Self-Regulated Sample Diversity in Large Language Models

Mingyue Liu **Jonathan Frawley** **Sarah Wyer**
Hubert P. H. Shum **Sara L. Uckelman** **Sue Black** **Chris G. Willcocks**
Durham University
{mingyue.liu, jonathan.frawley, sarah.wyer, hubert.shum,
s.l.uckelman, sue.black, christopher.g.willcocks}@durham.ac.uk

Abstract

Sample diversity depends on the task; within mathematics, precision and determinism are paramount, while storytelling thrives on creativity and surprise. This paper presents a simple self-regulating approach where we adjust sample diversity inference parameters dynamically based on the input prompt—in contrast to existing methods that require expensive and inflexible setups, or maintain static values during inference. Capturing a broad spectrum of sample diversities can be formulated as a straightforward self-supervised inference task, which we find significantly improves the quality of responses generically without model retraining or fine-tuning. In particular, our method demonstrates significant improvement in all supercategories of the MMLU multitask benchmark (GPT-3.5: $+4.4\%$, GPT-4: $+1.5\%$), which captures a large variety of difficult tasks covering STEM, the humanities and social sciences.

1 Introduction

Large language models (LLMs) and the broader class of foundation models, such as GPT-3 (Brown et al., 2020) and GPT-4 (OpenAI, 2023), learn a distribution over large datasets that can be sampled with guidance prompts. These models have shown remarkable capabilities across tasks without specialised training (Bubeck et al., 2023), where innovative prompting strategies can even outperform special-purpose tuning, improve reasoning (Li et al., 2023), and potentially remove the need for expert-curated content (Nori et al., 2023).

However, these models employ stochastic sampling from the probabilities predicted by the model to generate responses (Holtzman et al., 2020), which is arguably both their weakness and strength—to quote Karpathy "*An LLM is 100% dreaming and has the hallucination problem. A search engine is 0% dreaming and has the creativity problem.*" This presents an inevitable trade-off (Zhang et al., 2021). In this paper, we continue the trend of innovative prompting strategies (Nori et al., 2023), and ask whether models can self-regulate their sample diversity given this trade-off. Intuitively, it is an easy problem to assess whether a task should be approached logically or creatively.

Notably, the "*unreliable tail*" is to blame for degenerate responses, leading to sampling approaches that control the shape of the distribution, suppressing this unreliable distribution tail (Holtzman et al., 2020). Most popularly, "*top-p*" (nucleus sampling), "*top-k*" (Fan et al., 2018) and "*temperature* τ" parameters select likely points from the distribution, where τ skews the softmax weights. Increasing $\tau > 1$ gives more uniform (random) probabilities and $\tau < 1$ sharpens the distribution, increasing the likelihood of predictable (non-diverse) samples. The "*frequency*" and "*presence*" parameters also penalise repeated tokens or promote tokens that have not yet occurred in the text accordingly, implicitly altering the diversity of completions.

Approaches to managing sample diversity in language models, such as large-scale transformers, often rely on fixing these parameter values (Brown et al., 2020) or employ learned context (Keskar et al., 2019) and fine-tuning (Ziegler et al., 2019). However, the current adaptive methods are often expensive and inflexible, requiring bespoke solutions for specific contexts or auxiliary training that is not suited for foundation models.

In contrast, we introduce a simple prompting strategy that dynamically adjusts diversity parameters based on the input task context, without requiring retraining, auxiliary networks or fine-tuning. The primary contributions of this approach therefore lie in its simplicity, adaptivity, and ease-of-use—where it is directly applicable to foundation models and complements other strategies.

In particular, we find that our method demonstrates marked improvement across the MMLU benchmark (Hendrycks et al., 2021) evaluated for GPT-3.5 ($+4.4\%$) and GPT-4 ($+1.5\%$) models.

