik Narasimhan. Cognac: Controllable text generation with language constraints. In *arXiv*, 2022.

John Joon Young Chung, Vishakh Padmakumar, Melissa Roemmele, Yuqian Sun, and Max Kreminski. Modifying large language model post-training for diverse creative writing. *arXiv preprint arXiv:2503.17126*, 2025.

Joel E Cohen. Markov's inequality and chebyshev's inequality for tail probabilities: A sharper image. *The American Statistician*, 69(1):5–7, 2015.

Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, et al. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

Shrey Desai and Greg Durrett. Calibration of pre-trained transformers. *arXiv preprint arXiv:2003.07892*, 2020.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024.

Yu Fei, Yasaman Razeghi, and Sameer Singh. Nudging: Inference-time alignment via model collaboration. *arXiv preprint arXiv:2410.09300*, 2024.

Dmitriy Genzel and Eugene Charniak. Entropy rate constancy in text. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 199–206, 2002.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Yanzhu Guo, Guokan Shang, and Chloé Clavel. Benchmarking linguistic diversity of large language models. *arXiv preprint arXiv:2412.10271*, 2024.

Neha Gupta, Harikrishna Narasimhan, Wittawat Jitkrittum, Ankit Singh Rawat, Aditya Krishna Menon, and Sanjiv Kumar. Language model cascades: Token-level uncertainty and beyond. In *The Twelfth International Conference on Learning Representations*, 2024.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Hong, Zhen Wang, Daisy Wang, and Zhiting Hu. Reasoning with language model is planning with world model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 8154–8173, 2023.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=d7KBjmI3GmQ`.

John Hewitt, Christopher D Manning, and Percy Liang. Truncation sampling as language model desmoothing. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 3414–3427, 2022.

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

Mete Ismayilzada, Antonio Laverghetta Jr, Simone A Luchini, Reet Patel, Antoine Bosselut, Lonneke van der Plas, and Roger Beaty. Creative preference optimization. *arXiv preprint arXiv:2505.14442*, 2025.

T Jaeger and Roger Levy. Speakers optimize information density through syntactic reduction. *Advances in neural information processing systems*, 19, 2006.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2021.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of RLHF on LLM generalisation and diversity. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=PXD3FAVHJT.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023.

Oh Joon Kwon, Daiki E Matsunaga, and Kee-Eung Kim. Gdpo: Learning to directly align language models with diversity using gflownets. *arXiv preprint arXiv:2410.15096*, 2024.

Thom Lake, Eunsol Choi, and Greg Durrett. 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.

Jack Lanchantin, Angelica Chen, Shehzaad Dhuliawala, Ping Yu, Jason Weston, Sainbayar Sukhbaatar, and Ilia Kulikov. Diverse preference optimization. *arXiv preprint arXiv:2501.18101*, 2025.

Roger Levy. Expectation-based syntactic comprehension. *Cognition*, 106(3):1126–1177, 2008.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and William B Dolan. 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.

Margaret Li, Weijia Shi, Artidoro Pagnoni, Peter West, and Ari Holtzman. Predicting vs. acting: A trade-off between world modeling & agent modeling. *arXiv preprint arXiv:2407.02446*, 2024a.

Yucheng Li, Frank Guerin, and Chenghua Lin. Latesteval: Addressing data contamination in language model evaluation through dynamic and time-sensitive test construction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 18600–18607, 2024b.

Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. The unlocking spell on base llms: Rethinking alignment via in-context learning. In *The Twelfth International Conference on Learning Representations*, 2023.

Siyang Liu, Sahand Sabour, Yinhe Zheng, Pei Ke, Xiaoyan Zhu, and Minlie Huang. Rethinking and refining the distinct metric. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 762–770, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.86. URL https://aclanthology.org/2022.acl-short.86/.

Li-Chun Lu, Shou-Jen Chen, Tsung-Min Pai, Chan-Hung Yu, Hung-yi Lee, and Shao-Hua Sun. Llm discussion: Enhancing the creativity of large language models via discussion framework and role-play. In *First Conference on Language Modeling*, 2024.

Ximing Lu, Melanie Sclar, Skyler Hallinan, Niloofar Mireshghallah, Jiacheng Liu, Seungju Han, Allyson Ettinger, Liwei Jiang, Khyathi Chandu, Nouha Dziri, and Yejin Choi. AI as humanity's salieri: Quantifying linguistic creativity of language models via systematic attribution of machine text against web text. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ilOEOIqolQ.

Kyle Mahowald, Evelina Fedorenko, Steven T Piantadosi, and Edward Gibson. Info/information theory: Speakers choose shorter words in predictive contexts. *Cognition*, 126(2):313–318, 2013.

Clara Meister, Tiago Pimentel, Patrick Haller, Lena Jäger, Ryan Cotterell, and Roger Levy. Revisiting the uniform information density hypothesis. *arXiv preprint arXiv:2109.11635*, 2021.

William Merrill and Ashish Sabharwal. The parallelism tradeoff: Limitations of log-precision transformers. *Transactions of the Association for Computational Linguistics*, 11:531–545, 2023.

George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11): 39–41, 1995.

Avinash Mudireddy, Tyler Bell, and Raghu Mudumbai. Slaves to the law of large numbers: An asymptotic equipartition property for perplexity in generative language models. *arXiv preprint arXiv:2405.13798*, 2024.

Sonia Krishna Murthy, Tomer Ullman, and Jennifer Hu. One fish, two fish, but not the whole sea: Alignment reduces language models' conceptual diversity. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *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. 11241–11258, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.561. URL https://aclanthology.org/2025.naacl-long.561/.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. Dont give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 1797–1807, 2018.

Brendan O'Connor. Perplexity as branching factor as shannon diversity index, 2013. URL https://brenocon.com/blog/2013/01/perplexity-as-branching-factor-as-shannon-diversity-index/. Accessed: 2024-09-28.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Vishakh Padmakumar and He He. Does writing with language models reduce content diversity? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Feiz5HtCD0.

