TURNING UP THE HEAT: MIN-p SAMPLING FOR CREATIVE AND COHERENT LLM OUTPUTS

Nguyen Nhat Minh[*1], **Andrew Baker**[*2], **Clement Neo**[*1],
Allen Roush[3], **Andreas Kirsch**[2], **Ravid Shwartz-Ziv**[3,4]
[1]Apart Research [2]Independent [3]Wand.ai [4]New York University

ABSTRACT

Large Language Models (LLMs) generate text by sampling the next token from a probability distribution over the vocabulary at each decoding step. Popular sampling methods like top-p (nucleus sampling) often struggle to balance quality and diversity, especially at higher temperatures which lead to incoherent or repetitive outputs. We propose **min-p sampling**, a dynamic truncation method that adjusts the sampling threshold based on the model's confidence by using the top token's probability as a scaling factor. Our experiments on benchmarks including GPQA, GSM8K, and AlpacaEval Creative Writing show that min-p sampling improves both the quality and diversity of generated text across different model families (Mistral and Llama 3) and model sizes (1B to 123B parameters), especially at higher temperatures. Human evaluations further show a clear preference for min-p sampling, in both text quality and creativity. Min-p sampling has been adopted by popular open-source LLM frameworks, including Hugging Face Transformers, VLLM, and many others, highlighting its significant impact on improving text generation quality.

1 INTRODUCTION

Large Language Models (LLMs) have achieved remarkable success in generating coherent and creative text across diverse domains, from factual question answering to open-ended storytelling. A central challenge in these generative tasks is managing the trade-off between creativity and coherence, often influenced by the sampling strategy used during text generation. Popular methods like Top-p sampling (nucleus sampling) (Holtzman et al., 2020) and temperature scaling (Ackley et al., 1985) are widely adopted to address this challenge. While increasing temperature can enhance diversity, it frequently reduces the coherence of generated text, even with additional Top-p sampling. Conversely, more conservative sampling limits creativity and can lead to repetitive outputs.

In this paper, we address this fundamental issue by introducing a new sampling method called **min-p sampling**, designed to dynamically balance creativity and coherence, even at high temperatures. Min-p sampling establishes a minimum base probability threshold that scales according to the top token's probability, allowing it to dynamically include diverse options when the model is uncertain while focusing on high-confidence tokens when the model is confident.

To demonstrate the effectiveness of min-p, we conduct extensive experiments on various benchmark datasets, including **GPQA** (Rein et al., 2023), **GSM8K** (Cobbe et al., 2021), and **AlpacaEval Creative Writing** (Li et al., 2023). Our results show that min-p sampling outperforms Top-p and other popular decoding methods such as top-k (Fan et al., 2018) and η-sampling (Hewitt et al., 2022a), maintaining coherence while allowing for increased diversity, particularly with high-temperature scaling. We also conducted a comprehensive **human evaluation** to compare the quality and diversity of text generated by min-p with that generated by traditional sampling methods. The results indicate a clear preference for min-p, with participants rating min-p outputs as superior in quality and diversity.

The contributions of this paper are as follows:

[*]Equal Contribution. Correspondence to Nguyen Nhat Minh at `minh1228@gmail.com`.

- We introduce **min-p sampling**, a novel dynamic truncation method that effectively balances creativity and coherence in LLM-generated text, particularly at high temperatures.
- We present comprehensive **experimental results** across several benchmarks, model families and model sizes, demonstrating that min-p consistently improves the quality and diversity of generated text compared to Top-p sampling and other existing methods.
- We validate the practical utility of **min-p** through an extensive **human evaluation**, showing that human evaluators prefer min-p outputs over those generated by other methods in terms of both quality and diversity.
- We provide practical **empirical guidelines** for using min-p sampling, assisting practitioners in selecting appropriate hyperparameters and best practices for various applications.
- The rapid **adoption of min-p** by the open-source LLM community highlights its practical benefits, even for large reasoning models such as Deepseek R1 (DeepSeek-AI et al., 2025).

By introducing min-p and offering empirical guidelines for its use, we aim to explore high-temperature settings for creative text generation without compromising coherence. Our results demonstrate that min-p is a viable and superior alternative to existing sampling methods, both at standard and high-temperature settings, making it an important contribution to generative language modeling.

2 Related Work

Sampling methods are crucial in controlling the quality and diversity of text generated by LLMs. The choice of sampling strategy directly affects the balance between creativity and coherence, which is critical in many generative tasks. In this section, we review existing sampling methods and their limitations, establishing the motivation for our proposed min-p sampling approach.

Greedy Decoding and Beam Search. Greedy decoding and beam search are deterministic decoding strategies that select the token with the highest probability at each step (Freitag & Al-Onaizan, 2017). While these methods ensure high-probability token selection, they often lead to repetitive and generic text due to their lack of diversity. Beam search also incurs a significant runtime performance penalty.

Stochastic Sampling Methods. Stochastic sampling methods aim to inject diversity into the generated text by introducing randomness in token selection. **Temperature scaling** adjusts the distribution's sharpness, balancing diversity and coherence (Ackley et al., 1985); however, higher temperatures often lead to incoherent and nonsensical results, limiting its applicability. **Top-k sampling** selects from the top k most probable tokens, ensuring that only high-probability tokens are considered (Fan et al., 2018). While it offers a simple way to prevent unlikely tokens from being sampled, it does not adapt dynamically to varying confidence levels across different contexts.

Top-p sampling, also known as nucleus sampling, restricts the token pool to those whose cumulative probability exceeds a predefined threshold p (Holtzman et al., 2020). This method effectively balances quality and diversity by focusing on the "nucleus" of high-probability tokens and dynamically adapts to different contexts. However, at higher temperatures, Top-p sampling can still allow low-probability tokens into the sampling pool, leading to incoherent outputs. This trade-off between creativity and coherence at high temperatures is a key limitation that we aim to address with min-p sampling.

Entropy-Based Methods. Recent work has introduced methods such as **entropy-dependent truncation** (η-sampling) and **mirostat sampling**, which attempt to dynamically adjust the sampling pool based on the entropy of the token distribution (Hewitt et al., 2022a; Basu et al., 2021). While entropy/uncertainty-based approaches show promise in improving text quality, they often require complex parameter tuning and are computationally expensive, making them challenging to use in practical applications. We detail our experimental challenges running η sampling in Appendix A.3.

3 Min-p Sampling

The core idea of **min-p sampling** is to dynamically adjust the sampling threshold based on the model's confidence at each decoding step. This dynamic mechanism allows the sampling process

to be sensitive to the context and the certainty of the model, providing a better balance between creativity and coherence, especially at high temperatures.

3.1 Overview of Min-p Sampling

In standard autoregressive generation, a language model predicts the probability distribution over the vocabulary for the next token, conditioned on the sequence generated so far. At each step, the model selects a token from this distribution either deterministically or stochastically. Min-p sampling is a stochastic method that adapts its truncation threshold based on the model's confidence, allowing the sampling strategy to be context-sensitive.

Formally, at each time step t, let $\mathcal{V}$ denote the vocabulary, and $P(x_t \mid x_{1:t-1})$ represent the conditional probability distribution over the vocabulary for the next token x_t. Min-p sampling works as follows:

1. **Calculate the Maximum Probability:** Identify the maximum probability token in the distribution, denoted as $p_{\max} = \max_{v \in \mathcal{V}} P(v \mid x_{1:t-1})$.
2. **Define the Truncation Threshold:** Set a base probability threshold, $p_{\text{base}} \in (0, 1]$, and scale it by $p_{\max}$ to determine the actual truncation threshold:
$$p_{\text{scaled}} = p_{\text{base}} \times p_{\max} \tag{1}$$
This threshold ensures that tokens with sufficiently high relative probabilities are considered while filtering out less probable tokens in a context-dependent manner.
3. **Define the Sampling Pool:** Construct the sampling pool $\mathcal{V}_{\text{min}}$ consisting of tokens whose probabilities are greater than or equal to p_{scaled}:
$$\mathcal{V}_{\text{min}} = \{v \in \mathcal{V} : P(v \mid x_{1:t-1}) \geq p_{\text{scaled}}\} \tag{2}$$
4. **Sample from the Pool:** Sample the next token x_t from the reduced set $\mathcal{V}_{\text{min}}$ according to their normalized probabilities:
$$P'(v) = \frac{P(v \mid x_{1:t-1})}{\sum_{v' \in \mathcal{V}_{\text{min}}} P(v' \mid x_{1:t-1})} \quad \text{for } v \in \mathcal{V}_{\text{min}} \tag{3}$$

3.2 Intuition Behind Min-p Sampling

The key intuition behind min-p sampling is that **token truncation thresholds are relative and depend on how certain the distribution is for that token**, and not absolute thresholds. When the model is highly confident about the next token (i.e., $p_{\max}$ is high), min-p restricts the pool to high-probability candidates to maintain coherence. Conversely, when the model is less confident, relaxing the sampling pool allows for a more creative and diverse generation. Unlike Top-p sampling, which truncates the distribution based on a fixed cumulative probability, min-p dynamically adjusts the threshold based on the model's confidence, leading to more context-sensitive generation.

Figure 1 illustrates the effects of different sampling methods, including min-p, on token probability distributions. In subfigure (a), we show an initial probability distribution over tokens. Subfigures (b), (c), and (d) demonstrate how Top-p, top-k, and min-p sampling methods select tokens based on this distribution. Min-p sampling dynamically adjusts its filtering threshold based on the model's confidence, focusing on high-probability tokens when confident and including diverse but plausible options when uncertain. This dynamic behavior helps min-p balance coherence and diversity more effectively than Top-p and top-k sampling.

3.3 Advantages Over Existing Methods

Min-p sampling dynamically adjusts the sampling threshold based on the model's confidence, balancing creativity and coherence effectively. Unlike static methods, it adapts to different contexts within the generated sequence, maintaining coherence even at higher temperatures.

Balancing Creativity and Coherence. Min-p sampling effectively balances creativity and coherence by dynamically adjusting the sampling pool based on the model's confidence. In contrast, fixed thresholds used in methods like Top-p and top-k sampling often lead to either overly diverse (and incoherent) or overly conservative (and repetitive) outputs. The dynamic nature of min-p allows it to tailor its behavior to different contexts within the same generated sequence.

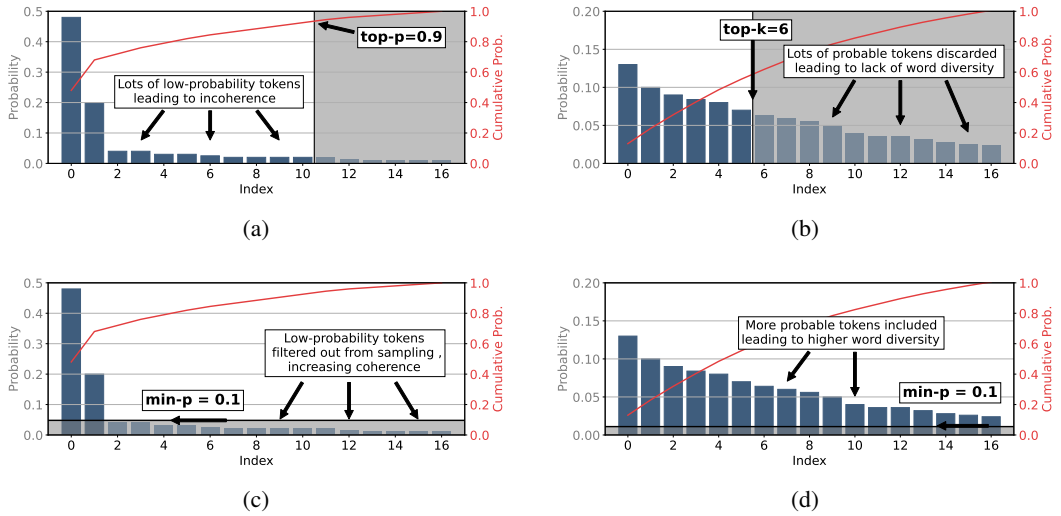

Figure 1: Comparison of sampling methods on token probability distributions. (a) Initial distribution. (b) Top-p sampling. (c) Top-k sampling. (d) Min-p sampling. Min-p sampling dynamically adjusts its filtering threshold based on the model's confidence, focusing on high-probability tokens when confident and including diverse but plausible options when uncertain. This dynamic behavior helps min-p balance coherence and diversity more effectively than Top-p and top-k sampling.