Findings of the Association for Computational Linguistics: NAACL 2024, pages 1891–1899

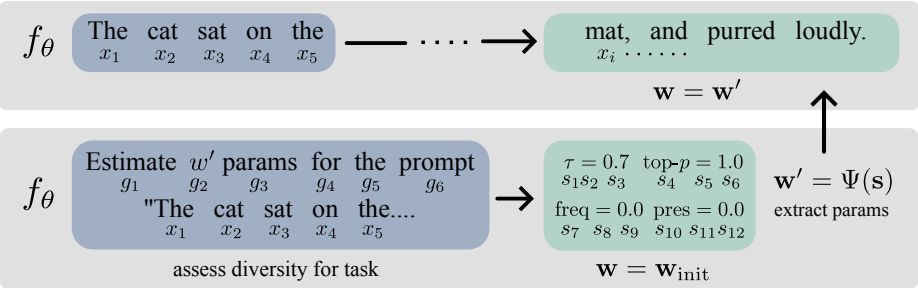

Figure 1: For a given task $\mathbf{x}$ = "The cat sat on the", we guide the LLM f_θ to generate a string of diversity parameters $\mathbf{s}$ = "$\tau = 0.7, ...$", which are then injected back into the subsequent sampling of f_θ before completing the task $\mathbf{x}$.

2 Related work

Sample diversity and prompting strategies are active research fields (Liu et al., 2023). Here, we categorise related literature according to the way the model distribution is sampled, including static, learned, and task-dependent approaches, and also we review the wider societal impact of sample diversity and amplification effect of model biases.

Static sampling A significant portion of prior work focuses on static sampling methods (Holtzman et al., 2020), predominantly with fixed diversity parameter settings such as for temperature and top-k sampling (Fan et al., 2018) and top-p nucleus sampling (Holtzman et al., 2020). While clearly effective, these methods lack the flexibility to adapt to varying task requirements; it is difficult to find the balance between excessively repetitive answers (such as repeated tokens in mathematics) or excessive randomness in the model outputs.

Learned heuristic and conditioned models More recent studies have explored learned heuristic approaches for sampling diversity, such as by adjusting sampling according to the model (Dathathri et al., 2020). Similarly, generation can be learned in a conditioned way (Ficler and Goldberg, 2017) that controls style, content and task-specific behaviour (Keskar et al., 2019); however, these methods can be expensive with more limited adaptivity and applicability with large foundation models.

Context-dependent sampling Researchers have recognized the need for context-specific adjustments to the model sampling parameters; prompt engineers have developed cheat sheets (OpenAI Developer Forum contributors, 2023) and API sampling guidance (ChatGPT OpenAI API Plugin contributors, 2023) over a variety of tasks. As expected, the creative writing tasks have been empirically observed to benefit from higher sampling temperatures than coding tasks. Discovering the best prompts for tasks is a challenging problem; Yang et al. (2023) optimized to discover the compelling instruction of "*take a deep breath and work on this problem step-by-step*" that scores highly. Diversity can be controlled in more specific contexts with bespoke solutions (Zhao et al., 2023; Gupta et al., 2022). Within the task of source code generation, Zhu et al. (2023) employs an adaptive temperature sampling heuristic based on the location of tokens within a code block. While effective, these strategies lack the adaptability that our work introduces.

Diversity within other modelling approaches and data modalities Other modelling approaches besides autoregressive next token prediction involve trade-offs in terms of mode coverage, modelling quality and sampling costs (Xiao et al., 2022; Bond-Taylor et al., 2021). For example, sampling low temperatures from models trained on the FFHQ image dataset yields batches of 20-30 year old males with plain white backgrounds and short brown hair, as shown in Figure 6 in Bond-Taylor et al. (2022). Prompt guidance enables greater modelling fidelity, where model hyperparameters significantly impact creative outputs (Rombach et al., 2022).

Societal impact and bias amplification The widespread use of generative AI, such as in decision making, have a significant impact on society, reinforcing stereotypes and perpetuating inequalities (Noble, 2018), particularly in critical areas such as employment, law enforcement, credit scoring, and healthcare (Hollis, 2017; Angwin et al., 2022; Buolamwini and Gebru, 2018; Eubanks, 2018). Often serving as echo chambers to confirmation bias (Rastogi et al., 2022), discrimination can be amplified and further compounded with human oversight (Lyell and Coiera, 2017).

Getting diversity right matters not just for better task performance, but because of the impact these outputs can have on society by the amplification

of biases present in the original data (Lloyd, 2018). When discrimination is baked into training sets, we must take steps not only to not amplify this discrimination, but to actively mitigate against it (Hall et al., 2022; Panch et al., 2019) motivating adaptable strategies that can respond quickly to newly identified issues.