Koyena Pal, Jiuding Sun, Andrew Yuan, Byron C Wallace, and David Bau. Future lens: Anticipating subsequent tokens from a single hidden state. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pp. 548–560, 2023.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red teaming language models with language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 3419–3448, 2022.

Matthew Renze and Erhan Guven. The effect of sampling temperature on problem solving in large language models. *arXiv preprint arXiv:2402.05201*, 2024.

Abulhair Saparov and He He. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=qFVVBzXxR2V.

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

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Shivalika Singh, Freddie Vargus, Daniel Dsouza, Börje Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, Laura OMahony, et al. Aya dataset: An open-access collection for multilingual instruction tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11521–11567, 2024.

Yifan Song, Guoyin Wang, Sujian Li, and Bill Yuchen Lin. The good, the bad, and the greedy: Evaluation of llms should not ignore non-determinism. *arXiv preprint arXiv:2407.10457*, 2024.

Taylor Sorensen, Jared Moore, Jillian Fisher, Mitchell Gordon, Niloofar Mireshghallah, Christopher Michael Rytting, Andre Ye, Liwei Jiang, Ximing Lu, Nouha Dziri, et al. Position: a roadmap to pluralistic alignment. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 46280–46302, 2024.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Guy Tevet and Jonathan Berant. 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.

Yufei Tian, Tenghao Huang, Miri Liu, Derek Jiang, Alexander Spangher, Muhao Chen, Jonathan May, and Nanyun Peng. Are large language models capable of generating human-level narratives? In *EMNLP*, 2024.

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

Saranya Venkatraman, Nafis Irtiza Tripto, and Dongwon Lee. Collabstory: Multi-llm collaborative story generation and authorship analysis. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 3665–3679, 2025.

Vivek Verma, Nicholas Tomlin, and Dan Klein. Revisiting entropy rate constancy in text. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 15537–15549, 2023.

Ziyu Wan, Xidong Feng, Muning Wen, Stephen Marcus McAleer, Ying Wen, Weinan Zhang, and Jun Wang. Alphazero-like tree-search can guide large language model decoding and training. In *Forty-first International Conference on Machine Learning*, 2024.

Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse kl: Generalizing direct preference optimization with diverse divergence constraints. In *The Twelfth International Conference on Learning Representations*, 2024.

Yuxia Wang, Daniel Beck, Timothy Baldwin, and Karin Verspoor. Uncertainty estimation and reduction of pre-trained models for text regression. *Transactions of the Association for Computational Linguistics*, 10:680–696, 2022.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.

Sean Welleck, Amanda Bertsch, Matthew Finlayson, Hailey Schoelkopf, Alex Xie, Graham Neubig, Ilia Kulikov, and Zaid Harchaoui. From decoding to meta-generation: Inference-time algorithms for large language models. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=eskQMcIbMS. Survey Certification.

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

Fan Wu, Emily Black, and Varun Chandrasekaran. Generative monoculture in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=yZ7sn9pyqb.

Fang Wu, Weihao Xuan, Ximing Lu, Zaid Harchaoui, and Yejin Choi. The invisible leash: Why rlvr may not escape its origin. *arXiv preprint arXiv:2507.14843*, 2025b.

Wilson Wu, John X Morris, and Lionel Levine. Do language models plan ahead for future tokens? In *The First Conference on Language Modeling*, 2024.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, YIFEI LI, Jie Fu, Junxian He, and Bryan Hooi. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=gjeQKFxFpZ.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023.

Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek F Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. Benchmarking llms via uncertainty quantification. *arXiv preprint arXiv:2401.12794*, 2024.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36, 2024.

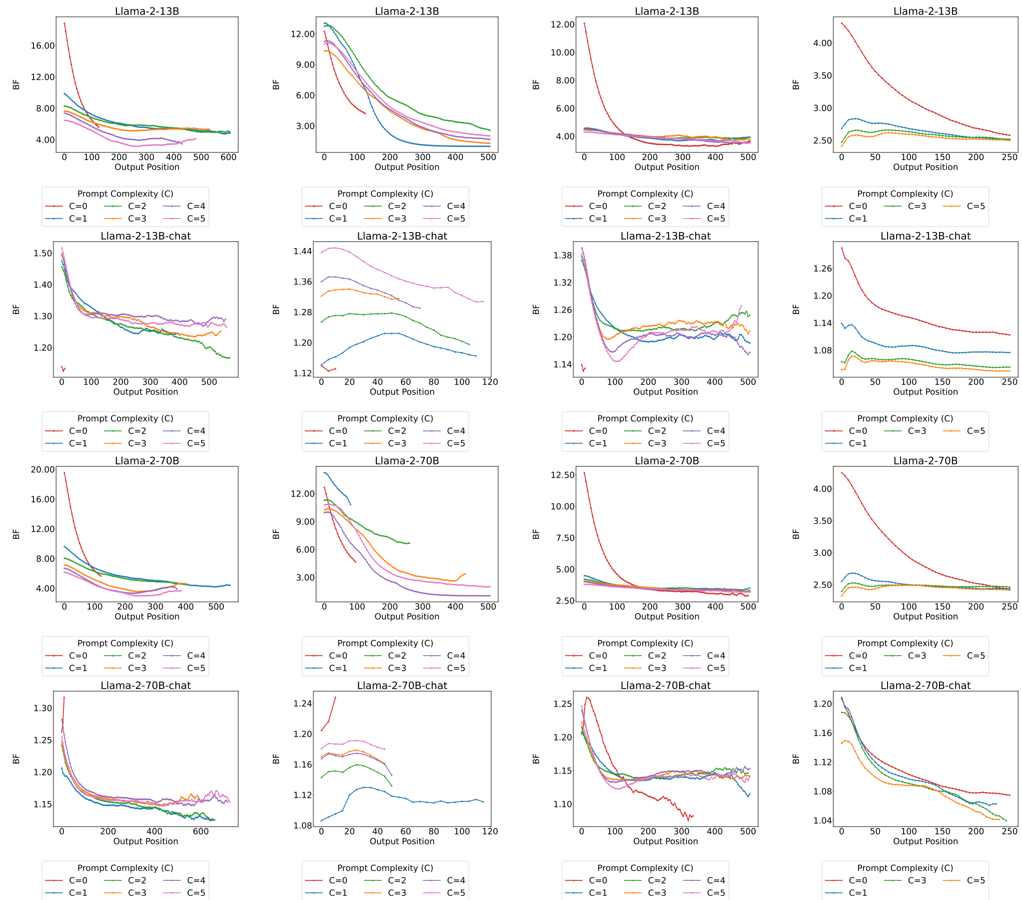


Figure 7: **BF Output Dynamic for Llama-2-families.** For better visualization, we compute the exponential moving averaged values of perplexity with the smoothing factor set as 0.1.