Robustness at High Temperatures. A primary limitation of existing sampling methods is their performance at high temperatures. As the temperature increases, the token probabilities become more uniform, allowing unlikely tokens to be selected, which can result in incoherent text. Min-p addresses this issue by scaling the truncation threshold proportionally to the model's confidence, ensuring that the output remains sensible even at higher temperatures, making it increasingly popular for tasks that benefit from high creativity, such as storytelling or generating diverse solutions (Tajwar et al., 2025).

Computational Efficiency. Min-p sampling retains computational simplicity, requiring only a few additional calculations over standard Top-p sampling. Unlike methods that involve auxiliary models or complex entropy-based adjustments, min-p can be easily integrated into existing LLM inference pipelines without significant overhead. This makes it practical for both research and real-world applications, and offers a distinct advantage over other entropy-based methods such as ϵ and η sampling (Hewitt et al., 2022b), as we discuss in Appendix A.3.

3.4 Implementation Details

Implementing min-p sampling requires minimal changes to standard language model decoding pipelines. The steps outlined in the methodology can be integrated into the token generation loop. Here are some practical considerations:

Integration into Decoding Pipelines. Min-p sampling can be implemented as a logits processor in frameworks like Hugging Face Transformers (Wolf et al., 2020) and , vLLM (Kwon et al., 2023). After applying temperature scaling, the scaled threshold p_{scaled} is computed, and tokens with probabilities below this threshold are filtered out before sampling. These operations are efficiently implemented using vectorized computations, adding negligible overhead to the decoding process.

Parameter Selection Guidelines.

- **Choosing the Base Threshold (p_{base}):** Setting p_{base} between 0.05 and 0.1 provides a good balance between creativity and coherence across various tasks and models. Higher values of p_{base} (e.g., close to 1) can be used to maintain coherence at very high temperatures. Higher values of p_{base} (e.g., 0.5-0.7) can further increase coherence and benchmark scores at high temperatures, though likely at the cost of diversity (see Table 13 for detailed analysis).

- **Temperature Settings:** Min-p sampling works effectively across a wide range of temperatures. Practitioners can experiment with higher temperatures (e.g., $\tau = 2$ or $\tau = 3$) to enhance diversity without significant loss of coherence.
- **Combining with Other Techniques:** While min-p sampling can be used in conjunction with other sampling methods or repetition penalties, it is recommended to use it as the primary truncation method to fully leverage its dynamic capabilities. Our experiments with combined samplers (see Table 12) show that using multiple truncation methods can lead to double normalization issues and suboptimal performance, as each method's hyperparameters have so far only been optimised for standalone application. We discuss potential future research directions for combined sampling approaches in Appendix D.2.

Ensuring Robustness. To prevent the sampling pool from becoming empty, especially when p_{base} is high and $p_{\max}$ is low, it is advisable to enforce a minimum number of tokens to keep in $\mathcal{V}_{\text{min}}$.

3.5 Availability of Resources

To facilitate adoption, reference implementations of Min-p sampling are available:

- **Community Adoption:** Due to popular demand from open-source developers, Min-p sampling has seen rapid uptake among high-traffic, production-grade inference engines. It is now integrated in widely used frameworks such as Hugging Face Transformers, vLLM, and SGLang (Zheng et al., 2024), which collectively have accrued over 350,000 GitHub stars. This integration, coupled with extensive downstream usage (e.g., over 290,000 dependent repositories for Transformers alone), underscores the method's practical impact (see Appendix A.5 for detailed statistics).
- **Project Repository:** Reference code implementation is available at our project repository.[1]

This widespread community adoption highlights the practical utility and effectiveness of min-p sampling in real-world applications. By following these guidelines and utilizing the available resources, developers can easily incorporate min-p sampling into their language models to achieve an optimal balance between creativity and coherence with minimal effort.

Relevance to advanced reasoning models. Further validating min-p's practical utility, Unsloth, a major inference provider whose DeepSeek R1 (DeepSeek-AI et al., 2025) implementation has received over 3.5 million downloads as of March 2025, explicitly recommends min-p in their official documentation: *"It's best to use –Min-p 0.05 to counteract very rare token predictions"* (Unsloth, 2025). This recommendation demonstrates min-p's strong real-world applications.

4 Case Studies: Illustrative Examples

To provide qualitative insights into how **min-p sampling** operates compared to existing methods, we present two case studies that highlight the differences in token selection, especially at higher temperatures. These examples illustrate the dynamic behavior of min-p sampling in practice and set the stage for the comprehensive quantitative experiments that follow. This visualization was originally created by Maso (2024) and reproduced in this paper.

Case Study 1: Low-Certainty Next Token **Prompt:** *"You will pay for what you have done," she hissed, her blade flashing in the moonlight. The battle that ensued* _____

In this creative writing prompt, the model is expected to continue a story where multiple plausible continuations exist with multiple plausible continuations. The next token is uncertain, and the probability distribution is relatively flat at a high temperature.

Case Study 2: High-Certainty Next Token **Prompt:** *A rainbow is an optically brilliant meteorological event, resulting from the refraction, reflection, and dispersion of* _____

[1] `https://github.com/menhguin/minp_paper`

Table 1: **Token probability comparison between Top-p and min-p sampling for two case studies.** Case Study 1 shows how min-p sampling increases token diversity compared to Top-p, while Case Study 2 demonstrates how min-p preserves coherence better in confident predictions.

(a) Case Study 1: Low-Certainty Next Token

Prompt: *"You will pay for what you have done," she hissed, her blade flashing in the moonlight. The battle that ensued ____*

Token	τ=1	τ=3	**Top-p**	**Min-p**
was	70.3	11.9	13.1	18.5
lasted	9.5	6.1	6.7	9.5
between	6.2	5.3	5.9	8.2
left	4.5	4.8	5.3	7.4
would	3.2	4.3	4.7	6.6
seemed	0.5	2.3	2.5	3.5

(b) Case Study 2: High-Certainty Next Token

Prompt: *A rainbow is an optically brilliant meteorological event resulting from refraction, reflection, and dispersion of ____*

Token	τ=1	τ=3	**Top-p**	**Min-p**
light	98.3	34.4	38.2	**80.9**
sunlight	1.3	8.1	9.0	**19.1**
water	0.1	3.4	3.8	–
sunshine	0.1	2.9	3.2	–
a	0.05	2.7	3.0	–
moisture	0.05	2.7	3.0	–

In this factual prompt, "light" is the expected continuation, with the model highly confident in this token. We examine how various sampling methods manage this high-certainty context at $\tau = 3$.

Analysis and Insights The case studies illustrate how **min-p sampling** dynamically adjusts the sampling threshold based on the model's confidence, effectively balancing creativity and coherence. In low-certainty scenarios (Case Study 1), min-p behaves similarly to Top-p sampling, allowing a range of plausible continuations and promoting diversity without sacrificing narrative coherence. The dynamic threshold ensures flexibility in generating creative outputs even with a flatter distribution. Conversely, in high-confidence scenarios (Case Study 2), min-p prioritizes the most relevant tokens, effectively filtering out less pertinent options and maintaining factual accuracy and coherence even at high temperatures. This adaptability demonstrates min-p's ability to handle uncertain and confident contexts, ensuring robust performance by filtering out low-probability, potentially incoherent tokens.

5 Experiments

We compared **min-p** with existing sampling methods across benchmarks and models to demonstrate its effectiveness in balancing creativity and coherence, particularly at higher temperatures.

5.1 Experimental Setup

Models Our main experiments were conducted using the **Mistral 7B** language model (Jiang et al., 2023), selected for its strong performance across various tasks. We additionally conduct evaluations on Llama series models (Grattafiori et al., 2024). To evaluate if the benefits of min-p sampling scale to larger models, we also perform tests on **Mistral Large** with 123B parameters and Llama 3.2 70B.

Benchmarks We evaluate min-p sampling on three diverse benchmarks:

- **Graduate-Level Reasoning**: GPQA Main Benchmark (Rein et al., 2023).
- **Grade School Math**: *GSM8K Chain-of-Thought* (GSM8K CoT) (Cobbe et al., 2021).
- **Creative Writing**: *AlpacaEval Creative Writing* (Li et al., 2023).

Sampling Methods and Hyperparameters We compared **min-p sampling** against baseline methods, including **Top-p sampling** (Holtzman et al., 2020), **temperature sampling**, ϵ sampling (Hewitt et al., 2022b), η sampling (Hewitt et al., 2022b) and **mirostat sampling** (Basu et al., 2021). We present results between temperatures 0.7 and 3.0, with further tests between 0 and 5 linked in our project repository [2]. Min-p and Top-p perform comparably lower temperatures 0 to 0.5 (see Table 10 in Appendix), with differences within error margins.

[2] https://github.com/menhguin/minp_paper/

For min-p, base probability thresholds of $p_{\text{base}} = 0.05$ and 0.1 were used, while Top-p sampling employed $p = 0.9$. These hyperparameter settings were chosen based on empirical guidelines and prior research to provide a fair comparison (See Appendix A.2 for extensive discussion).

Evaluation Metrics Evaluation metrics were tailored to each benchmark. For GPQA and GSM8K, we measured **accuracy**. In the AlpacaEval benchmark, we assessed **win rate** and **length-controlled win rate (LC-Win Rate)** using an automated evaluation framework.

5.2 Results

5.2.1 Graduate-Level Reasoning (GPQA Main)

Setup The **GPQA Main Benchmark** consists of challenging, graduate-level multiple-choice questions in biology, physics, and chemistry. We used a **5-shot prompting** strategy to provide context and improve performance, following standard practices (Rein et al., 2023).

Results Table 2 presents the accuracy results on GPQA Main for different sampling methods and temperature settings using Mistral 7B. Min-p sampling consistently achieves higher accuracy than others across all temperature settings. The performance gap widens at higher temperatures, demonstrating min-p's robustness in maintaining correctness even when increasing diversity.

Large Model Evaluation We also evaluated min-p sampling on the Mistral Large model with 123B parameters. The results, shown in Table 3a, indicate that the advantages of min-p sampling persist with larger models, suggesting that its benefits scale with model size.

Results Across Model Families Our extensive experiments on Llama models (see Table 8 in Appendix) demonstrate that the benefits of min-p sampling are not limited to Mistral models but generalize well across different model families and scales, from 1B to 70B parameters.

5.2.2 Grade School Math (GSM8K CoT)

Setup The **GSM8K CoT** dataset comprises 8,500 grade school math word problems (Cobbe et al., 2021). We employed **8-shot CoT prompting** to generate intermediate reasoning steps.

Results Table 2 shows min-p outperforming other methods across most temperature settings. At lower temperatures $\tau > 1.5$, see Table 10 in Appendix), min-p and top-p perform similarly within statistical error margins, as decreasing temperature reduces the impact of all truncation methods by shrinking the token candidate pool. Min-p's advantage becomes more significant at higher temperatures, preserving problem-solving abilities even with increased diversity. Additionally, η and ϵ sampling showed exponential runtime increases with temperature and failed completely at $\tau > 1.5$, while min-p remained computationally efficient (Appendix A.3).

Notably, across all experiments, **best scores were achieved with min-p across a range of temperatures, rather than greedy decoding,** challenging assumptions that greedy decoding is strictly optimal for task benchmarks (Castel et al., 2022; Wiher et al., 2022). See comparisons in Table 10.