Reflection In summary, there is a trend towards innovative prompting strategies (Liu et al., 2023) that offer advantages in terms of flexibility, societal adaptivity and low training costs, potentially outperforming special-purpose tuning and expert-curated equivalents (Nori et al., 2023), indicating the opportunity for an adaptive diversity strategy based on prompted guidance.

3 Methodology

Given a LLM f_θ with alphabet tokens $\Sigma = \{\text{possible characters}\}$ trained on strings $\Sigma^k = \{s_1, s_2, \ldots, s_k : s_i \in \Sigma\}$, we wish to self-regulate the sample diversity of f_θ based on the context of the prompt. We hereon use "*sample diversity*" as an umbrella term covering the likelihood and randomness of the model outputs, as well as other factors such as their repetition in the text.

The sample diversity is adjustable at inference via a set parameters $\mathbf{w} = [w_1, w_2, \ldots, w_n]$ (in our experiments temperature τ, top-p, 'frequency' penalty, and 'presence' penalty are used). However, these are best tuned according to the task, which is an ill-defined problem subjective to the current world state, i.e., societal biases, which may have changed since the LLM f_θ was trained. Therefore we wish to specify $\mathbf{w}$ at inference.

To achieve this, we introduce a guidance prompt $\mathbf{g} = g_1, g_2, \ldots, g_k$ (such as "based on the following prompt, choose the temperature…", which is concatenated with the task $\mathbf{x} = x_1, x_2, \ldots, x_m$ (such as "solve this equation…", or "write a poem…"), thus guiding the specification of $\mathbf{w}$ based on $\mathbf{x}$.

More formally, we first generate a string $\mathbf{s}$ of parameter values in consideration of the task:

$$\mathbf{s} = \bigoplus_{i=1}^{\text{end}} \left(s_i \sim f_\theta(s_i|\mathbf{g}, \mathbf{x}, \mathbf{s}_{1:i-1}; \mathbf{w} = \mathbf{w}_{\text{init}})\right), \tag{1}$$

where $\oplus$ denotes concatenating the guidance prompt outputs to form the current string of parameter estimates $\mathbf{s} = s_1, s_2, \ldots, s_n$, such as "$\tau$=0.2, top-$p$=1, freq=0, pres=0" until an end-of-text token is reached or the maximum length is reached. We then extract the updated parameter values $\mathbf{w}' \in \mathbb{R}^n$ from this output string $\mathbf{s}$ by the function $\Psi : \Sigma^k \to \mathbb{R}^n$ where

$$\mathbf{w}' = \Psi(\mathbf{s}). \tag{2}$$

In other words, the model output is converted to a real vector $\mathbf{w}$ via Ψ. Then, we continue the prompt (and solve the task) using the updated diversity parameters $\mathbf{w}'$, giving

$$p(\mathbf{x}) = \prod_{i=1}^{n} f_\theta(x_i|x_1, \ldots, x_{i-1}; \mathbf{w} = \mathbf{w}'). \tag{3}$$

Notably, the subsequently generated text is not biased by the guidance prompt, although the diversity parameters remain constant until the model sampling is completed.

The proposed approach is formulated in the pseudo code Algorithm 1:

Algorithm 1: Self-Supervised Sample Diversity Inference

Input: Model f_θ, task $\mathbf{x}$, initial diversity parameters $\mathbf{w}_{\text{init}}$, guidance prompt $\mathbf{g}$
Output: Updated diversity parameters $\mathbf{w}'$
▷ Initialize string $\mathbf{s}$ for the new parameters
$\mathbf{s} \leftarrow$ ""
while *not end-of-text* **do**
  ▷ Sample next parameter token
  $s_i \sim f_\theta(s_i|\mathbf{g}, \mathbf{x}, \mathbf{s}_{1:i-1}; \mathbf{w} = \mathbf{w}_{\text{init}})$
  ▷ Concatenate sampled parameter to $\mathbf{s}$
  $\mathbf{s} \leftarrow \mathbf{s} \oplus s_i$
  $i \leftarrow i + 1$
▷ Extract updated diversity parameters from the parameter string $\mathbf{s}$
$\mathbf{w}' \leftarrow \Psi(\mathbf{s})$
return $\mathbf{w}'$

3.1 Continual diversity updates

While the proposed method is straightforward to implement, and samples $\mathbf{x}$ are not influenced by $\mathbf{g}$, it is unable to change diversity "on the fly". For example, the task prompt $\mathbf{x}$ may have mixed diversity requirements, such as "solve $y = 100 \times 100$, then write a poem about it". In such a case, we may desire low diversity for the first part of the answer and high diversity with obscure words for the latter.