A Case Study Implementation Details

We use the scripts in Qwen-2.5-Math (Yang et al., 2024) for standard reasoning benchmarks.[9] We sample 200 examples from MMLU-STEM and compute the performance numbers under 64 trials and report the average performance.

B Dataset-Specific Processing

For all datasets we used in the paper, we carefully controlled whether the prompt length and expected output length would exceed the model's maximum length.

MMLU (Hendrycks et al., 2021) is a widely-used multiple-choice reasoning question. Unless otherwise explained, we use the full test set of MMLU to avoid potential contamination, following benchmarking settings reported in most LLM technical reports(Touvron et al., 2023; Dubey et al., 2024; Guo et al., 2025). We formulate prompt complexity C as the number of in-context samples. For example, $C = 1$ means we only add one in-context sample. For prompting setup and postprocessing details, we follow the standard implementation in Qwen-2.5-Math (Yang et al., 2024).

Cognac (Chen et al., 2022) is a controlled generation task requiring language model *not* to generate specified banned words provided in the prompt. We use the WordNet subset (Miller, 1995) of Cognac as this is the only released setting in Cognac paper, where the topic is a root node and

[9] https://github.com/QwenLM/Qwen2.5-Math/tree/main

the constraint is defined as a subtree. We sampled 200 instances using the provided data generation codes in our experiments. To ensure most model generations ended properly in the decoding process, we relax the constraint of maximum decoded tokens T from 60 to 512. We use the same prompt templates following their Github repo.[10]

Creative Story Generation (Chakrabarty et al., 2024) provides the plots and story continuation from both machine and human. We adopt the provided 11 human-written story plots in the original dataset as the prompt. In this task, we set the maximum token $T = 1024$ to ensure the continued story written by LLM can have a proper ending. We formulate prompt complexity C as providing $C \times 25$ words in the plot.

Random Strings Similar to Bigelow et al. (2024), we sample 200 random strings with length $L \sim U(256, 512)$ from the tokenizer vocabulary as the prompt. Prompt complexity C is formulated by providing $C \times 15$ tokens in the prompt, ensuring each article contains at least 100 tokens.

BBCLatestNews (Li et al., 2024b) is a news collection dataset aims at collecting news that is beyond the time cut for training LLMs. Unlike creative story plots, news articles are typically more structured and organized, although headlines can still be surprising. We select news articles from January to July 2024 to minimize data contamination, as the Llama models have a knowledge cut-off in late 2023. We formulate prompt complexity C as providing $C \times 15$ words in the prompt.

C Entropy Underestimation via Monte Carlo Sampling

To demonstrate the limitations of Monte Carlo (MC) sampling for entropy estimation in long sequences, we conducted an empirical study. We prompted Llama-3-8B-Instruct with 5-shot CoT examples from the MMLU dataset. We then estimated the entropy of its generated responses using a varying number of MC samples: $M \in \{4, 8, 16, 32, 64\}$.

As illustrated in Figure 9, the estimated entropy consistently increases with the number of samples. This trend confirms that MC estimation with a small sample size systematically **underestimates** the true entropy because it fails to capture the long tail of the full probability distribution. While increasing the sample count mitigates this bias, it does so at a significant computational cost. In contrast, AEP Theorem 4.1 allows us to use the negative log-likelihood (NLL) of a single typical sequence for a more efficient and accurate estimation.

D Full BF Output Dynamics Investigation

Here we present full task-wise and model-wise BF output dynamic for Llama-2 in Figure 7 and Llama-3 in Figure 8. We can observe the trends as in § 5.1: ① **The average BF for the base model** (≈ 12) **is roughly ten times higher than the aligned model** (≈ 1.2). ② **BF would often drop smoothly as more output tokens are generated**.