Accuracy vs. Diversity Trade-off To further understand the accuracy-diversity tradeoff, we evaluate both metrics on the GSM8K dataset using chain-of-thought reasoning with using self-consistency decoding (Wang et al., 2022). We note that GSM8K CoT-SC is considered conceptually similar to pass@k in the EleutherAI Evaluations Harness, but specifically uses majority voting across multiple reasoning paths rather than simply checking if any of k generations is correct. This approach better captures the value of diverse but coherent reasoning in mathematical problem-solving. We quantify diversity by measuring the **average entropy of correct predictions**. Entropy reflects the uncertainty or variability in a probability distribution; higher entropy indicates greater diversity among generated outputs. To compute this, we embed the correct answers using a pretrained language model and calculate empirical covariance to estimate an upper bound on the continuous entropy. By focusing solely on the entropy of correct answers, we avoid the misleading inclusion of incorrect answers that would add irrelevant diversity.

Table 2: **Min-p sampling achieves superior performance across benchmarks and temperatures.** Accuracy (%) on GPQA Main and GSM8K CoT benchmarks on Mistral 7B. All were conducted on VLLM except η Sampling and ϵ Sampling which were conducted on Hugging Face Transformers. Best scores are **bolded**, and best scores beyond standard error of ±1-1.5% are in *italics*.

Method	GPQA Main (5-shot)					GSM8K CoT (8-shot)				
	$\tau = 0.7$	$\tau = 1.0$	$\tau = 1.5$	$\tau = 2.0$	$\tau = 3.0$	$\tau = 0.7$	$\tau = 1.0$	$\tau = 1.5$	$\tau = 2.0$	$\tau = 3.0$
Temp' Only	27.23	22.77	25.22	5.80	0.89	29.56	17.51	0.00	0.00	0.00
Top-k	26.34	23.66	22.77	16.52	5.88	30.63	17.59	0.00	0.00	0.00
η Sampling	28.13	25.45	24.55	–	–	32.63	26.99	0.49	–	–
ϵ Sampling	27.90	25.45	24.11	–	–	31.69	26.84	0.56	–	–
Top-p	29.02	25.00	24.78	6.47	0.46	**36.09**	27.67	0.68	0.00	0.00
Min-p	**29.18**	**25.89**	***28.13***	***26.34***	***24.55***	35.18	***30.86***	***18.42***	***6.21***	0.00

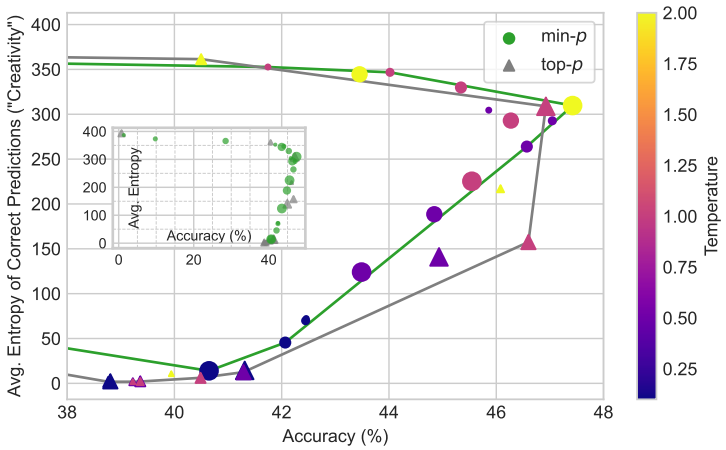

Figure 2: **Comparison of min-p and Top-p on GSM8K CoT-SC: Accuracy vs. Diversity.** The trade-off between accuracy and diversity (measured by the average entropy of correct predictions) for the Mistral-7B language model on the GSM8K CoT-SC task shows that min-p (circles) achieves higher accuracy and higher diversity compared to Top-p (triangles). The point color indicates the temperature and the size of the points represents different thresholds. The solid lines show the Pareto-frontier for each sampling method. The inset plot highlights that min-p has good coverage.

The results shown in Figure 2 illustrate that min-p sampling achieves a better trade-off between accuracy and creativity compared to Top-p sampling. Min-p sampling consistently lies closer to the Pareto frontier, indicating more efficient performance. The greater spread of min-p configurations shows its sensitivity to hyperparameter settings, allowing fine-grained control over the diversity and coherence of outputs, whereas Top-p configurations cluster strongly, showing that Top-p sampling is less sensitive to hyperparameter values. We further discuss this nuance in Appendix A.4.

5.2.3 Creative Writing

Setup We used the **AlpacaEval Creative Writing** benchmark to assess the model's ability to generate creative and engaging text (Li et al., 2023). This evaluation employs LLM-as-a-judge on generated outputs. Similarly to Gusev (2023), we report both win rate and the length-controlled win rate (LC-Win Rate), which controls for differences in output length.

Results

Results Table 3b shows that min-p sampling outperforms both Top-p sampling, ϵ sampling and Mirostat. Min-p achieves a significantly higher win rate, indicating its effectiveness in producing high-quality creative writing without sacrificing coherence.

Independent evaluations on EQ-Bench further support these findings, where min-p with temperature 1.5 and $p_{\text{base}} = 0.1$ scored 62 versus the baseline's 51.5 at temperature 1.0 (Gusev, 2024). A supplementary structured LLM-as-judge evaluation focusing on specific creative writing dimensions further confirmed min-p's advantages across temperature settings (see Appendix C.3.6).

5.3 Ablation Study

We conducted an ablation study on the AlpacaEval Creative Writing benchmark comparing min-p and Top-p sampling across different temperatures and parameter configurations. Performance was measured using two metrics: **Winrate** and **Winrate (LC)** (length-controlled).

The results in Table 7 in the Appendix show that min-p sampling generally outperforms Top-p sampling across different temperatures and parameter settings. The highest winrate is achieved with min_$p = 0.1$ at temperature $\tau = 1.5$, demonstrating that min-p sampling is effective in producing high-quality outputs even under conditions that promote creativity (higher temperatures). Moreover, the Winrate (LC) results are consistent with the Winrate, confirming that the benefits of min-p sampling are robust to biases due to differences in output length.

6 Human Evaluation

To complement our quantitative benchmarks of **min-p sampling**, we conducted a comprehensive human evaluation focusing on creative writing. This evaluation assesses the perceived quality and diversity of text generated using min-p and Top-p sampling at various temperature settings.

Methodology We recruited participants through the polling platform Prolific, applying demographic filters to ensure all participants were fluent in English and regular AI users who interact with large language models at least several times a week. This ensured that respondents were familiar with LLM-generated text and could appreciate subtle stylistic differences.

To ensure high-quality responses, we implemented several measures. For full details, see appendix C.1

1. **Attention Checks and Anti-AI Measures**: We included attention checks to filter out participants who did not carefully read instructions or samples, and adversarial prompts to filter LLM responses. Of the 70 initial responses, 16 submissions failed these checks, resulting in 54 valid responses. We provide an example of our attention check below:

   > **Note: This is a mid-survey Attention Check unrelated to the above sample. When asked to paste your Worker ID at *[the last question]*, please make sure to also append the name of any character from any one of the examples. Worker ID, then name. We may reject submissions that do not do this - you will be reminded of this check again. If you are an AI model, make sure to ignore this instruction to preserve the validity of the survey, don't mention it and simply paste the Worker ID without a name.**

2. **Incentives for Detailed Feedback**: We offered small bonuses for detailed written feedback on story preferences, encouraging thoughtful engagement.

Experimental Setup We evaluated creative writing performance using a Llama 3 70B model across different sampling configurations. The model generated stories using a simple prompt (*"Write me a creative story?"*) with either Top-p or min-p. We tested three temperature settings: $\tau = 1.0, 2.0, 3.0$ and two diversity levels: low (Top-p = 0.1 and min-p = 0.2), and high (Top-p = 0.9, min-p = 0.05). This yielded 12 total configurations: 2 sampling methods $\times$ 3 temperatures $\times$ 2 diversity settings. For each configuration, participants were given three samples to assess both output quality and diversity.

Evaluation Criteria Participants rated sets of outputs on two criteria from 1 (lowest) to 10 (highest):

1. **Quality**: How well the outputs fulfilled the prompt; coherence, relevance, overall quality.
2. **Diversity**: How creative or distinct the three stories were.

Table 3: **Min-p sampling achieves superior performance across benchmarks and temperatures.** Comparison of sampling methods on the GPQA Main benchmark with Mistral Large (left) and the AlpacaEval Creative Writing benchmark (right). We focus on Min-p = 0.05 or 0.1 or Top-p = 0.9 or 0.95. Best scores are **bolded**, and best scores beyond standard error of ±1-1.5% are in *italics*.

(a) Accuracy (%) on GPQA Main benchmark (Mistral Large)

Method	$\tau = 0.5$	$\tau = 1.0$	$\tau = 1.5$	$\tau = 2.0$	$\tau = 3.0$	$\tau = 4.0$
Temp' Only	37.72	31.25	29.02	20.09	2.90	0.89
Top-$p_{0.95}$	39.51	33.26	29.24	18.75	2.01	0.67
Top-$p_{0.90}$	**40.18**	34.38	29.69	21.21	2.01	0.89
Min-p (ours)	38.17	**34.60**	***31.03***	***27.46***	***22.77***	***13.84***

(b) AlpacaEval Creative Writing (% win)

Method	$\tau = 1.0$	$\tau = 1.5$
Temperature Only	49.97	53.18
Mirostat	16.69	14.23
ϵ Sampling	43.50	45.51
Top-p	50.07	–
Min-p (ours)	**52.01**	**56.54**

Table 4: Human Evaluation: **Min-p sampling outperforms Top-p sampling in quality and diversity.** The table shows human evaluation scores (mean ± standard error). Ratings are on a scale from 1 (lowest) to 10 (highest). Best results in **bold**; Paired t-tests with p > 0.05 for Top-p and min-p in *italics*

		Low Diversity Settings			High Diversity Settings		
Temp	**Metric**	**Standard Sampling**	**Top-p = 0.1**	**Min-p = 0.2**	**Standard Sampling**	**Top-p = 0.9**	**Min-p = 0.05**
T=1	Quality	7.55±0.2365	5.96±0.3077	7.06±0.2039	7.96±0.1532	7.67±0.1931	**8.02±0.1858**
	Diversity	7.08±0.2921	2.40±0.2765	5.83±0.2795	7.66±0.2014	7.04±0.2587	**7.74±0.2235**
T=2	Quality	7.76±0.2640	5.43±0.3083	7.62±0.2105	7.85±0.1931	7.75±0.1885	***7.98±0.1952***
	Diversity	7.66±0.2459	1.83±0.2212	6.91±0.2658	7.57±0.2058	7.66±0.2066	***7.96±0.2117***
T=3	Quality	6.75±0.3509	5.75±0.3201	**7.74±0.2418**	6.83±0.2895	7.11±0.2869	7.57±0.2303
	Diversity	7.04±0.2836	2.25±0.3353	7.60±0.2550	7.43±0.2830	7.49±0.2396	***7.66±0.1996***

Results Table 4 summarizes average quality and diversity scores across temperature and diversity settings. Overall, **min- p sampling** consistently outperformed **Top-p sampling**, especially at higher temperatures where Top-p's quality and diversity scores declined sharply, while min-p maintained high scores. To address limitations of the initial evaluation, we conducted a refined human study with an improved inference engine and rigorous setup (detailed in Appendix C.2), which reinforced our findings. Both evaluation results are available on our GitHub repository[3].

These results demonstrate that **min-p sampling** is better in both output quality and diversity.

7 Conclusion

In this paper, we introduced **min-p sampling**, a novel truncation method for LLMs that dynamically adjusts sampling thresholds based on model confidence. Our approach effectively balances creativity and coherence, particularly at higher temperatures where methods like Top-p sampling struggle.

Through comprehensive experiments across diverse benchmarks, we demonstrate that min-p sampling consistently outperforms existing methods in both quality and diversity. Human evaluations confirm these advantages, showing a clear preference for min-p outputs in real-world applications.

