Extended Abstract

Motivation Reinforcement Learning from Human Feedback (RLHF) is a popular paradigm for fine-tuning language models when explicit ground truth rewards are difficult to obtain. Direct Preference Optimization (DPO) has emerged as a promising method due to its simplicity and lack of reliance on sampling-based reinforcement learning. However, standard DPO treats all preference examples equally, ignoring differences in prompt difficulty or informativeness. Inspired by recent curriculum learning research, we hypothesize that introducing a difficulty-aware curriculum can guide the model to learn from easy-to-hard examples, improving convergence stability and generalization.

Method We begin by fine-tuning the Qwen2.5-0.5B model using Supervised Fine-Tuning (SFT) on the SmolTalk data set to establish a strong instruction-following foundation. We then apply Direct Preference Optimization (DPO) on UltraFeedback to align the model to human preferences. To improve learning stability and generalization, we modify standard DPO by introducing a curriculum that gradually increases the difficulty of training examples across epochs. We experimented with multiple difficulty measures to sort the data: (1) reward margin, $RM(x) = |r(x, y^+) - r(x, y^-)|$, measuring how confidently the chosen response is preferred; (2) response length difference, $RLD(x) = ||y^+| - |y^-||$, capturing a shallow definition of structural variation between completions; and (3) prompt complexity, $PC(x) = \text{StdDev}_{i=1}^{k} \text{PPL}_{p_{LM}}(y^{(i)} \mid x)$, capturing a deeper semantic different between preferred and dispreferred completions based on perplexity variance as judged by an external language model p_{LM}. We found prompt complexity to be the most effective sorting strategy, yielding the best win-rate improvements. The resulting curriculum presents samples with less complex prompts earlier and gradually incorporates harder ones, allowing the model to build alignment capacity over time. However, at each epoch, we randomly incorporate 10% of the excluded samples, which we call the anti-curriculum data set, into the training set to prevent overfitting on easier structures.

Implementation Our implementation uses Qwen2.5-0.5B with a max sequence length of 1024 and prompt-response truncation to the 95th percentile length to avoid memory issues. We implement SFT with standard next-token prediction, and DPO with KL-penalty controlled preference classification. For curriculum DPO, we partition training data using complexity metrics and apply epoch-wise sampling to gradually introduce harder samples. Trained models from our experiments are evaluated on held-out UltraFeedback validation data, with win rate evaluated using the Nemotron-70B reward model.

Results Curriculum-guided DPO consistently outperformed standard DPO across multiple hyperparameter configurations. Our best model, which used prompt complexity as the complexity metric, achieved a validation win rate fold change of 1.150 over Vanilla DPO, which already outperforms the warm-started SFT baseline. We observed that early-stage learning was more stable with the curriculum, and that curriculum allowed effective exploration of difficult examples in later epochs.

Discussion Our results demonstrate the importance of training dynamics in preference optimization. Just like DPO is sensitive to hyperparameter settings, its performance is also sensitive to sample difficulty. Our curriculum framework provides an interpretable way to improve learning efficiency. Notably, from a qualitative perspective, the model is able to handle longer, more complex prompts more effectively, suggesting improved compositional generalization. This also highlights the potential of curriculum-based strategies to act as an implicit regularizer in alignment-focused training regimes.

Conclusion We show that integrating curriculum learning into DPO leads to improved alignment performance on preference data. Our curriculum-guided DPO framework is modular and effective across a wide range of hyperparameters. In future work, we aim to extend this framework to verifier-based tasks and explore multi-objective RL formulations that account for diverse human preferences. More broadly, our findings suggest that incorporating task structure and example difficulty can serve as a general recipe for stable and well-performing alignment training.