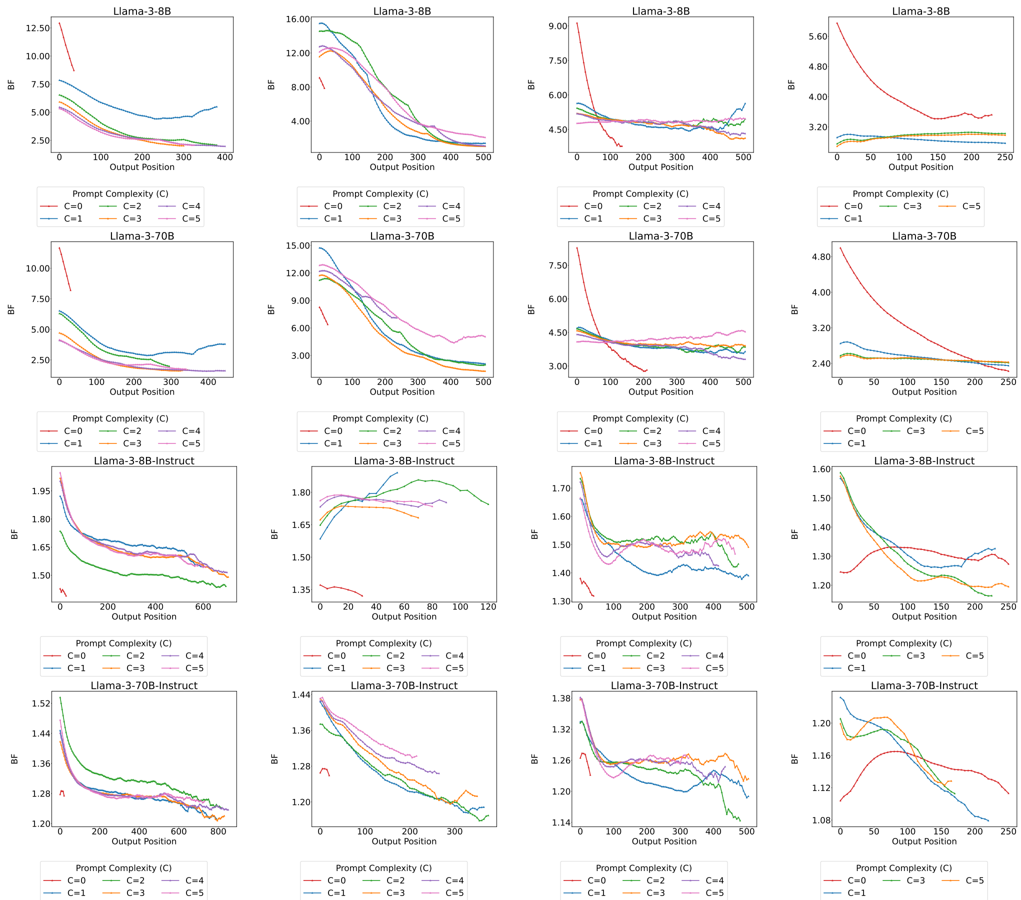
E Curious Case of Prompt Complexity

Intuitively, greater prompt specificity (larger C) reduces BF by narrowing the models output space through more informative context. However, our experimental results reveal task-varied effects. As illustrated in Figure 10 for the Cognac task, greater prompt complexity can *increase* BF–potentially due to the cognitive burden of processing negation or complex linguistic structures. In contrast, for tasks like News Generation, higher C generally leads to lower BF, consistent with the expected narrowing of output diversity. Comprehensive task-wise BF results are provided in Appendix F.

F Full Task-wise BF Evaluation on Different Prompt Complexity

The full task-wise BF evaluation results over different prompt complexity can be found in Figure 11. Here we can see that prompt complexity modulates BF in highly non-consistent ways across models

[10] `https://github.com/princeton-nlp/Cognac/tree/main`



(a) Creative StoryGen (b) Random Strings (c) BBCNewsLatest (d) MMLU

Figure 8: **BF Output Dynamic for Llama-3-families.** For better visualization, we compute the exponential moving averaged values of perplexity with the smoothing factor set as 0.1.

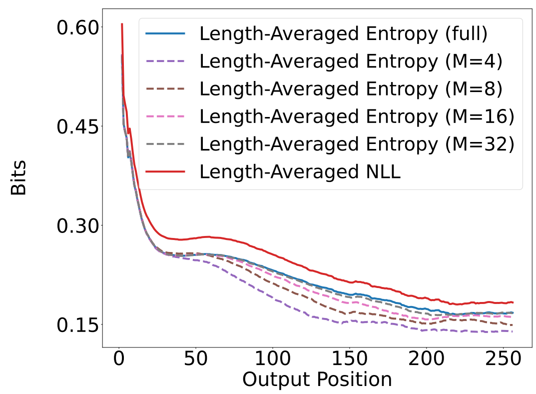
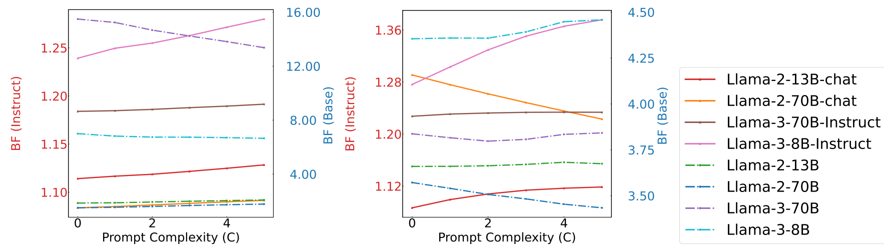


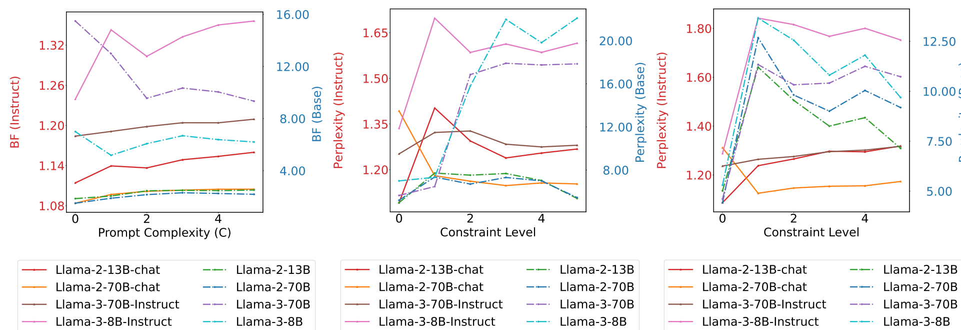
Figure 9: **Monte Carlo (MC) sampling systematically underestimates entropy.** The plot shows that the estimated entropy of sequences from Llama-3-8B-Instruct increases with the number of MC samples (M). A small sample size fails to cover the vast output space, leading to an underestimation of the true entropy. This bias is difficult to eliminate without incurring substantial computational costs.

and tasks, and there are no clear monotonic patterns, contradicting the intuition that with more context given, the model should have more confidence in what to generate.

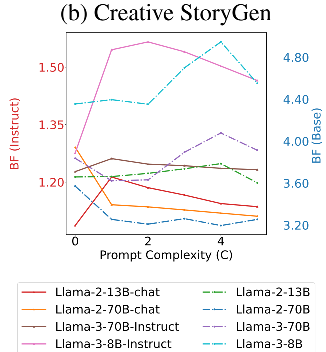


(a) Cognac (b) BBCNewsLatest

Figure 10: **Task-varied influence of prompt complexity C on BF.** On Cognac, we see BF increases with increased C, while on BBCNewsLatest, increasing C can lead to reduced BF.