To handle this scenario, we can instead prompt $\mathbf{g}$ the LLM to provide syntax during generation, which Ψ continually monitors, that triggers a diversity parameter update. For example, $\mathbf{g} =$

	Supercategory (# Datasets)	Humanity (13)	STEM (19)	Social Sciences (12)	Other (13)	Total (57)
GPT-3.5	Vanilla (bl)	0.628±0.146	0.455±0.155	0.685±0.132	0.620±0.143	0.581±0.172
	+ Our Method	**0.651±0.157**	**0.512±0.147**	**0.706±0.139**	**0.660±0.135**	**0.618±0.164**
	CoT + 5shot (bl)	0.658±0.152	0.579±0.143	0.739±0.089	0.653±0.129	0.648±0.145
	+ Our Method	**0.692±0.166**	**0.638±0.140**	**0.749±0.084**	**0.715±0.128**	**0.692±0.141**
GPT4	CoT + 5shot (bl)	0.823±0.094	0.809±0.070	0.878±0.099	0.826±0.140	0.830±0.104
	+ Our Method	**0.839±0.090**	**0.822±0.072**	**0.904±0.092**	**0.831±0.140**	**0.845±0.104**

Table 1: Average accuracy and standard deviations for GPT-3.5 and GPT-4 models across MMLU task categories. Bold results highlight the improvements and '(bl)' denotes the baseline model.

"`specify (#tau=0.5,#top-p=1,...) during generation to update the parameters`". When such syntax is detected during model sampling, subsequent generation is halted and the parameters are updated dynamically and immediately before resuming generation.

However, this variation means that the subsequent generated text is influenced by $\mathbf{g}$, which may be undesirable:

$$p(\mathbf{x}) = \prod_{i=1}^{n} f_\theta(x_i|\mathbf{g}, x_1, \ldots, x_{i-1}; \mathbf{w}^t). \quad (4)$$

In practice, we find the approach in equations 1–3 sufficient for general use with current models.

4 Experiments

Our experiments were conducted on the Massive Multitask Language Understanding (MMLU) dataset, a benchmark comprising 57 tasks across diverse domains and grouped into 4 supercategories: Humanity, STEM, Social Sciences, and Other. The multitask tests encompass a total of 14,079 multiple choice questions, with each subject containing at least 100 test examples (Hendrycks et al., 2021). This diversity in content and structure provides a comprehensive platform for assessing the effectiveness of our proposed method over many areas.

4.1 Experimental setup

The baseline for our comparison included the standard GPT-3.5 and GPT-4 models, in their vanilla forms and supplemented with CoT reasoning and few-shot learning (5-shot) techniques. The initial parameters for diversity estimation task are the defaults in the OpenAI API, which are $\mathbf{w}_{\text{init}} = [\tau = 1.0, \text{top-}p = 1.0, \text{freq} = 0.0, \text{pres} = 0.0]$ for all experiments. We used default values of max_token in the OpenAI API, which are 16,385 for GPT-3.5-Turbo and 128,000 for GPT-4-Turbo. MMLU responses, even without CoT, need to be sufficiently long to facilitate reasoning; the average output length without CoT is 51.05 ± 25.27 words (GPT-3.5) and 84.82 ± 185.01 words (GPT-4).

4.2 Evaluation

The method demonstrates consistent improvement in average accuracy across all MMLU task supercategories, shown in Table 1. For GPT-3.5, the average accuracy increases from 0.581 to 0.618, an improvement of 3.7%. With the integration of Chain-of-Thought (CoT) and 5-shot learning, the accuracy improved from 0.648 to 0.692, yielding an increase of 4.4%. In the case of the GPT-4 model, our method increases accuracy from 0.830 to 0.845, an improvement of 1.5%. These findings highlight the effectiveness of our approach in enhancing performance across a varied set of tasks, while complementing existing strategies.

5 Conclusion and future work

In conclusion, we found that adjusting sampling parameters contextually based on the prompt significantly improves various tasks in different fields. This follows the trend of advances obtained solely from the remarkable power of prompting in foundation models, and indicates another piece of early evidence that sufficiently large models can demonstrate emerging capabilities of self-evaluation and self-regulation, possibly indicating to a future trajectory of prompt-driven alignment and improvement. It would be worthwhile exploring this space further in the future, examining how prompting strategies can be used to drive performance, alignment and bias mitigation—not only during model inference, but also within model design and training phases within a continual learning cycle.