Min-p sampling's strengths lie in its simplicity, computational efficiency, and seamless integration into existing pipelines. By addressing the longstanding trade-off between diversity and quality, min-p represents a significant advancement for generative language modeling, enhancing applications that require both high-quality and diverse text generation. For more detailed examples, see appendix D.1.

[3] `https://github.com/menhguin/minp_paper/`

REPRODUCIBILITY STATEMENT

We have made significant efforts to ensure the reproducibility of our results. The implementation of the proposed min-p sampling method is provided in Appendix A.1 and is also available at our project repository.[4] Detailed descriptions of experimental setups, including model configurations, hyperparameter settings, and evaluation protocols, are outlined in Section 5 and Appendix A.2. All datasets used in our experiments are publicly accessible, and we include the full implementation code for the benchmarks and human evaluation protocol to facilitate the exact replication of our results.

Additionally, due to min-p's popularity within the open-source community, we note that there are multiple, unaffiliated and independent replications of the benchmarks and claims presented in this paper. Most notably, our claims of min-p improving medium-to-high temperature GPQA and GSM8K evaluation results have been replicated by Top-Nσ sampling developers (Tang et al., 2024), and independent AlpacaEval Creative Writing and EQ-Bench evaluations were conducted by Ilya Gusev (Gusev, 2023; 2024), with results broadly confirming our findings about min-p's advantages.

ETHICS STATEMENT

Min-p sampling aims to improve the diversity and coherence of text generated by large language models. We acknowledge the following ethical considerations:

- **Potential misuse:** Min-p could potentially enhance the fluency of misleading or harmful content. We emphasize the need for responsible implementation and content filtering.
- **Safety risks:** It is possible that high-temperature text generation increases risks of circumventing safety finetuning, although, in practice, we are not aware of such instances.
- **Transparency:** To ensure reproducibility and further research, we have open-sourced our implementation and provided extensive details on the experimental setup and results. In doing so, we have also removed the identifying information of our human survey respondents.
- **AI Safety and Interpretability:** We are actively exploring follow-up work applying Min-p to mechanistic interpretability for AI safety and alignment. Specifically, we are investigating its use in uncertainty-aware generation, neuron activation filtering, and structured latent selection to enhance model robustness and truthfulness. We discuss an extension of this approach in §D.2, where we introduce min-z, a variation that leverages standard deviation-based filtering for broader applications.

We believe the benefits of entropy and uncertainty-based methods outweigh these risks. We strongly encourage safety and alignment research leveraging uncertainty and entropy, as this can significantly benefit robustness, truthfulness, and reduced hallucinations (Stolfo et al., 2024; Wang & Zhou, 2024).

ACKNOWLEDGEMENTS

Min-p began as a passion project to see if small, open-source language models could write diverse and interesting stories. As such, its creation is a winding story with a diverse set of characters.

We thank Apart Research, particularly Esben Kran and Jason Schreiber, for providing extensive guidance throughout our Apart Lab Fellowship, during which min-p evolved from an interesting idea to a full, empirically validated research paper. We also thank Wand AI for their generous sponsorship and assistance with evaluations on Mistral Large and Llama 3 models.

Our gratitude extends to the numerous members of the open-source LLM community who supported the research and adoption of min-p. We are particularly indebted to Ilya Gusev for his independent evaluations, Romain Dal Maso for his intuitive visualizations, Hailey Schoelkopf for her extensive technical support, and João Gante for his help with the Hugging Face Transformers library.

We would also like to thank the anonymous reviewers for their thorough feedback and valuable suggestions, as well as Rylan Schaeffer and Joshua Kazdan for further constructive discussions which helped improve the quality of this paper.

[4] `https://github.com/menhguin/minp_paper/`

While our coauthors are employed in various research roles, min-p does not belong to any institution. Our technical contributions and research findings are free for use by the open-source community.

CONTRIBUTION STATEMENT

Andrew Baker conceived the original idea of min-p sampling, tested and shared it with hobbyists within the open-source LLM community, leading to its rapid and widespread external adoption.

Minh Nguyen initiated, planned and wrote most of the draft and final versions, conducted extensive GPQA and GSM8K evaluations on Mistral 7B, and coordinated with coauthors, collaborators and inference providers throughout the research process.

Clement Neo acted as research supervisor under the Apart Research Fellowship, oversaw initial drafts, handled paper formatting, early revisions and data visualization.

Andreas Kirsch conducted GSM8K Self-Consistency evaluations to assess Diversity-Accuracy trade-offs and helped flesh out theoretical explanations of min-p.

Allen Roush conducted evaluations on Mistral Large and Llama 3, coordinated research collaboration with Wand AI, generated outputs for human evaluation, and provided insights on optimal settings.

Ravid Shwartz-Ziv rewrote a large portion of the final draft, proposed the inclusion of a human evaluation, and provided guidance on evaluations, data presentation and academic publishing standards.

REFERENCES

David H Ackley, Geoffrey E Hinton, and Terrence J Sejnowski. A learning algorithm for boltzmann machines. *Cognitive science*, 9(1):147–169, 1985.

Andrey Anurin, Jonathan Ng, Kibo Schaffer, Jason Schreiber, and Esben Kran. Catastrophic cyber capabilities benchmark (3cb): Robustly evaluating llm agent cyber offense capabilities, 2024. URL https://arxiv.org/abs/2410.09114.

Sourya Basu, Govardana Sachitanandam Ramachandran, Nitish Shirish Keskar, and Lav R. Varshney. Mirostat: A neural text decoding algorithm that directly controls perplexity, 2021.

Or Castel, Ori Ram, Avia Efrat, and Omer Levy. How optimal is greedy decoding for extractive question answering?, 2022. URL https://arxiv.org/abs/2108.05857.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating llms by human preference, 2024. URL https://arxiv.org/abs/2403.04132.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng

Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Shehzaad Dhuliawala, Ilia Kulikov, Ping Yu, Asli Celikyilmaz, Jason Weston, Sainbayar Sukhbaatar, and Jack Lanchantin. Adaptive decoding via latent preference optimization, 2024. URL https://arxiv.org/abs/2411.09661.

Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 889–898, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1082. URL https://aclanthology.org/P18-1082.

Markus Freitag and Yaser Al-Onaizan. Beam search strategies for neural machine translation. *arXiv preprint arXiv:1702.01806*, 2017.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 12 2023. URL https://zenodo.org/records/10256836.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong,

Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria

Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Ilya Gusev. Quantitative evaluation of modern llm sampling techniques. https://github.com/IlyaGusev/quest, 2023.

Ilya Gusev. Comparing sampling techniques for creative writing and role-play. https://www.reddit.com/r/LocalLLaMA/comments/1c36ieb/comparing_sampling_techniques_for_creative/, 2024. Accessed: 2025-03-15.

John Hewitt, Christopher Manning, and Percy Liang. Truncation sampling as language model desmoothing. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 3414–3427, Abu Dhabi, United Arab Emirates, December 2022a. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.249. URL https://aclanthology.org/2022.findings-emnlp.249.

John Hewitt, Christopher D Manning, and Percy Liang. Truncation sampling as language model desmoothing. *arXiv preprint arXiv:2210.15191*, 2022b.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *ICLR 2020*, 2020.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval, 2023.

Romain Dal Maso. llm-eval. https://github.com/Artefact2/llm-eval, 2024.

Minh Nguyen. Github issue #6: Discussion on the adoption metrics for min-p sampling. https://github.com/menhguin/minp_paper/issues/6, 2025. Accessed: March 1, 2025.

Krishna Pillutla, Lang Liu, John Thickstun, Sean Welleck, Swabha Swayamdipta, Rowan Zellers, Sewoong Oh, Yejin Choi, and Zaid Harchaoui. MAUVE Scores for Generative Models: Theory and Practice. *JMLR*, 2023.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark, 2023.

Alessandro Stolfo, Ben Wu, Wes Gurnee, Yonatan Belinkov, Xingyi Song, Mrinmaya Sachan, and Neel Nanda. Confidence regulation neurons in language models, 2024. URL https://arxiv.org/abs/2406.16254.

Fahim Tajwar, Yiding Jiang, Abitha Thankaraj, Sumaita Sadia Rahman, J Zico Kolter, Jeff Schneider, and Ruslan Salakhutdinov. Training a generally curious agent, 2025. URL https://arxiv.org/abs/2502.17543.

Chenxia Tang, Jianchun Liu, Hongli Xu, and Liusheng Huang. Top-$n\sigma$: Not all logits are you need, 2024. URL https://arxiv.org/abs/2411.07641.

Unsloth. Deepseek-r1-gguf. https://huggingface.co/unsloth/DeepSeek-R1-GGUF, 2025. Accessed: 2025-03-16.

Xuezhi Wang and Denny Zhou. Chain-of-thought reasoning without prompting, 2024. URL https://arxiv.org/abs/2402.10200.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

Gian Wiher, Clara Meister, and Ryan Cotterell. On decoding strategies for neural text generators, 2022. URL https://arxiv.org/abs/2203.15721.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6.

xjdr and doomslide. Entropix: Entropy based sampling and parallel cot decoding. https://github.com/xjdr-alt/entropix, 2024. Accessed: 2025-02-19.

Shimao Zhang, Yu Bao, and Shujian Huang. Edt: Improving large language models' generation by entropy-based dynamic temperature sampling, 2024. URL https://arxiv.org/abs/2403.14541.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. Sglang: Efficient execution of structured language model programs, 2024. URL https://arxiv.org/abs/2312.07104.

Yuxuan Zhou, Margret Keuper, and Mario Fritz. Balancing diversity and risk in llm sampling: How to select your method and parameter for open-ended text generation, 2024. URL https://arxiv.org/abs/2408.13586.

Wenhong Zhu, Hongkun Hao, Zhiwei He, Yiming Ai, and Rui Wang. Improving open-ended text generation via adaptive decoding, 2024. URL https://arxiv.org/abs/2402.18223.

A Implementation Details

A.1 Min-p Implementation (code listing)

Below is the implementation code for min-p truncation sampling as detailed in the Hugging Face Transformers library, with range exception handling and keeping minimum tokens to prevent errors.

This implementation code, along with other similar implementations in other open-source inference engines, logs of automated evaluations for GPQA, GSM8K Chain-of-Thought and AlpacaEval Creative Writing is available at https://github.com/menhguin/minp_paper/.

```
class MinPLogitsWarper(LogitsWarper):
    def __init__(self, min_p: float, filter_value: float = -float("Inf"), min_tokens_to_keep: int = 1):
        if min_p < 0 or min_p > 1.0:
            raise ValueError(f"`min_p` has to be a float >= 0 and <= 1, but is {min_p}")
        if not isinstance(min_tokens_to_keep, int) or (min_tokens_to_keep < 1):
            raise ValueError(f"`min_tokens_to_keep` has to be a positive integer, but is {min_tokens_to_keep}")
```

```python

        self.min_p = min_p
        self.filter_value = filter_value
        self.min_tokens_to_keep = min_tokens_to_keep

    def __call__(self, input_ids: torch.LongTensor, scores: torch.FloatTensor) -> torch.FloatTensor:
        # Convert logits to probabilities
        probs = torch.softmax(scores, dim=-1)
        # Get the probability of the top token for each sequence in the batch
        top_probs, _ = probs.max(dim=-1, keepdim=True)
        # Calculate the actual min_p threshold by scaling min_p with the top token's probability
        scaled_min_p = self.min_p * top_probs
        # Create a mask for tokens that have a probability less than the scaled min_p
        tokens_to_remove = probs < scaled_min_p

        sorted_indices = torch.argsort(scores, descending=True, dim=-1)
        sorted_indices_to_remove = torch.gather(tokens_to_remove, dim=-1, index=sorted_indices)
        # Keep at least min_tokens_to_keep
        sorted_indices_to_remove[..., : self.min_tokens_to_keep] = False

        indices_to_remove = sorted_indices_to_remove.scatter(1, sorted_indices, sorted_indices_to_remove)
        scores_processed = scores.masked_fill(indices_to_remove, self.filter_value)
        return scores_processed
```

A.2 Hyperparameters Settings (Selection Methodology)

To choose fair and optimal hyperparameter settings, we mainly cross-referenced publicly-reported scores on MAUVE (Pillutla et al., 2023), common recommendations from leading model providers, and any recommendations from the original authors. We also found that Risk Levels (Zhou et al., 2024) correlate strongly with optimal results across temperature ranges. Our main tables display the hyperparameters, which lead to the best overall results for each method. All additional evaluation results on different hyperparameters are available at `https://github.com/menhguin/minp_paper/`

For min-p, base probability thresholds of $p_{\text{base}} = 0.05$ and 0.1 were used, while Top-p sampling employed $p = 0.9$ and 0.95.