(a) Cognac (b) Creative StoryGen (c) Random Strings



(d) BBCNewsLatest

Figure 11: BF changes with prompt complexity (C) for Different Tasks. We can see prompt complexity affects BF in a task-varied way.

G Generalization to Additional Tasks and Models

To confirm the generalizability of our findings (§ 5), we extend our experiments to new domains: summarization on XSUM (Narayan et al., 2018), multilingual tasks on Aya (Singh et al., 2024). We formulate prompt complexity C as providing $C \times 25$ words in the prompt. We also verify our findings on a new model, Qwen3-4B (Team, 2025).[11] As presented in Figure 12, our core conclusions remain robust across these diverse conditions.

H Proof of AEP

The following proof is a simplified version of the one in (Mudireddy et al., 2024), presented for completeness and to refine its original bounds. For the formal measure-theoretical treatment, we

[11]For the Qwen3 family, we use the Qwen3-4B-Base and Qwen3-4B-Instruct-2507 pair. Other aligned variants can be activated into a reasoning mode, exhibiting behavior distinct from the models in our main study, and were thus excluded for a fair comparison.

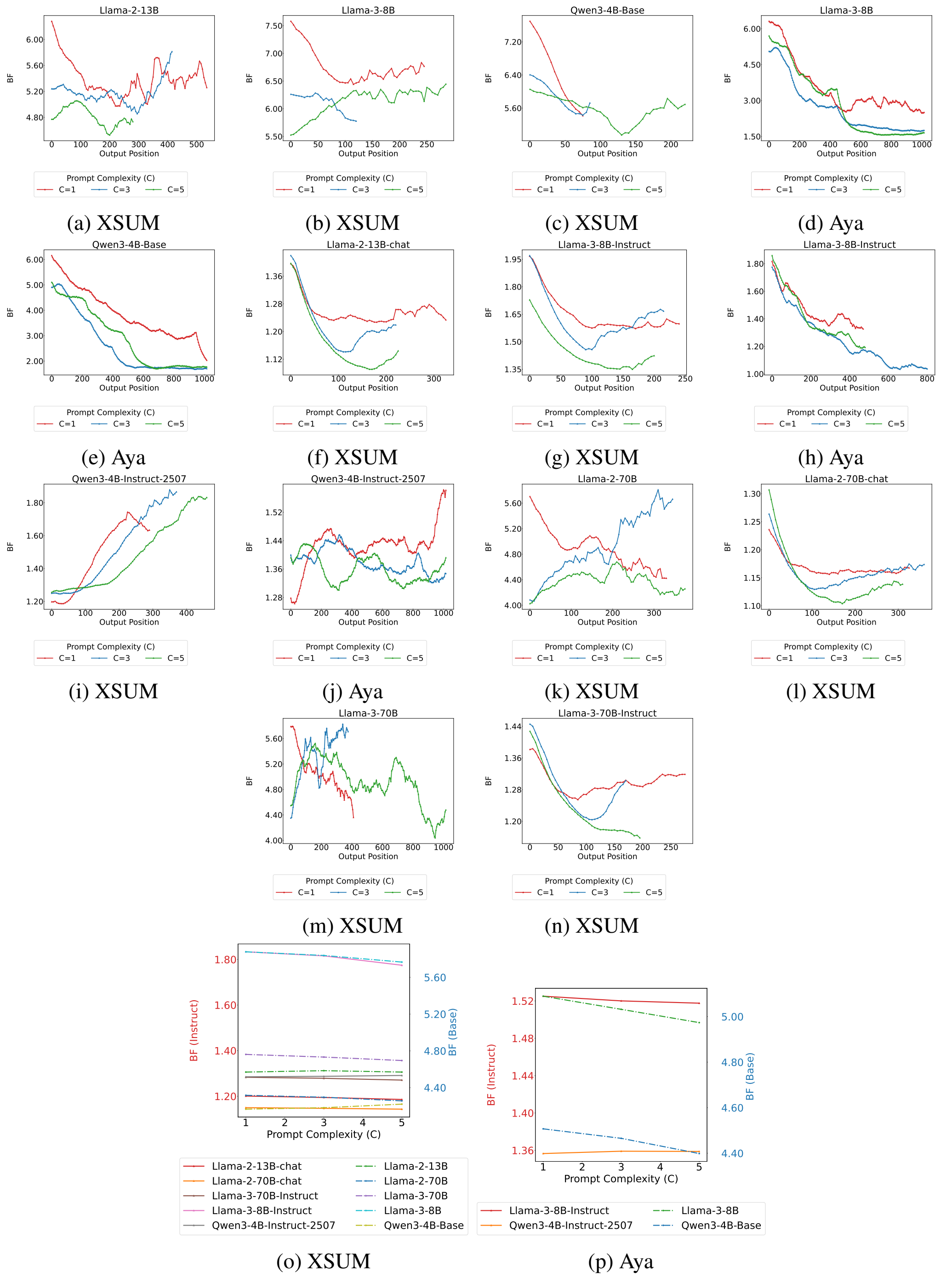


Figure 12: **Additional Verification on Summarization, Multilingual and Qwen Model family.** For better visualization, we compute the exponential moving averaged values of perplexity with the smoothing factor set as 0.1.

refer readers to the original paper. While a more direct proof using the weak law of large numbers is possible, we use Chebyshev's inequality to provide a more self-contained and accessible argument.

The key observation here is that under current computation architecture, the probability implemented by transformers are log-precision (Merrill & Sabharwal, 2023), and thus $|\log P(y_{1:N}|x;\theta)|$ is bounded (e.g., $|\log P(y_{1:N}|x;\theta)| \leq M$). For the truncated probability $\tilde{P}(y_{1:N}|x;\theta)$, we can essentially only consider the non-zero probability over truncated vocabulary and the same thing holds. Depending on the quantization scheme implemented, examples of M include $32, 64$, etc.