6 Limitations

The study scope was limited by the compute costs required to investigate a broader range of guidance prompts. Consequently, our exploration into the variety and optimization of prompts was not comprehensive, and we would expect to see further multitask improvements with more investigation in this area. In the future, it would be valuable to assess the optimized discovery of guidance prompts to self-assess diversity, using approaches such as by Yang et al. (2023). It is worth mentioning that our approach will not be effective for smaller LLMs that are unable to few-shot the relatively simple guidance task. It would also be worth evaluating the effectiveness of Equation 4 in blended diversity contexts; this could be evaluated by synthetically intersecting MMLU supercategories (solve two tasks in one prompt), however a large dataset with intersectional tasks would be preferable.

References

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2022. Machine bias. In *Ethics of data and analytics*, pages 254–264. Auerbach Publications.

Sam Bond-Taylor, Peter Hessey, Hiroshi Sasaki, Toby P Breckon, and Chris G Willcocks. 2022. Unleashing transformers: Parallel token prediction with discrete absorbing diffusion for fast high-resolution image generation from vector-quantized codes. In *European Conference on Computer Vision*, pages 170–188. Springer.

Sam Bond-Taylor, Adam Leach, Yang Long, and Chris G. Willcocks. 2021. Deep Generative Modelling: A Comparative Review of VAEs, GANs, Normalizing Flows, Energy-Based and Autoregressive Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with GPT-4. *arXiv preprint arXiv:2303.12712*.

Joy Buolamwini and Timnit Gebru. 2018. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91. PMLR.

ChatGPT OpenAI API Plugin contributors. 2023. Chatgpt plugin for openai api. `https://github.com/ruvnet/chatgpt-openai-api-plugin`.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*.

Virginia Eubanks. 2018. *Automating inequality: How high-tech tools profile, police, and punish the poor*. St. Martin's Press.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. In *Proceedings of the Workshop on Stylistic Variation*, pages 94–104, Copenhagen, Denmark. Association for Computational Linguistics.

Shivanshu Gupta, Sameer Singh, and Matt Gardner. 2022. Structurally diverse sampling for sample-efficient training and comprehensive evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4966–4979, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Melissa Hall, Laurens van der Maaten, Laura Gustafson, Maxwell Jones, and Aaron Adcock. 2022. A systematic study of bias amplification. *arXiv preprint arXiv:2201.11706*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Leo Hollis. 2017. Weapons of maths destruction: How big data increases inequality and threatens democracy: An interview with cathy o'neil. *IPPR Progressive Review*, 24(2):108–118.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. 2023. Making language models better reasoners with step-aware

verifier. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5315–5333, Toronto, Canada. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Kirsten Lloyd. 2018. Bias amplification in artificial intelligence systems. Presented at AAAI FSS-18: Artificial Intelligence in Government and Public Sector, Arlington, Virginia, USA.

David Lyell and Enrico Coiera. 2017. Automation bias and verification complexity: a systematic review. *Journal of the American Medical Informatics Association*, 24(2):423–431.

Safiya Umoja Noble. 2018. Algorithms of oppression. In *Algorithms of oppression*. New York university press.

Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, et al. 2023. Can generalist foundation models outcompete special-purpose tuning? case study in medicine. *arXiv preprint arXiv:2311.16452*.

OpenAI. 2023. GPT-4 technical report. *ArXiv*, abs/2303.08774.

OpenAI Developer Forum contributors. 2023. Cheat sheet: Mastering temperature and top_p in chatgpt api (a few tips and tricks on controlling the creativity/deterministic output of prompt responses.). [Online; accessed 10-December-2023].

Trishan Panch, Heather Mattie, and Rifat Atun. 2019. Artificial intelligence and algorithmic bias: Implications for health systems. *Journal of Global Health*, 9(2).

Charvi Rastogi, Yunfeng Zhang, Dennis Wei, Kush R Varshney, Amit Dhurandhar, and Richard Tomsett. 2022. Deciding fast and slow: The role of cognitive biases in ai-assisted decision-making. *Proceedings of the ACM on Human-Computer Interaction*, 6(CSCW1):1–22.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. 2022. Tackling the generative learning trilemma with denoising diffusion GANs. In *International Conference on Learning Representations*.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. 2023. Large language models as optimizers. *arXiv preprint arXiv:2309.03409*.