Supervised Fine-Tuning and Curriculum-Guided Direct Preference Optimization on Qwen2.5-0.5B

Christopher Sun
Department of Computer Science
Stanford University
csun27@stanford.edu

Abishek Satish
Department of Computer Science
Stanford University
absatish@stanford.edu

Abstract

Direct Preference Optimization (DPO) is a state-of-the-art method for fine-tuning language models with human preference data. However, it treats all training examples equally, which can limit stability and performance. In this work, we propose a curriculum-guided variant of DPO that introduces examples in order of difficulty, based on prompt complexity, response length difference, and reward margin. We first fine-tune Qwen2.5-0.5B using Supervised Fine-Tuning (SFT) on the SmolTalk data set, then apply our curriculum-based DPO on UltraFeedback preferences. Among several strategies tested, sorting by prompt complexity performed best. Our method improves early learning dynamics and enables better generalization on harder prompts. The final model achieves a 1.150-fold win rate improvement over a vanilla DPO implementation baseline on held-out validation data, as measured by the Nemotron-70B reward model. This shows that simple curriculum strategies can significantly enhance preference optimization.

1 Introduction

Instruction-tuned language models have demonstrated strong capabilities across a wide range of tasks, yet aligning them with human preferences remains challenging in real-world deployments. Reinforcement Learning from Human Feedback (RLHF) has emerged as the prevailing paradigm to bridge this gap, enabling models to learn from preference comparisons rather than explicit supervision. Among RLHF approaches, Direct Preference Optimization (DPO) has gained traction as a stable alternative to policy gradient methods like PPO. By formulating preference learning as a closed-form loss, DPO avoids the complexities of sampling and reward modeling, making it appealing for both academic research and practical alignment pipelines.

Despite its effectiveness, standard DPO applies uniform importance to all training examples, failing to account for varying levels of difficulty or informativeness. In practice, training data exhibits wide variation – some preference pairs are trivial to resolve, while others involve complex reasoning, subtle distinctions, or ambiguous intent. Applying equal weight to such diverse samples can lead to unstable training dynamics, over-regularization, or inefficient updates. These limitations are particularly pronounced when fine-tuning smaller models that lack the capacity to easily resolve conflicting signals.

Curriculum learning offers a principled solution to this issue by introducing training examples in a structured sequence from easy to hard. This technique, originally proposed by Bengio et al., has proven effective in both vision and language domains. Recent work such as 2D-Curri-DPO has extended curriculum learning to preference optimization, demonstrating that curriculum-guided sampling can improve both convergence and final alignment performance. However, such approaches often rely on dual metrics or handcrafted schedules, raising questions about generalizability. Our

goal is to investigate whether simpler curriculum heuristics coupled with rigorously defined difficulty estimates can deliver improvement over a naive DPO implementation.

To this end, we propose and evaluate a curriculum-guided variant of DPO, using prompt complexity, response length difference, and reward margin as potential difficulty measures. We hypothesize that by reordering the presentation of examples during training, we can improve both sample efficiency and generalization on unseen prompts. Our study begins with supervised fine-tuning of Qwen2.5-0.5B on SmolTalk followed by DPO training on UltraFeedback preference pairs. We implement curriculum schedules that partition training data into tiers based on precomputed difficulty scores and adjust sampling over time.

We compare these strategies against vanilla DPO under consistent settings, measuring performance via win rate on held-out data using the Nemotron-70B reward model. Our best model, trained with a prompt complexity-based curriculum, achieves a 1.150-time win rate improvement over the DPO baseline (where performance is evaluated against the warm-started SFT model), outperforming both uniform and reward-margin setups. These findings support the hypothesis that training dynamics, not just loss formulations, are critical to preference alignment. In the remainder of this report, we detail our methodology, results, and implications of our findings for future preference optimization pipelines.

2 Related Work

Curriculum learning, originally proposed by Bengio et al. Bengio et al. (2009), suggests that models can learn more effectively when training examples are presented in an easy-to-hard order. This paradigm has inspired a range of applications in NLP, including recent work by Liu et al. Liu et al. (2022), who demonstrate that curriculum-based scheduling significantly improves the efficiency of transformer models when learning to fill in the middle of sequences. Our work extends this line of thinking to preference optimization.