Then, note that $\bar{H}\left(\mathrm{Y}_{1:N}|x;\theta\right)$ is actually the first moment of variable $\log P\left(y_{1:N}|x;\theta\right)$ divided by N, if we can further show that the second moment of $\log P\left(y_{1:N}|x;\theta\right)$ is bounded by some $G(N)$ which only increases linearly in N, by Chebyshev Inequality (Cohen, 2015), we can prove:

$$P\left(\left|-\frac{1}{N}\log\tilde{P}\left(y_{1:N}|x;\theta\right)-\bar{H}\left(\mathrm{Y}_{1:N}|x;\theta\right)\right|\geq\epsilon\right)\leq\frac{\mathrm{Var}(\log P\left(y_{1:N}|x;\theta\right))}{N^2\epsilon^2}\leq\frac{G(N)}{N^2\epsilon^2}\tag{6}$$

Then taking $N\to\infty$ on both sides, we will see:

$$P\left(\left|-\frac{1}{N}\log\tilde{P}\left(y_{1:N}|x;\theta\right)-\bar{H}\left(\mathrm{Y}_{1:N}|x;\theta\right)\right|\geq\epsilon\right)\to 0\tag{7}$$

Equivalently:

$$\lim_{N\to\infty}P\left(\left|-\frac{1}{N}\log\tilde{P}\left(y_{1:N}|x;\theta\right)-\bar{H}\left(\mathrm{Y}_{1:N}|x;\theta\right)\right|<\epsilon\right)=1\tag{8}$$

and we can complete the proof.

Now let's prove a specific choice of $G(N)=NM^2$ upper-bounds $\mathrm{Var}\left[\log P\left(y_{1:N}|x;\theta\right)\right]$.

First, we note that

$$\log P\left(y_{1:N}|x;\theta\right)=\sum_{t=1}^{N}\log P\left(y_t|[x;y_{1:t-1}];\theta\right)\tag{9}$$

$$H\left(\mathrm{Y}_{1:N}|x;\theta\right)=\sum_{t=1}^{N}H\left(\mathrm{Y}_t|[x;\mathrm{Y}_{1:t-1}];\theta\right)=\sum_{t=1}^{N}\mathbb{E}_{y_{1:t-1}}\left[H\left(\mathrm{Y}_t|[x,y_{1:t-1}];\theta\right)\right]\tag{10}$$

$$\begin{aligned}\mathrm{Var}\left[\log P\left(y_t|[x,y_{1:t-1}];\theta\right)\right]&=\mathbb{E}\left[\log P\left(y_t|[x,y_{1:t-1}];\theta\right)-H\left(\mathrm{Y}_t|[x;\mathrm{Y}_{1:t-1}];\theta\right)\right]^2\\&=\mathbb{E}[\log^2P\left(y_t|[x,y_{1:t-1}];\theta\right)]-H^2\left(\mathrm{Y}_t|[x;\mathrm{Y}_{1:t-1}];\theta\right)\\&\leq M^2\end{aligned}\tag{11}$$

The last line is because $|\log P\left(y_{1:N}|x;\theta\right)|\leq M$.

Following Mudireddy et al. (2024), we use induction on N to prove $\mathrm{Var}\left[\log P\left(y_{1:N}|x;\theta\right)\right]\leq NM^2$. $N=1$ trivially holds as $\mathrm{Var}\left[\log P\left(y_1|[x];\theta\right)\right]\leq 1\cdot M^2$ (directly set $N=1$ in Equation (11)).

Assuming $\mathrm{Var}\left[\log P\left(y_{1:N}|x;\theta\right)\right]\leq NM^2$ holds for $N=K$, for $N=K+1$, we have:

$$\begin{aligned}\mathrm{Var}\left[\log P\left(y_{1:K+1}|x;\theta\right)\right]&=\mathbb{E}\left[\log P\left(y_{1:K+1}|x;\theta\right)-H\left(\mathrm{Y}_{1:K+1}|x;\theta\right)\right]^2\\&=\mathbb{E}[\log P\left(y_{1:K}|x;\theta\right)-H\left(\mathrm{Y}_{1:K}|x;\theta\right)\\&\quad+\log P\left(y_{K+1}|[x,y_{1:K}];\theta\right)-H(\mathrm{Y}_{K+1}|[x,\mathrm{Y}_{1:K}];\theta)]^2\\&=\mathbb{E}[C(K)]^2+2*\mathbb{E}[C(K)D(K+1)]+\mathbb{E}[D(K+1)]^2\\&\leq KM^2+2*\mathbb{E}[C(K)D(K+1)]+M^2\end{aligned}\tag{12}$$

where:

$$C(K)=\log P\left(y_{1:K}|x;\theta\right)-H\left(\mathrm{Y}_{1:K}|x;\theta\right)\tag{13}$$

$$D(K+1)=\log P\left(y_{K+1}|[x,y_{<K+1}];\theta\right)-H(\mathrm{Y}_{K+1}|[x,\mathrm{Y}_{1:K}];\theta)\tag{14}$$

Using the law of iterated expectation, we can compute $\mathbb{E}[C(K)D(K+1)]$ as:

$$\begin{aligned}\mathbb{E}[C(K)D(K+1)]&=\mathbb{E}_{y_{1:K}}\mathbb{E}[C(K)D(K+1)|_{y_{1:K}}]\\&=\mathbb{E}_{y_{1:K}}\left[C(K)\mathbb{E}[D(K+1)|_{y_{1:K}}]\right]\end{aligned}\tag{15}$$

$$=\mathbb{E}_{y_{1:K}}\left[C(K)\mathbb{E}_{y_{K+1}|y_{1:K}}[\log P\left(y_{K+1}|y_{1:K}x;\theta\right)-H(\mathrm{Y}_{K+1}|[x,y_{1:K}];\theta)]\right]\tag{16}$$

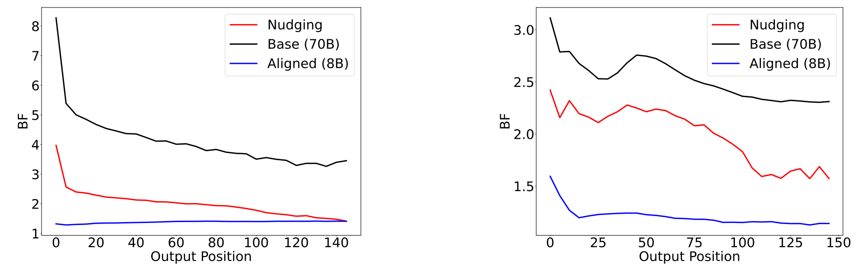
$$=\mathbb{E}_{y_{1:K}}\left[C(K)\cdot 0\right]=0\tag{17}$$

Equation (15) is because $C(K)$ is a deterministic function when $y_{1:K}$ has been realized. Equation (16) is because $H(\mathrm{Y}_{K+1}|y_{1:K})]=E[\log P\left(y_{K+1}|y_{1:K}x;\theta\right)]$.