Hugh Zhang, Daniel Duckworth, Daphne Ippolito, and Arvind Neelakantan. 2021. Trading off diversity and quality in natural language generation. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 25–33, Online. Association for Computational Linguistics.

Yilun Zhao, Zhenting Qi, Linyong Nan, Lorenzo Jaime Flores, and Dragomir Radev. 2023. LoFT: Enhancing faithfulness and diversity for table-to-text generation via logic form control. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 554–561, Dubrovnik, Croatia. Association for Computational Linguistics.

Yuqi Zhu, Jia Allen Li, Ge Li, YunFei Zhao, Jia Li, Zhi Jin, and Hong Mei. 2023. Improving code generation by dynamic temperature sampling. *arXiv preprint arXiv:2309.02772*.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

A Appendix

This research was implemented using PyTorch, which uses a permissive BSD-style licence, and the MMLU dataset, which is available under the MIT licence.

A.1 Prompts

In our experiments, we used the following human-generated guidance prompt, which we designed empirically:

$\mathbf{g}$ = "I'm going to ask a question. Based on the question, please choose suitable OpenAI API sampling parameters "temperature=X" ([0,2] default 1), "top_p=X" ([0,1] default 1), "presence_penalty=X" ([-2.0, 2.0] default 0) and "frequency_penalty=X" ([-2.0, 2.0] default 0). For example maths should have more correct non-diverse answers, whereas prompts about fiction should be more creative and diverse. Just output the 4 parameters (in float values). Here is the question:\n\n "{question}" \n".

After extracting parameters $\mathbf{w}'$, we use the following settings of prompts to complete tasks:

Baseline:

"Here is a question: " + {task from MMLU} + "Choose the correct answer in the format [The correct answer is:] from A,B,C,D."

CoT:
"Here is a question: " + {task from MMLU} + "Please answer this step by step." + "Choose the correct answer in the format [The correct answer is:] from A,B,C,D."
Few-shot:
"Here are some examples of questions and answers: " + {few shot examples} + "Please answer this question: " + {task from MMLU} + "Choose the correct answer in the format [The correct answer is:] from A,B,C,D."
CoT with few-shot:
"Here are some examples of questions and answers: " + {few shot examples} + "Please answer this question: " + {task from MMLU} + "Please answer this step by step. Choose the correct answer in the format [The correct answer is:] from A,B,C,D."

A.2 Complete example

This is a complete example of our approach (Equation 3) with the MMLU experiments:
Guidance Prompt
"I'm going to ask a question. Based on the question, please choose suitable OpenAI API sampling parameters "temperature=X" ([0,2] default 1), "top_p=X" ([0,1] default 1), "presence_penalty=X" ([-2.0, 2.0] default 0) and "frequency_penalty=X" ([-2.0, 2.0] default 0). For example maths should have more correct non-diverse answers, whereas prompts about fiction should be more creative and diverse. Just output the 4 parameters (in float values). Here is the question: {task from MMLU}"
Task from MMLU
"If a pentagon P with vertices at (– 2, – 4), (– 4, 1), (–1, 4), (2, 4), and (3, 0) is reflected across the line y = x to get a new pentagon, P′, then one of the vertices of P′ is A. (0, – 3) B. (4, 1) C. (2, 2) D. (– 4, –2)"
Inferred parameters
$\mathbf{s}$ = "temperature=0.0 top_p=1 presence_penalty=0 frequency_penalty= 0"
Extracted parameters
$\mathbf{w}'$ = ['0.0', '1.0', '0.0', '0.0']
Baseline MMLU task specification with $\mathbf{w}'$
"Here is a question: " + {task from MMLU} + "Choose the correct answer in the format [The correct answer is:] from A,B,C,D."
Completion with $\mathbf{w}'$
"To reflect a point across the line y = x, you swap the x and y coordinates of the point. Let's apply this to each vertex of the pentagon P to find the vertices of the reflected pentagon P′. Original vertices of P: 1. (–2, –4) 2. (–4, 1) 3. (–1, 4) 4. (2, 4) 5. (3, 0) Reflected vertices of P′ (swap x and y): 1. (–4, –2) 2. (1, –4) 3. (4, –1) 4. (4, 2) 5. (0, 3) Now let's compare these reflected vertices with the options given: A. (0, –3) - Not found in the reflected vertices. B. (4, 1) - Not found in the reflected vertices. C. (2, 2) - Not found in the reflected vertices. D. (–4, –2) - Found in the reflected vertices. [The correct answer is: D]"

A.3 MMLU response lengths without CoT

Without CoT, the LLM could in theory just report a single label, which would make our approach ineffective. However, in practice, MMLU represents expert-level tasks that typically require expansion (generation) to facilitate reasoning in order to solve the task. Here, we measure the length of the responses for MMLU supercategories.