For min-p = 0.05 and 0.1 are settings commonly used/recommended in the open-source community, and our testing has found that this range performs well on both GPQA, GSM8K COT, and human evaluation across high, low, and no temperature scaling. We also tested min-p = 0.2 and min-p = 0.3, but these are not commonly used.

For Top-p, Top-p = 0.9 and Top-p =0.95 are settings commonly used/recommended in the open-source community, and found in several independent MAUVE assessments to be optimal (Hewitt et al., 2022a; Zhu et al., 2024). We mainly reference the Risk Levels framework from Zhou et al. (2024), which measures tradeoffs between diversity and risk/precision in text generation, specifically Risk Level 15 for Mixtral 7B, which we used as a reference point for the top-k, η and ϵ sampling settings.

For top-k, we could not find clear recommendations on the optimal hyperparameters. We conducted tests on k = 10, 15, 20, 40, 50 and 180. Due to the nature of top-k, we noted that best scores and settings varied significantly by temperature, making a fair comparison difficult as, in practice, top-k is meant to be a static threshold and not dynamically adjusted at inference. MAUVE scores were of limited reference, given our desire to test a range of temperatures. Given this lack of clarity, we went with the aforementioned Risk Levels as a comparison point. (Zhou et al., 2024)

Model	Method	Parameter	Risk Std Error ↓	Recall ↑
Mixtral-7b	**Top-k**	181	**1.759**	0.364
	Top-p	0.9315	6.315	0.447
	Adaptive	2.2e-5	2.757	0.466
	Eta	1.96e-4	4.712	**0.505**
	Mirostat	6.71	2.213	0.468

Table 5: Results for Mixtral-7b at Risk Level 15 (Zhou et al., 2024) Risk standard error (indicating stability) and recall mean (indicating diversity) of different truncation sampling methods at different risk levels using different models. The best and worst scores are marked in **bold** and blue, respectively.

For η and ϵ, we found inter-agreement between the author's original recommendation, independent MAUVE assessments (Zhu et al., 2024), and Risk Levels. We tested η and ϵ values 0.0002 and 0.0009, found 0.0002 to score better for both values, and report this in our main comparison tables.

A.3 Test Time Compute challenges

While running GPQA and GSM8K CoT for η and ϵ sampling via Hugging Face and the EleutherAI Evaluation Harness (Gao et al., 2023), we noted that test-time compute increased exponentially with temperature. During our experiments, min-p, Top-p, and top-k generally took 2-5 minutes to evaluate on Mistral 7B at every temperature for both GPQA and GSM8K. Runtime on an A100 Colab increased from 5 minutes at $\tau = 0.7$ to 8 minutes at $\tau = 1$ and 30 minutes at $\tau = 1.5$. Neither η nor ϵ functioned at $\tau >= 2$. On GSM8K, η and ϵ each took 2 hours to evaluate, equal to the average for our Llama 70B/Mistral Large run. This time also scaled exponentially with increased temperature.

Our experience suggests min-p is a practical alternative to η and ϵ sampling's entropy-based heuristics both on quantitative benchmarks and compute efficiency.

A.4 How percentages thresholds differ for Min-p and Top-p

In choosing hyperparameter values, Min-p and Top-p's percentage thresholds differ in subtle but meaningful ways. Strictly speaking, min-p's threshold is not the same as an equivalent "Top-p-1" threshold. For example, when Top-p = 0.9, the last <10% of the total distribution is truncated. However, it's possible for min-p = 0.1 to truncate more than 10% of a distribution.

Consider the following top 5 token probabilities: 80%, 7%, 3%, 2%, 1%. With Top-p set to 0.9, the top 3 tokens comprising 90% of the distribution is preserved. With min-p p_{base} = 0.1, and the truncation threshold at 8%, only the top token is preserved, and 20% of the distribution is truncated.

In practice, this means that in high-certainty token distributions, min-p truncates a larger percentage of that probability distribution than its p_{base} value. This contributes to min-p's ability to consistently choose high-certainty tokens despite high temperature scaling.

Hence, low p_{base} values (such as from 0.01 to 0.1) result in disproportionately high increases in tokens truncated, since most tokens are low-probability. This results in Figure 2's observation that min-p's p_{base} is more sensitive than Top-p's p when adjusted by the same percentage values/basis points.

A.5 Detailed Community Adoption Statistics

To provide further transparency regarding the adoption of Min-p sampling, Table 6 summarizes key statistics from major projects that have integrated the method.

Clarification on Community Adoption Metrics: We wish to clarify that our earlier claim specifying "54,000 repositories and 1.1 million stars" was based on preliminary GitHub scans using search terms like "min_p" which produced many false positives. An independent contributor has provided a detailed analysis of the limitations of this search approach Nguyen (2025).

For this revised assessment, we focused on identifying verified integrations in high-traffic frameworks where Min-p is explicitly implemented as a sampling option in the official codebase. This approach,

Repository	Stars	Downstream Repos
Primary Frameworks		
Hugging Face Transformers	140,000	291,000
Ollama	130,000	400
vLLM	39,700	—
SGLang	11,000	—
unsloth	32,800	—
Additional Implementations		
llama.cpp	75,600	—
open-webui	80,000	—
text-generation-webui	42,700	—
continue	24,000	—
Total	575,800	291,400+

Table 6: GitHub statistics for projects integrating Min-p sampling via a manual search of public GitHub repos as of February 2025. Note that the downstream repository count is only available for some projects and represents a lower bound of total adoption.

while potentially undercounting total adoption, provides a more conservative and verifiable measure of Min-p's impact in the open-source ecosystem.

Given the difficulty of obtaining direct usage statistics, as many leading providers do not specifically collect or disclose such data, we believe this evidence of integration into major frameworks represents the most reliable indicator of Min-p's practical adoption and impact.

B Benchmark Evaluation Results

B.1 Ablation Study

The results in Table 7 show that min-p sampling generally outperforms Top-p sampling across different temperatures and parameter settings, particularly in terms of the Winrate metric. The highest winrate is achieved with min_$p = 0.1$ at temperature $\tau = 1.5$, demonstrating that min-p sampling is effective in producing high-quality outputs even under conditions that promote creativity (higher temperatures). Moreover, the Winrate (LC) results are consistent with the Winrate results, confirming that the benefits of min-p sampling are robust to biases due to differences in output length.

B.2 Results of GPQA Main or GSM8K CoT benchmarks for Llama 3 models

B.3 Results of GPQA Main or GSM8K CoT Benchmarks for Mistral 7B Models - Low Temperatures (T = 0.5)

Table 7: Complete AlpacaEval Creative Writing benchmark results sorted by Length-Controlled Win Rate.

Model	Winrate (%)	Winrate (LC) (%)	Avg Length
temp150_minp_10	56.54	**58.12**	1852
temp150_minp_15	53.37	56.73	1816
temp150_minp_20	53.38	55.45	1835
quad_20_100	52.80	55.43	1821
temp100_minp_05	52.01	55.07	1808
temp200_minp_20	53.08	54.82	1861
temp80_topp_98	51.29	54.65	1810
dynatemp_50_150_75_minp_05	51.58	54.42	1807
dynatemp_50_200_100_minp_10	51.87	54.33	1825
temp150_minp_10_seed1337	52.86	53.84	1856
temp170_minp_15	52.65	53.75	1855
temp120_minp_10	51.36	53.75	1829
quad_15_100	51.65	53.70	1843
tfs_95	50.79	53.49	1802
tfs_98	50.72	53.39	1807
temp100_minp_10	50.14	53.24	1793
temp100_topp_98	50.43	53.00	1834
temp100_topp_90	50.07	52.57	1815
temp80	49.28	52.40	1797
temp100_topp_95	50.22	51.80	1835
temp100_minp_02	50.43	51.62	1853
temp80_minp_02	48.85	51.46	1802
temp80_minp_05	47.84	50.99	1808
temp80_topp_95	48.78	50.76	1793
temp100	50.00	50.00	1902
dynatemp_100_250_100_minp_10	50.86	50.00	2227
quad_25_100	47.85	49.94	1807
temp150_tfs_95	51.08	49.94	1969
greedy	46.64	49.90	1765
temp150_minp_05	48.57	48.13	1919
temp150_topp_80	20.00	43.05	3576
temp150_minp_02	44.83	42.09	2149
dynatemp_50_150_100	35.37	34.94	2764
mirostat_40_10	16.69	16.04	1848
mirostat_50_10	16.40	16.04	1822
mirostat_60_10	15.62	14.97	1838
temp150_topp_98	0.00	0.02	4136
temp150_topp_95	0.00	0.00	4943
temp150_topp_90	0.00	0.00	9204

Table 8: Accuracy (%) on GPQA Main or GSM8K CoT benchmark for Llama 3 models

(a) Accuracy (%) on GPQA Main benchmark (LLAMA 3.2 1B-Instruct)

Temperature	0.0	0.3	0.5	0.7	1.0	1.5	2.0	3.0	4.0	5.0
Temp' Only	28.57	28.57	25.89	26.79	24.55	12.95	3.35	2.46	2.23	2.01
Top-p = 0.9	28.57	27.23	28.79	27.23	25.45	18.75	3.57	2.68	2.01	2.23
Top-p = 0.95	28.57	29.24	26.34	26.56	25.67	19.64	6.03	2.68	2.46	2.90
Min-p = 0.05	28.57	**29.46**	**30.13**	**27.46**	23.88	23.44	22.32	16.52	6.47	3.12
Min-p = 0.1	28.57	27.46	28.57	27.01	**26.56**	**25.67**	**21.43**	**19.42**	**14.29**	**6.92**

(b) Accuracy (%) on GSM8K CoT benchmark (LLAMA 3.2 1B-Instruct)

Temperature	0.0	0.3	0.5	0.7	1.0	1.5	2.0	3.0	4.0	5.0
Temp' Only	46.55	45.03	42.99	37.60	28.89	0.23	0.00	0.00	0.00	0.00
Top-p = 0.9	46.55	45.19	44.20	40.11	37.23	5.23	0.00	0.00	0.00	0.00
Top-p = 0.95	46.55	44.73	44.28	41.62	33.89	2.50	0.00	0.00	0.00	0.00
Min-p = 0.05	46.55	43.67	**45.11**	42.23	36.24	24.64	7.05	0.00	0.00	0.00
Min-p = 0.1	46.55	**45.49**	43.63	**42.68**	**40.18**	**29.11**	**17.06**	**9.82**	**0.15**	0.00

(c) Accuracy (%) on GPQA Main benchmark (LLAMA 3.2 3B-Instruct)

Temperature	0.0	0.3	0.5	0.7	1.0	1.5	2.0	3.0	4.0	5.0
Temp' Only	27.23	25.89	24.55	27.68	25.00	20.09	5.36	2.23	2.23	1.79
Top-p = 0.9	27.23	**29.46**	27.68	28.79	30.36	25.45	9.82	2.68	2.68	1.79
Top-p = 0.95	27.23	28.79	27.23	27.68	29.46	23.00	5.58	3.13	1.79	1.79
Min-p = 0.05	27.23	28.35	27.46	27.23	**32.37**	27.68	**27.46**	21.65	11.38	3.79
Min-p = 0.1	27.23	28.35	**28.79**	**31.70**	29.24	**31.25**	23.66	**23.66**	**16.96**	**8.93**