Plug Equation (17) in Equation (12), we can obtain $\mathrm{Var}\left[\log P\left(y_{1:K+1}|x;\theta\right)\right]\leq(K+1)M^2$, which completes the induction. With this specific choice of $G(N)=NM^2$, we can prove Equation (8).

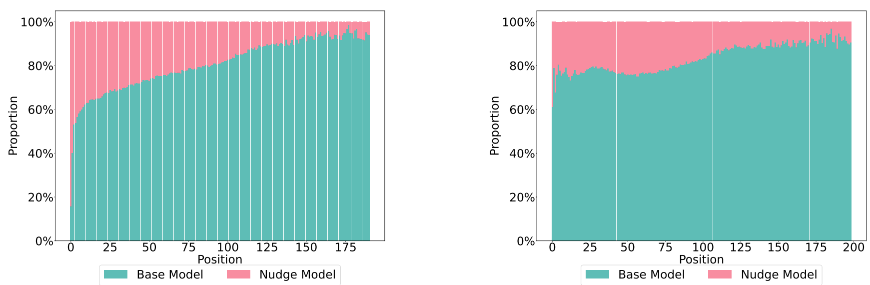
I Full Nudging Experiment Results

Due to space limits, we put the nudging experiment results for MMLU here. Though on MMLU, nudging does not reduce BF that quickly as over Just-Eval-Instruct, it does bring down BF of base models significantly, which verifies our hypothesis in § 7.



(a) Just-Eval-Instruct (b) MMLU

Figure 13: Output Perplexity Dynamics in Nudging Experiments.



(a) Just-Eval-Instruct (b) MMLU

Figure 14: Nudging Ratio Histogram.

J BF and Information Density

Our BF measure can also be interpreted as capturing the information density that LLMs target to facilitate efficient communication (Genzel & Charniak, 2002; Jaeger & Levy, 2006; Levy, 2008; Mahowald et al., 2013; Meister et al., 2021; Verma et al., 2023). Prior work has leveraged both token-level log-probabilities and entropy rates ($\bar{H}$) as proxies for information density in human and machine communication. In Theorem 4.1, we formalize the connection between these views, showing that BF–defined as the exponentiated entropy rate–aligns naturally with this theoretical framework. Unlike prior studies focused primarily on linguistic theory or cognitive science, our work operationalizes this principle at scale across modern LLMs, linking information density to alignment training, decoding dynamics, and output variability in a unified analysis.

K Discussion: Diversity and BF Correlation

Following the branching factor (BF) analysis in § 2, a higher BF suggests greater lexical diversity in finite samples. To examine the relationship between BF and traditional diversity metrics, we compute Distinct-N (Li et al., 2016), incorporating necessary LLM-specific adaptations (Tevet & Berant, 2021; Guo et al., 2024; Kirk et al., 2024). We then conduct a correlation analysis between Distinct-N and BF.

Our results, presented in Figure 15, show **no consistent correlation** between BF and Distinct-N. Depending on the model and task, the relationship can be strongly positive, strongly negative, or entirely absent (e.g., Llama-3-70B-Instruct on Cognac at Figure 15b). This empirical inconsistency highlights a fundamental conceptual point: *BF measures a property of the underlying probability distribution, whereas diversity metrics measure a surface property of finite samples.*

BF, as the exponentiated entropy, characterizes the "width" of the model's entire output distribution. In contrast, metrics like Distinct-N describe a small set of sampled outputs and are known to be unreliable proxies for distributional properties, being sensitive to confounding factors like generation length (Liu et al., 2022).[12] This distinction is critical, as two models can produce samples of similar diversity while having fundamentally different underlying distributions (e.g., with infinite KL-divergence), a nuance that BF captures but sample-based metrics miss. Therefore, our work focuses on probability concentration, measured by BF, as a more fundamental and insightful tool for understanding a model's generative process.

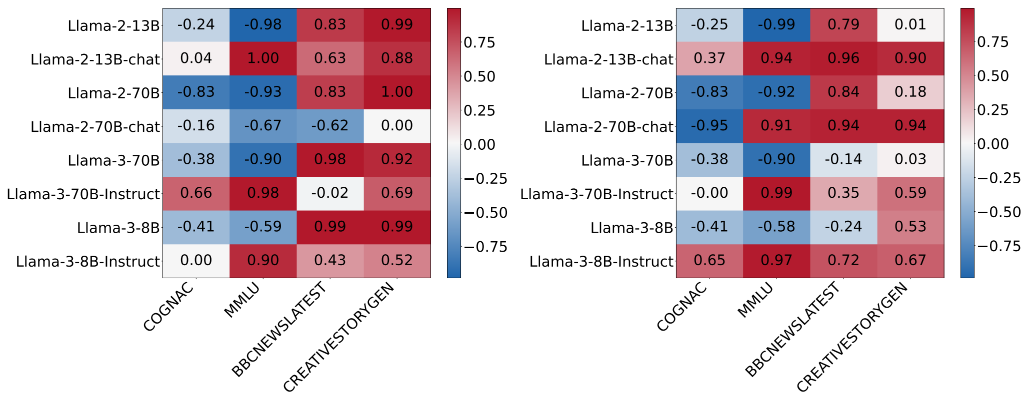
Viewing alignment through the lens of BF reduction provides a unified framework that explains several disparate observations: it clarifies how alignment shrinks the generative horizon, why aligned models are less sensitive to decoding methods, and how techniques like Chain-of-Thought stabilize generation by shifting information to low-BF regions. This focus on distributional properties aligns with emerging research highlighting the importance of a model's entropy in understanding and improving advanced reasoning capabilities Cui et al. (2025); Wu et al. (2025b).