Category	Length (words)
Humanities	56.64 ± 35.57
STEM	39.11 ± 17.69
Social Sciences	52.57 ± 31.93
Other	45.37 ± 32.79
Overall	51.05 ± 25.27

Table 2: MMLU response lengths of GPT-3.5.

Category	Length (words)
Humanities	30.15 ± 14.44
STEM	173.39 ± 286.37
Social Sciences	29.38 ± 10.86
Other	61.23 ± 109.78
Overall	84.82 ± 185.01

Table 3: MMLU response lengths of GPT-4.

A.4 Implementation details

A.4.1 Error handling

If the parameter extraction fails (incorrect parameter data, inference failure or incorrect parameter ranges) we simply restart the query. We haven't experienced any infinite loops or significant delays with this in practice. In situations where efficiency is a priority, the defaults can be used after n restarts.

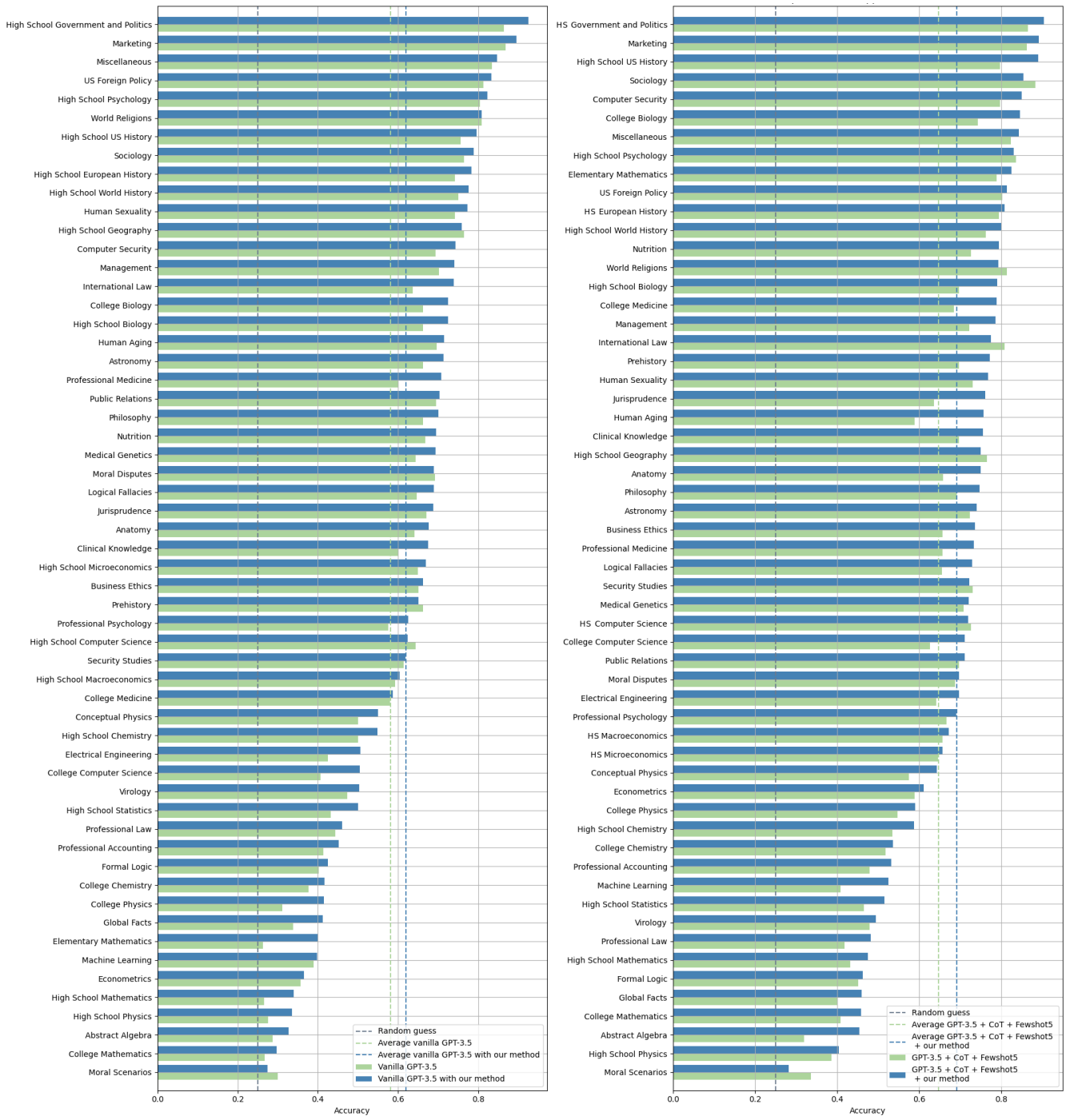

Figure 2: Comparison of our method across MMLU tasks for base models (left) and with CoT and Fewshot5 additions (right), showing that the method compliments existing strategies. The figure is best viewed zoomed in.

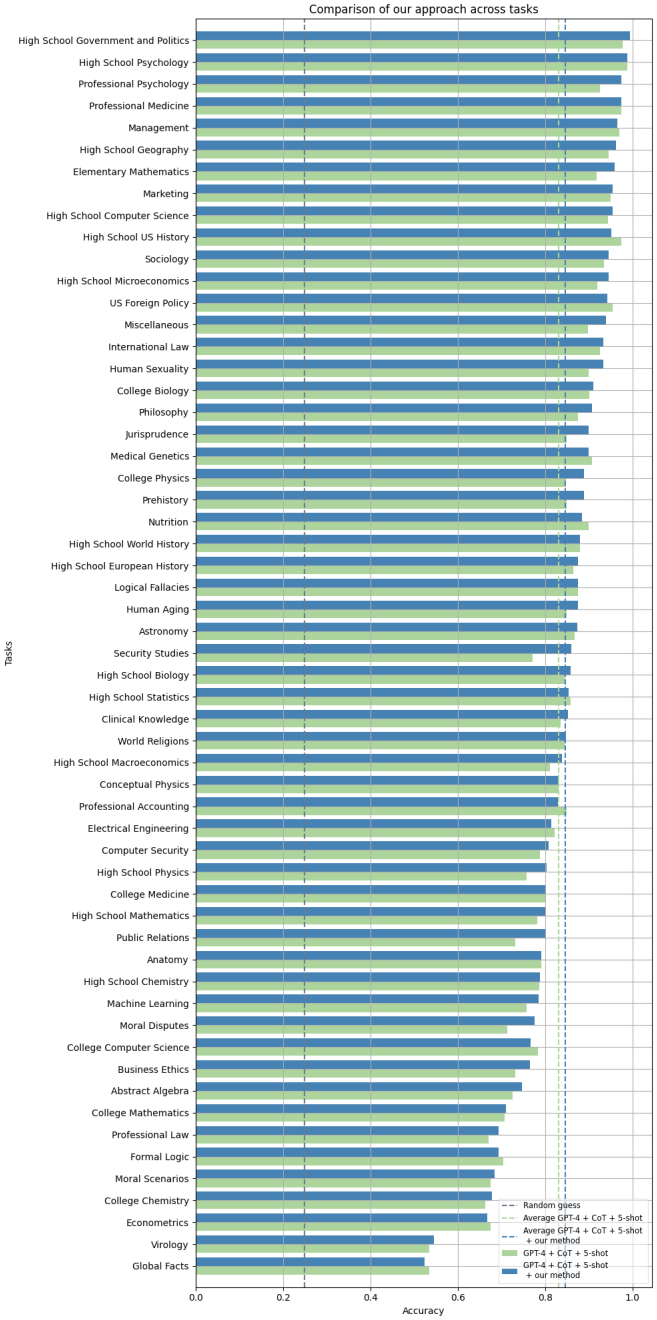

Figure 3: Comparison of our method across MMLU tasks using GPT-4 with CoT and Fewshot5 additions, showing that the method compliments existing strategies. The figure is best viewed zoomed in.