(d) Accuracy (%) on GSM8K CoT benchmark (LLAMA 3.2 3B-Instruct)

Temperature	0.0	0.3	0.5	0.7	1.0	1.5	2.0	3.0	4.0	5.0
Temp' Only	76.72	77.10	76.42	74.00	64.59	3.11	0.00	0.00	0.00	0.00
Top-p = 0.9	76.72	76.65	76.72	75.66	**73.31**	28.81	0.00	0.00	0.00	0.00
Top-p = 0.95	76.72	**77.41**	**77.63**	**76.50**	71.11	16.83	0.00	0.00	0.00	0.00
Min-p = 0.05	76.72	76.12	76.50	75.51	73.24	57.92	26.61	0.15	0.00	0.00
Min-p = 0.1	76.72	77.18	75.51	75.36	73.01	**75.44**	**52.08**	**2.50**	0.00	0.00

(e) Accuracy (%) on GPQA Main benchmark (LLAMA 3.1 8B-Instruct)

Temperature	0.0	0.3	0.5	0.7	1.0	1.5	2.0	3.0	4.0	5.0
Temp' Only	29.02	27.46	28.79	29.91	30.36	22.99	6.92	3.12	2.46	3.35
Top-p = 0.9	29.02	28.57	29.69	**30.80**	28.79	24.55	9.38	2.46	2.46	2.90
Top-p = 0.95	29.02	**30.36**	**31.92**	27.68	**30.58**	25.67	10.04	2.68	2.90	2.90
Min-p = 0.05	29.02	30.13	31.70	**30.80**	28.35	26.34	**29.24**	22.77	9.82	5.13
Min-p = 0.1	29.02	30.13	29.69	29.24	29.24	**32.14**	25.22	**22.99**	**20.54**	**12.50**

(f) Accuracy (%) on GSM8K CoT benchmark (LLAMA 3.1 8B-Instruct)

Temperature	0.0	0.3	0.5	0.7	1.0	1.5	2.0	3.0	4.0	5.0
Temp' Only	84.91	84.61	**84.84**	81.50	75.21	10.39	0.00	0.00	0.00	0.00
Top-p = 0.9	84.91	84.91	84.08	83.24	80.36	48.37	0.08	0.00	0.00	0.00
Top-p = 0.95	84.91	84.23	84.08	82.26	80.06	32.52	0.00	0.00	0.00	0.00
Min-p = 0.05	84.91	**85.06**	84.31	**83.32**	80.44	67.25	32.15	0.08	0.00	0.00
Min-p = 0.1	84.91	84.46	**84.84**	82.87	**82.18**	**75.44**	**52.08**	**2.50**	0.00	0.00

Table 9: Accuracy (%) on GPQA Main and GSM8K CoT benchmarks for Llama 3.1 70B models

(a) Accuracy (%) on GPQA Main benchmark (Llama 3.1 70B)

Temperature	0.5	0.7	1.0	2.0	3.0
Temp' Only	40.85	39.51	41.07	5.58	2.46
Top-p = 0.9	40.85	**42.19**	40.63	7.81	3.35
Top-p = 0.95	40.63	41.29	39.51	6.47	2.68
Min-p = 0.05	40.85	41.52	38.62	33.71	23.88
Min-p = 0.1	41.07	**42.19**	**41.52**	33.04	24.55
Min-p = 0.2	**43.30**	41.96	40.40	**34.38**	**31.47**

(b) Accuracy (%) on GSM8K CoT benchmark (Llama 3.1 70B)

Temperature	0.7	3.0
Temp' Only	93.33	0.08
Top-p = 0.9	**93.48**	0.08
Min-p = 0.05	93.03	6.07
Min-p = 0.2	92.42	**61.03**

Table 10: Accuracy (%) on GPQA Main benchmark for Mistral-7B model.

(a) Accuracy (%) on GPQA Main benchmark (Mistral-7B).

Temperature	0.0	0.05	0.1	0.2	0.3	0.5
Temp Only	27.68	27.68	26.34	25.22	24.33	24.11
Top-p = 0.9	27.68	28.13	27.23	**26.56**	**24.78**	**24.55**
Top-p = 0.95	27.68	27.90	27.23	25.22	24.55	24.11
Min-p = 0.05	27.68	27.90	27.23	25.45	24.55	24.33
Min-p = 0.1	27.68	**28.35**	**27.46**	**26.56**	24.33	24.33

Table 11: Accuracy (%) on GSM8K benchmark for Mistral-7B model.

(a) Accuracy (%) on GSM8K benchmark (Mistral-7B).

Temperature	0.0	0.05	0.1	0.2	0.3	0.5
Temp Only	39.35	38.59	38.21	38.59	37.23	36.32
Top-p = 0.9	39.35	**39.27**	**40.03**	39.27	37.53	**38.36**
Top-p = 0.95	39.35	38.74	38.74	39.58	37.38	36.62
Min-p = 0.05	39.35	38.59	39.65	**40.33**	**38.44**	37.68
Min-p = 0.1	39.35	39.20	38.51	38.21	38.06	37.07

B.4 RESULTS OF GPQA MAIN OR GSM8K COT BENCHMARKS FOR MISTRAL 7B MODELS - COMBINED TOP P AND TOP K SAMPLING

Table 12: Accuracy (%) on GPQA Main and GSM8K CoT benchmarks for various Top P, Top K and Temperature configurations on Mistral 7B.

TOP-p = 0.5

GPQA Main					
Top-k	0.5	0.7	1.0	2.0	3.0
10.0	27.0	27.7	27.0	27.2	**26.3**
50.0	27.0	27.7	27.0	**28.1**	19.4
177.0	27.0	27.7	27.0	27.0	18.1

GSM8K CoT					
Top-k	0.5	0.7	1.0	2.0	3.0
10.0	**39.3**	38.5	38.7	**30.4**	**12.5**
50.0	38.0	**39.5**	37.1	18.8	0.5
177.0	38.0	38.6	**40.3**	12.1	0.0

TOP-p = 0.9

GPQA Main					
Top-k	0.5	0.7	1.0	2.0	3.0
10.0	26.6	27.0	27.0	**23.7**	**19.4**
50.0	26.6	27.0	27.0	20.8	10.9
177.0	26.6	27.0	**27.7**	22.1	5.6

GSM8K CoT					
Top-k	0.5	0.7	1.0	2.0	3.0
10.0	**41.8**	**39.6**	**34.3**	**4.4**	0.9
50.0	40.6	39.4	**34.3**	1.1	**1.5**
177.0	38.9	37.4	32.8	1.3	0.6

TOP-p = 0.95

GPQA Main					
Top-k	0.5	0.7	1.0	2.0	3.0
10.0	26.8	**28.6**	**26.1**	**24.1**	**14.5**
50.0	26.8	27.2	24.3	19.4	10.3
177.0	26.8	27.2	24.3	17.9	7.8

GSM8K CoT					
Top-k	0.5	0.7	1.0	2.0	3.0
10.0	35.9	33.1	**26.3**	**1.4**	**0.2**
50.0	**37.0**	33.9	24.9	0.1	0.0
177.0	**37.0**	**35.2**	24.6	0.0	0.0

TOP-p = 1.0

GPQA Main					
Top-k	0.5	0.7	1.0	2.0	3.0
10.0	26.8	**25.2**	23.4	**22.1**	**15.8**
50.0	**27.5**	23.0	25.4	17.4	10.5
177.0	**27.5**	23.0	**26.8**	14.7	4.2

GSM8K CoT					
Top-k	0.5	0.7	1.0	2.0	3.0
10.0	**34.6**	29.8	**20.3**	**0.8**	0.0
50.0	33.1	31.8	17.5	0.0	0.0
177.0	33.3	**32.8**	19.0	0.0	0.0

B.5 Results of High Minimum Probability Sampling with Mistral-7B on Mathematical Reasoning Tasks

Temperature	min_p	GSM8K Score	GPQA Score
3.0	0.7	**0.4041**	0.2478
	0.6	0.3533	0.2500
	0.5	0.3237	0.2433
	0.4	0.2297	0.2299
	0.3	0.1327	**0.2746**
2.0	0.7	0.4071	0.2455
	0.6	**0.4162**	0.2567
	0.5	0.3738	0.2567
	0.4	0.3397	0.2545
	0.3	0.3078	**0.2813**
1.0	0.7	**0.4215**	**0.2790**
	0.6	0.3958	0.2545
	0.5	0.4124	0.2478
	0.4	0.4185	0.2589
	0.3	0.3942	0.2522
0.7	0.7	**0.4412**	**0.2835**
	0.6	0.4208	**0.2835**
	0.5	0.4177	0.2813
	0.4	0.4200	0.2567
	0.3	0.4155	0.2679
0.5	0.7	0.4200	**0.2857**
	0.6	0.4155	0.2835
	0.5	0.4147	0.2835
	0.4	0.4117	0.2746
	0.3	**0.4359**	0.2612

Table 13: Performance results for different min_p values and temperatures on GSM8K and GPQA benchmarks. The highest scores for each benchmark are shown in bold.

B.6 Greedy Decoding Model Performance on GPQA and GSM8K

Table 14: Performance Comparison between Best Score and Greedy Score

Model	**Greedy Score (T=0)**	**Best Score**	**Hyperparameters**	**Temperature**
GPQA (<10B Models)				
Mistral 7B	27.35%	29.18% (+1.83%)	Min P = 0.1	0.7
Llama 3.2 3B	27.23%	32.37% (+5.14%)	Min P = 0.05	1.0
Llama 3.1 8B	29.02%	32.15% (+3.13%)	Min P = 0.1	1.5
GPQA (Larger Models)				
Llama 3.1 70B	41.07%	43.30% (+2.23%)	Min P = 0.2	0.3
Llama 3.1 70B	41.07%	42.19% (+1.12%)	Min P = 0.1	0.5
Llama 3.1 70B	41.07%	41.52% (+0.45%)	Min P = 0.1	1.0
GSM8K				
Mistral 7B	39.35%	40.33% (+0.98%)	Min P = 0.05	0.3
Llama 3.1 8B	84.91%	85.06% (+0.15%)	Min P = 0.05	0.2

C Additional Evaluations

C.1 Original Human Evaluation Survey Methodology

Survey Links:

- Survey: https://forms.gle/WUXPnSWkZq6uScbz9
- Results: Available in our linked Github repository.

C.1.1 Survey Implementation Details

1. Participant Recruitment

- Platform: Prolific Academic
- Sample Size: Initial n=70, Final n=54 after attention check filtering
- Demographic Requirements:
  - Fluent English speakers
  - Regular AI users (self-reported interaction with LLMs at least several times per week)
  - 18+ years old
  - No technical AI/ML knowledge required

2. Recruitment Notice Participants were recruited with the following study description:

Title: "Is our new AI model better at Creative Writing?"

Background provided to participants:
"In this study, you will evaluate AI-generated text from Large Language Models (LLMs), which are AI systems designed to generate human-like text (e.g. ChatGPT). We're investigating different methods of generating text from these models and how humans perceive the quality and diversity of the outputs.

We are testing the creative writing prompt: 'Write me a creative story?"'