L Confounder Investigation: Data Contamination

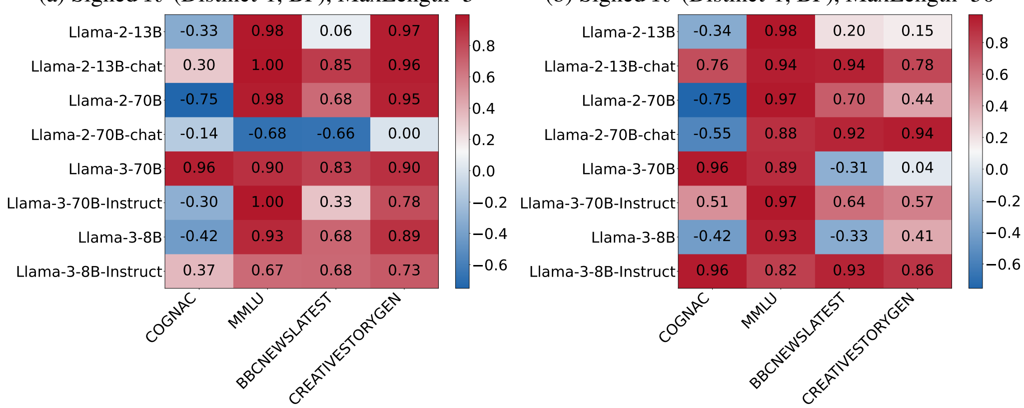
A potential confounder in our analysis is the influence of data contamination. If prompts closely resemble the training data (including pretraining and alignment tuning, i.e., "data contamination"), smaller BF values would be expected, and vice versa. To evaluate this, we use the Min-K$\%$ metric (Shi et al., 2024), which quantifies the overlap between experimental prompts and training data. Following Shi et al. (2024), we set $K = 20$ and compute the average log-likelihood for the minimum $K\%$ of tokens. Using these Min-K$\%$ values, we perform a linear regression with BF to assess their correlation. For each task-model pair, Signed R^2 values are reported to indicate the strength and sign (positive or negative) of the correlation.

The results of the Min-K$\%$ analysis are presented in Figure 16. Significant negative correlations between BF and Min-K$\%$ are observed for models such as Llama-3-8B-Instruct, Llama-3-70B-Instruct, and Llama-2-70B-Chat across several tasks. Conversely, Llama-3-8B and Llama-2-13B-Chat models exhibit positive correlations. For other models, correlations are notably weaker. Overall, there is no consistent correlation pattern between BF and Min-K$\%$ across datasets and models, suggesting that data contamination cannot fully explain our findings.

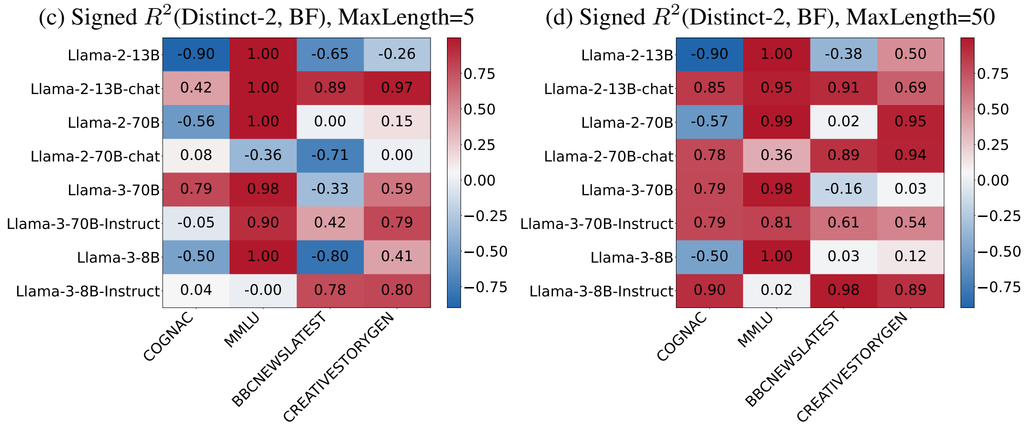
[12] While the EAD metric (Liu et al., 2022) mitigates this issue, it remains influenced by vocabulary size and is not model-agnostic.



(a) Signed R^2(Distinct-1, BF), MaxLength=5 (b) Signed R^2(Distinct-1, BF), MaxLength=50



(c) Signed R^2(Distinct-2, BF), MaxLength=5 (d) Signed R^2(Distinct-2, BF), MaxLength=50



(e) Signed R^2(Distinct-4, BF), MaxLength=5 (f) Signed R^2(Distinct-4, BF), MaxLength=50

Figure 15: Correlational Analysis of BF and Distinct-N. We can find there is no consistent correlation between Distinct-N and BF.

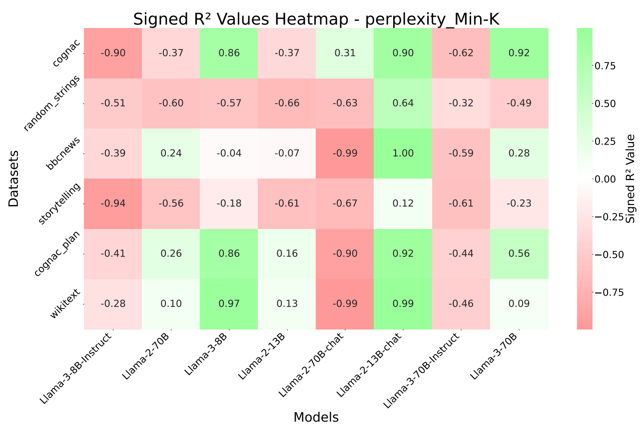


Figure 16: Signed R^2 values heatmap investigating correlation between EBF and Min-K %.