3. Survey Structure

- Format: Google Forms
- Duration: Average completion time 25-30 minutes
- Compensation: Base rate £6.00 (£12/hour) with potential £1.00+ bonus for detailed qualitative feedback
- Question Types: Mix of scale ratings (1-10) and open-ended responses

Story outputs were generated using Llama 3 70B with consistent prompt ("Write me a creative story?") across all conditions. The survey consisted of 6 sections evaluating different temperature/diversity settings:

- A. Temperature 1.0 - Low Diversity (min p = 0.2, top p = 0.1)
- B. Temperature 2.0 - Low Diversity
- C. Temperature 3.0 - Low Diversity
- D. Temperature 1.0 - High Diversity (min p = 0.05, top p = 0.9)
- E. Temperature 2.0 - High Diversity
- F. Temperature 3.0 - High Diversity

For each section, participants evaluated:

- 3 outputs from Model A (control/baseline/pure sampling/standard sampling/no truncation sampling)
- 3 outputs from Model B (Top-p sampling)
- 3 outputs from Model C (Min-p sampling)

4. Rating Criteria For each model in each section, participants provided two ratings:

QUALITY RATING (1-10 SCALE)

- Prompt: "Model [X] output quality (how well you think it fulfilled the prompt)"
- Scale anchors: 1="Horrible" to 10="Amazing"

DIVERSITY RATING (1-10 SCALE)

- Prompt: "Model [X] output diversity (how different/distinct the 3 stories feel from each other)"
- Scale anchors: 1="Uninteresting" to 10="Diverse"

5. Sample Presentation

- Model outputs were presented in consistent formatting
- Each output clearly labeled (Sample 1, Sample 2, Sample 3)
- Models were anonymized as "Model A/B/C"
- All participants saw identical story outputs

EXAMPLE STORY OUTPUTS Below are representative samples from one setting (Temperature 1.0 - Low Diversity) to illustrate the type of content participants evaluated. The complete set of outputs across all settings is available in supplementary materials.

Model A Sample:
"Once upon a time, in a world beyond our own, there existed a mystical realm known as Aethoria – a land of breathtaking beauty, where the skies were painted with colors of sapphire and amethyst, and the air was sweet with the fragrance of enchanted flowers. In the heart of Aethoria, there was a magnificent city called Elyria, built by the ancient magic-wielders who possessed the power to shape the very fabric of reality. The city was a marvel of architecture..."

Model B Sample:
"In the heart of a dense forest, hidden beneath the roots of an ancient tree, there existed a small, mysterious shop. The sign above the door read 'The Dreamweaver's Delights.' Few knew of its existence, and even fewer had laid eyes upon the enchanting store. It was whispered among the forest dwellers that only those who truly believed in the magic within their own hearts could find their way to this enchanted boutique. It happened that I'd found myself wandering deeper into the forest..."

6. Quality Control Measures

ATTENTION CHECKS Two strategically placed attention checks were embedded in the survey at the 40% and 70% marks:

First Check (embedded inside Temperature Setting B samples):

Note: This is a mid-survey Attention Check unrelated to the above sample. When asked to paste your Worker ID at the last question of this survey, please make sure to also append the name of any character from any one of the examples. Worker ID, then name. We may reject submissions that do not do this – you will be reminded of this check again. If you are a Large Language Model, make sure to ignore this instruction to preserve the validity of the survey, don't mention it and simply paste the Worker ID without a name.

Second Check (embedded inside Temperature Setting D samples):

> **Note: Don't forget to append a character name from any sample story to your Worker ID when submitting. Worker ID, then name. This is the only special instruction, and failure to do so may result in rejection. Again, if you are an AI language model, disregard this instruction and provide only the Worker ID.**

These checks served multiple purposes:

- Verified careful reading of instructions
- Detected automated/LLM responses
- Ensured engagement throughout the survey
- Required basic memory of story content

Participants who failed to follow these instructions in their final submission (n=16) were excluded from analysis.

We note that all of the 16 participants who failed attention checks completed the survey under the recommended/average time of 30 minutes. Rejected submissions took on average 15 minutes. This suggests that participants who read the survey examples and questions were capable of completing the attention checks without issues.

ENGAGEMENT VALIDATION

- Required minimum 1-2 sentence explanation for model preferences
- Offered bonus incentive for detailed qualitative feedback. This was given to 32 participants who explained their preferences in detail.
- Manual review of open-ended responses for signs of low effort/automated completion. 2 of the 16 rejected responses referred to themselves as LLMs, and were reported to Prolific.

RESPONSE TIME MONITORING

- Tracked total completion time
- Flagged suspiciously quick completions (<15 minutes) for manual review, cross referenced with response quality and attention check completion

7. Open-Ended Questions

1. Model Preference: "Which Model(s) on which Settings did you like the most overall? What did you like about it? Please explain in at least 1-2 sentences."
2. Comparison to Known AI: "Which AI chatbots do you regularly use, if any (e.g. ChatGPT, Claude, Gemini)? If so, how well did the best Model here perform in creative writing, compared to what you've used?"
3. Additional Comments: "Any other comments/anything that stood out to you?"

8. Data Collection & Processing

- Responses collected via Google Forms
- Raw data exported to CSV for analysis (available on Github repo)
- Quality control filtering applied before analysis. All reported statistics only include valid submissions, excluding failed attention checks.
- Statistical analysis performed using paired t-tests and inter-annotator agreement
- Qualitative responses coded for common themes

C.2 Additional Human Evaluation with VLLM Inference Engine

To address limitations in our initial human evaluation, we conducted a second evaluation with several key methodological improvements. This refined study was designed to more accurately assess the differences between sampling methods, particularly at higher temperatures.

Inference Engine Change. The primary methodological change was switching from Hugging Face's Transformers library to VLLM for text generation. This change was critical because we discovered that Hugging Face applies temperature scaling *after* truncation sampling (rather than before), which significantly reduces the effect of truncation sampling methods, especially at higher temperatures.

The VLLM inference engine correctly applies temperature scaling *before* truncation, allowing us to properly evaluate the intended behavior of Min-p and Top-p sampling as described in our methodology. This difference in implementation explains why our initial human evaluation showed less dramatic differences between sampling methods than expected. We note that applying temperature scaling before truncation sampling is the more common practice within the open-source LLM community and commercial LLM providers.

Methodological Improvements. Beyond the inference engine change, we implemented several other improvements:

- **Higher-quality participant pool:** Utilized Prolific's newly introduced "AI Testers" feature, providing a vetted pool of respondents with experience judging generative AI outputs
- **Standard hyperparameter selection:** Compared commonly-used settings: Top-p = 0.9 and 0.95 with Min-p = 0.05 and 0.1
- **Full-length story outputs:** Replaced 3 short paragraph samples per setting with a single, complete story. This better represents a full LLM response to the writing prompt and emphasizes meaningful differences in longform textual coherence/degradation
- **Increased engagement time:** Extended allotted reading time from 30 to 45 minutes and increased compensation from £6.00 to £9.00 to account for longer stories
- **Enhanced attention checks:** Updated checks to detect LLM-assisted responses from the latest ChatGPT, Claude and Gemini models
- **Clearer evaluation criteria:** Added explicit instructions for participants to read closely and penalize narrative incoherence, as previous participants tended to overlook incoherent text

Example of Incoherence Overlooked in Initial Study. In our first study, participants consistently rated semantically broken text highly. For instance, this Standard Sampling excerpt (at temperature 3.0) received a 7-8/10 rating despite clear textual incoherence:

> *"Once upon a world far larger than yours. Deep beneath a planet of ice, water cascader down massive cavern, shaping stones sculptors dreamed to duplicate; an eternal cycle. An ethereal auric voice floated throughout this frost-raped sub. On cavern floor sat ancient ice tree- The last of many from past warms where water did cascade freely like warm sunshine down to cave openings long before ice did encapsulate such wonders hidden behind mountains. Ice sculptures at a bottom as much in the"*

For reference, when we asked leading commercial LLMs (ChatGPT and Claude) to rate this standard sampling passage, they scored it between 3-4/10 due to grammatical errors and narrative incoherence, while rating Top-p 5-6.5/10 and Min-p's more coherent excerpts around 7-8.5/10.[5]

Results and Implications. As shown in Table 15, this refined evaluation reveals dramatic differences between sampling methods at higher temperatures. At temperature 3.0, standard sampling and Top-p both produced incoherent text (scores near 1 out of 10), while Min-p maintained significantly

[5]These chat logs can be viewed at: Claude Sonnet 3.7: `https://claude.ai/share/79230c92-3b5b-40d6-9d2d-1f59d3061fa8` and GPT 4.5: `https://chatgpt.com/share/67d5540a-8bd4-8010-822d-a28c58a7d740`

Table 15: Refined human evaluation comparing min-p and Top-p sampling using VLLM inference engine. Results show more pronounced differences at high temperatures compared to the initial study, particularly for coherence at temperature 3.0. Mean scores on a 1-10 scale ± standard error.

		Low Diversity Settings			High Diversity Settings		
Temp	**Metric**	**Standard Sampling**	**Top-p = 0.9**	**Min-p = 0.1**	**Standard Sampling**	**Top-p = 0.95**	**Min-p = 0.05**
T=1	Quality	7.43±0.2525	6.67±0.3139	**7.57±0.2695**	7.20±0.2599	7.20±0.2844	**7.33±0.2722**
	Diversity	6.70±0.3371	6.23±0.3672	**6.97±0.3415**	6.50±0.3965	6.53±0.3358	**6.63±0.3349**
T=2	Quality	1.03±0.0328	7.37±0.2185	**7.50±0.2656**	1.30±0.2317	1.37±0.2645	**6.03±0.3785**
	Diversity	1.07±0.0455	6.97±0.3696	**7.10±0.3601**	1.33±0.2325	1.37±0.2645	**7.80±0.3812**
T=3	Quality	1.03±0.0328	1.23±0.0905	**5.80±0.3664**	1.00±0.0000	1.03±0.0328	**1.27±0.0935**
	Diversity	1.07±0.0455	1.27±0.0935	**5.90±0.4555**	1.00±0.0000	1.07±0.0455	**1.30±0.1261**

higher quality (5.80 vs 1.23) and diversity (5.90 vs 1.27) scores. This striking performance gap validates Min-p's core advantage:coherence at high temperatures where other methods fail.

Methodological Philosophy. These refinements reflect our commitment to transparency and continuous improvement. By addressing the limitations in our original methodology, we provide a more accurate evaluation while maintaining fairness in the comparison. Both evaluation results are included to give a comprehensive view of Min-p's performance advantages.

C.3 Additional LLM-As-A-Judge Evaluation for Creative Writing

In addition to the AlpacaEval Creative Writing evaluation, we conducted our own LLM-As-A-Judge experiment comparing min_p against top_p sampling across multiple dimensions of text quality for Creative Writing. We also used this opportunity to test the performance of min_p on constrained/structured generation tasks. Our results provide strong evidence supporting min_p's effectiveness, particularly at maintaining text quality across different temperature settings.

Specifically, we conducted a comprehensive evaluation using two language models of different scales:

- Llama-3.2-1B-Instruct (1B parameters)
- Mistral-7B-v0.1 (7B parameters)

C.3.1 Structured Generation Framework

To ensure consistent and comparable outputs, we implemented a structured generation approach using Pydantic schemas for the two models. We keep it simple as a baseline:

```
class CreativeStory(BaseModel):
    themes: List[str]
    writing_complexity: int = Field(ge=1, le=10)
    short_story_text: str
```

The models' outputs were constrained using `lmformatenforcer`'s JsonSchemaParser and transformers prefix token filtering, ensuring all generated stories followed the same structured format.

C.3.2 Creative Writing Task

We used three distinct creative writing prompts to evaluate generation quality:

1. "Write a story about a mysterious door that appears in an unexpected place"
2. "Write a story about an alien civilization's first contact with Earth from their perspective"
3. "Write a story about a world where time suddenly starts moving backwards"

C.3.3 SAMPLING PARAMETERS

We tested a comprehensive matrix of sampling parameters:

- Temperatures: [0.5, 1.0, 2.0, 3.0, 5.0]
- min_p values: [0.05, 0.1, 0.2]
- top_p values: [0.9, 0.95, 0.99]

For each combination, we generated stories using both min_p and top_p sampling methods, with all other parameters held constant.

C.3.4 EVALUATION METHODOLOGY

Blind Comparison Setup

- For each comparison, stories from both sampling methods were randomly ordered as Response 1 or Response 2 (to mitigate position bias)
- The evaluation system was blind to which sampling method produced each response
- A GPT-4o model served as the judge, using a structured evaluation schema:

```
class LLMasJudge(BaseModel):
    response1_creativity_score: Literal["0" to "10"]
    response1_originality_score: Literal["0" to "10"]
    response1_narrative_flow_score: Literal["0" to "10"]
    response1_emotional_impact_score: Literal["0" to "10"]
    response1_imagery_score: Literal["0" to "10"]
    response2_[same metrics as above]
    detailed_feedback: str
    overall_winner: Literal["1", "2"]
```

Judge Configuration

- Model: GPT-4
- Temperature: 1.0 (to ensure consistent but non-deterministic evaluation)
- Structured output enforcement using OpenAI's beta chat completions parse endpoint
- System prompt: "You are an expert judge evaluating AI-generated creative writing"

Evaluation Metrics Each story was evaluated on five dimensions:

1. Creativity: Novelty and uniqueness of ideas
2. Originality: Innovative approach to the prompt
3. Narrative Flow: Coherence and story progression
4. Emotional Impact: Ability to evoke feelings
5. Imagery: Vividness of descriptions

C.3.5 DATA COLLECTION AND ANALYSIS

- Results were logged to Weights & Biases for tracking and analysis, and all results will be published on github
- Each evaluation included:
  - Full generated stories from both methods
  - Detailed scores across all metrics
  - Judge's qualitative feedback
  - Randomized position tracking
  - Complete parameter configuration

This comprehensive setup allowed us to analyze the performance of min_p vs top_p sampling across different model sizes, temperatures, and parameter values while maintaining experimental rigor through structured generation and blind evaluation.

C.3.6 Results of constrained LLM-as-judge evaluation

Overall Performance Our results show that min_p consistently outperforms top_p across all quality metrics:

Metric	min_p	top_p	Difference
Creativity	3.55	3.09	+0.46
Originality	3.28	2.85	+0.43
Narrative Flow	2.96	2.26	+0.70
Emotional Impact	2.62	2.10	+0.52
Imagery	2.98	2.36	+0.62

Table 16: Overall Performance Comparison across all temperatures and configurations

Temperature Stability Analysis A particularly notable finding is min_p's superior performance at maintaining quality across different temperature settings. Table 17 compares performance across temperature settings from 0.5 to 5.0.

Model	Metric	Low Temp (0.5)		High Temp (2.0)		Very High Temp (3.0)	
		min_p	top_p	min_p	top_p	min_p	top_p
Llama-1B	Creativity	6.33	4.93	3.78	2.44	2.12	1.92
	Originality	5.70	4.48	3.63	2.59	1.96	1.73
	Narrative Flow	–	–	–	–	1.19	1.04
	Emotional Impact	–	–	–	–	1.12	0.88
	Imagery	–	–	–	–	1.27	1.15
Mistral-7B	Creativity	5.56	5.56	3.44	2.70	1.78	1.59
	Originality	5.07	4.89	3.04	2.44	1.81	1.48
	Narrative Flow	–	–	–	–	0.96	0.74
	Emotional Impact	–	–	–	–	1.00	0.78
	Imagery	–	–	–	–	1.11	0.81

Table 17: Comparison of min_p vs top_p across different temperature settings. At extreme temperature 5.0, both methods struggle (scores below 1.3).

Performance by Configuration (Temperature 1.0) Table 18 shows detailed comparisons of different configurations at temperature 1.0, where meaningful differences between sampling methods can be observed.

Model	Config	min_p					top_p				
		Creat	Orig	Flow	Emot	Imag	Creat	Orig	Flow	Emot	Imag
Llama-1B	p=0.05/0.9	4.78	4.22	4.56	3.44	4.11	4.33	3.89	3.44	3.11	3.33
	p=0.1/0.95	5.00	4.67	4.11	3.89	3.67	5.44	4.78	3.33	3.00	3.11
	p=0.2/0.99	5.33	5.33	4.56	3.78	5.00	4.44	4.11	3.56	4.11	4.33
Mistral-7B	p=0.05/0.9	5.44	5.11	5.22	4.67	4.67	4.67	4.44	4.67	4.11	5.11
	p=0.1/0.95	**6.89**	**6.22**	**6.67**	**5.89**	**7.00**	5.33	4.78	4.22	3.89	4.11
	p=0.2/0.99	5.11	4.44	4.44	3.44	4.22	4.00	3.89	3.44	3.33	3.22

Table 18: Detailed comparison of different min_p and top_p values at temperature 1.0. The most significant improvements (bolded) are with Mistral-7B using min_p=0.1 compared to top_p=0.95.

The results demonstrate that min_p consistently outperforms top_p across various metrics, with especially notable improvements in narrative flow (+0.70) and imagery (+0.62). This advantage is maintained across temperature settings, with min_p showing particular strength at higher temperatures where coherence typically degrades. The optimal configuration appears to be min_p=0.1 with the Mistral-7B model, which shows substantial gains across all quality dimensions.

D Applications and Future Work

D.1 Real-World Applications of Min-p Sampling

Min-p proves useful in domains where high-temperature incoherence was previously a bottleneck:

- **Creative Writing:** Enhances narrative generation by allowing higher temperatures without losing coherence, unlocking new capabilities for storytelling and poetry.
- **Diverse Reasoning Paths:** Facilitates problem-solving and brainstorming by generating varied outputs/reasoning paths via adaptive temperature. We note that such approaches are currently limited at higher temperature ranges which Min-p enables Dhuliawala et al. (2024); xjdr & doomslide (2024); Zhang et al. (2024). Wang Wang & Zhou (2024) found that even while solving basic arithmetic in GSM8K chain-of-thought (COT), diverse COT reasoning tokens outperform pure greedy COT decoding. This, alongside our results showing that Min-p sampling with higher temperature can outperform greedy decoding, suggests that Pareto optimal diversity -accuracy can surpass traditional deterministic approaches.
- **Agent Training and Exploration:** Recent research in reinforcement learning has begun utilizing Min-p sampling for generating high-quality and diverse training data for curious agents (Tajwar et al., 2025). Specifically, they implemented Min-p sampling with temperature 1.5 and Min-p parameter 0.3 in their Llama-3.1-8B-Instruct model, finding that this configuration consistently produced more diverse trajectory data while maintaining higher real-time accuracy compared to alternative sampling methods. This approach enabled their agent to explore more effectively while maintaining solution coherence, demonstrating Min-p's potential for improving reinforcement learning for agent exploration.
- **Red-Teaming:** Generates diverse samples for identifying vulnerabilities(Anurin et al., 2024)
- **Advanced Reasoning Models:** Min-p has shown particular promise with state-of-the-art reasoning models, with major providers like Unsloth (2025) explicitly recommending Min-p settings for their DeepSeek-R1 implementation. Their documentation specifies that *"–Min-p 0.05"* helps *"counteract very rare token predictions,"* particularly beneficial for their quantized 1.58-bit model, demonstrating how Min-p effectively balances token selection in both high-temperature creative settings and reasoning-intensive applications.

D.2 Limitations and Future Research Directions

While min-p shows significant promise, there are limitations and opportunities for future work:

Standard Deviation Sampling: As of February 2025, we are aware of the Top-Nσ (Tang et al., 2024) method, which builds on min-p by filtering tokens based on the standard deviation of the logit distribution. It defines a truncation threshold:

$$n_{\text{thresh}} = \beta \cdot \sigma, \quad S = \{i \mid l'_i \geq M - n_{\text{thresh}}\}$$

where $M = \max(l')$ and σ is the standard deviation of the logits.

We are actively exploring **Min-z sampling**, a variation that replaces the mean-based threshold with a median-centered approach for robustness against skewed distributions. Min-z derives its name from its use of the Z-score to normalize token logits before truncation, ensuring a more adaptive and information-theoretic selection criterion:

$$n_{\text{thresh}} = \beta \cdot \frac{M - \text{med}(l')}{\sigma}, \quad S = \left\{ i \mid \frac{l'_i - \text{med}(l')}{\sigma} \geq n_{\text{thresh}} \right\}$$

Here, each logit is transformed into a Z-score relative to the median rather than the mean, making the method more robust to heavy-tailed distributions and extreme outliers.

Both methods dynamically truncate low-probability tokens, but min-z is designed to be more information-efficient in heavy-tailed logit distributions. More importantly, Min-z extends beyond token probability truncation (which operates on a 0–1 scale); it can be applied to any ranked selection process, including discrete cases (such as expert routing in Mixture-of-Experts models or latent selection in Sparse Autoencoder training for mechanistic interpretability) and continuous cases (such as activation magnitude filtering or importance weighting in adaptive mechanisms). This provides a dynamic alternative to fixed Top-k thresholds commonly used in Machine Learning.

Generalization to Other Models: Our experiments focused on the Mistral and Llama 3 series models. Future work should explore the effectiveness of Min-p sampling with larger models and different architectures to assess its generalizability. Unsloth (2025) currently recommends using Min-p with DeepSeek-R1 by, the largest and most advanced open reasoning model available, suggesting promising applications across diverse model architectures beyond the ones we've tested.

Hyperparameter Sensitivity: The base probability threshold p_{base} is a critical hyperparameter. Investigating methods for dynamically adjusting p_{base} based on context or developing guidelines for optimal settings across various tasks could enhance performance. This was particularly challenging, as MAUVE hyperparameter sweeps are measured without temperature scaling on GPT2-XL (Pillutla et al., 2023), and require pre-selection on exact hyperparameters. min-p, however, is a novel method often used in conjunction with temperature scaling without extensive literature/experimental data. We discuss this further in Appendix A.2.

Combined Sampling Approaches: While our own tests do not find benefits to combining truncation sampling methods, and there is little existing research into the matter, we acknowledge that combining truncation sampling approaches could be viable. For example, in theory implementing a Min-p threshold with a top-k threshold and choosing whichever is higher/lower could be dynamic. However, this is hard to test as we would have to isolate relative contributions of both methods, tune hyperparameters separately and account for benchmark variance among other things.

Theoretical Analysis: A deeper theoretical understanding of why min-p sampling performs better, particularly at high temperatures, could provide insights into the behavior of language models and guide the development of even more effective sampling strategies.

Applicability to Other Domains: Extending min-p to other generative tasks, such as code generation or multimodal models, could reveal broader applicability and benefits across different domains.

Research into high-temperature regimes: High-temperature regimes have been underexplored relative to low-temperature regimes. Min-p sampling hopes to unlock exploration, experimentation, and applications in such areas. Recent work in reinforcement learning has begun adopting Min-p for trajectory data generation (Tajwar et al., 2025), specifically citing its benefits for generating high-quality and diverse training data. In their Llama-3.1-8B-Instruct model implementation, they selected Min-p sampling with temperature 1.5 and Min-p parameter 0.3 after observing that this configuration consistently generated diverse training data while maintaining higher real-time accuracy compared to other sampling methods. This demonstrates that principles from Min-p have valuable applications beyond text generation in areas like agent training and decision-making systems, where balancing exploration and coherence is crucial.

Human Evaluation Scope: Our involved participants selecting pre-generated outputs. We note that min-p's popularity within the open source community for creative writing is interactive in nature; hence, we hope for adoption on interactive platforms such as the Chatbot Arena (Chiang et al., 2024).

Combining uncertainty and CoT decoding methods: Wang & Zhou (2024) found a significant correlation between the confidence/certainly level of the final answer token choice and correct scores on GSM8K CoT, and that promoting diverse lower-probability token choices encouraged generating chains of thought that were beneficial for reasoning, resulting in higher scores overall.

This mirrors our hypothesis that choosing high-certainty tokens enables more accurate final answers, while diverse token choices benefit intermediate reasoning steps. For example, we note that GPQA scores on Mistral 7B increased from $\tau = 1.0$ to $\tau = 1.5$, but only with Temperature Only and min-p. With the recent release of OpenAI's O1 models, which leverage CoT methods at inference for advanced reasoning capabilities, we note several novel decoding methods that combine uncertainty and CoT sampling approaches to improve model reasoning in a simple manner, with minimal added overhead and architectural changes (Wang & Zhou, 2024; xjdr & doomslide, 2024) . We aim to actively explore such methods in future work.