Most directly related to our approach is 2D-Curri-DPO by Li and Zhang Li and Zhang (2025), which applies a dual-difficulty curriculum to Direct Preference Optimization (DPO) by considering both prompt and response complexity. We build on this idea by testing multiple curriculum strategies, finding that sorting by prompt complexity alone yielded the most stable and performant training. Unlike Li and Zhang's multi-axis approach, our method maintains simplicity while achieving strong gains.

We also note that while traditional RLHF techniques Ouyang et al. (2022) and SFT-based alignment methods Wang et al. (2022) focus on high-quality response generation, they do not explicitly leverage difficulty-based curricula. Our results suggest that integrating curriculum learning into preference optimization is a promising direction for improving alignment quality and generalization in smaller models.

3 Methods

Our method consists of a two-stage training pipeline that combines supervised fine-tuning (SFT) with Direct Preference Optimization (DPO), followed by a curriculum-guided extension to improve optimization efficiency and generalization. We fine-tune Qwen2.5-0.5B, a decoder-only transformer pretrained for instruction following, and evaluate our models using win rate comparisons on held-out UltraFeedback preference pairs scored by the Nemotron-70B reward model.

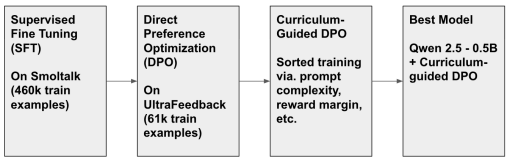

Figure 1: Curriculum-Guided Preference Optimization Workflow

3.1 Supervised Fine-Tuning (SFT)

We begin by fine-tuning Qwen2.5-0.5B on the SmolTalk data set, consisting of 460K training and 24K validation samples. The goal of SFT is to behaviorally initialize the policy model with instruction-following capabilities and coherent response generation. We employ a standard next-token prediction objective and truncate each input–response sequence to a maximum length of 1024 tokens. During preprocessing, we format the data set using the model's built-in chat template, retain only the first user-assistant turn pair, and truncate the assistant's response to the 95th percentile of observed response lengths to ensure consistency across examples.

The model is trained using the Adam optimizer with a learning rate of 5×10^{-5} and 100 warmup steps. We apply cosine learning rate scheduling and use mixed-precision training where available. Upon completion, this SFT model serves as both the initial policy and the frozen reference model for downstream DPO training.

3.2 Direct Preference Optimization (DPO)

In the second phase, we fine-tune the SFT model using DPO on the UltraFeedback data set, which contains over 61K training pairs and 2K validation pairs in the format (x, y^+, y^-), where y^+ is the preferred response to prompt x. DPO avoids explicit reward modeling or sampling from a reinforcement learning environment by using a closed-form loss that shifts probability from the less preferred response y^- to the preferred one y^+.

We follow the loss formulation proposed by Rafailov et al. (2023):

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(x,y^+,y^-)} \log \sigma \left(\beta \left[(\log \pi_\theta(y^+|x) - \log \pi_{\text{ref}}(y^+|x)) - (\log \pi_\theta(y^-|x) - \log \pi_{\text{ref}}(y^-|x)) \right] \right)$$

Here, π_θ is the trainable policy model, π_{ref} is the frozen reference model (from SFT), and β is a scalar hyperparameter that controls the KL penalty. In our implementation, we compute log-probabilities over the full tokenized sequence using masked attention, average the result over all non-padding tokens, and backpropagate the DPO loss via gradient accumulation. We experiment with different values of $\beta \in \{0.1, 0.2, 0.3, 0.45\}$ and use effective batch sizes up to 120 to stabilize training. Models are trained using a cosine schedule with learning rates ranging from 10^{-6} to 10^{-7}.

We observe that vanilla DPO is sensitive to these hyperparameters: training is prone to KL collapse or over-regularization in early epochs, particularly when preference differences are subtle or noisy. To address these issues, we introduce curriculum learning into the DPO framework.

3.3 Curriculum-Guided DPO

Curriculum learning is a technique where training examples are introduced in a structured order from easier to harder. This concept, introduced by Bengio et al. (2009), has been shown to improve convergence and robustness in a variety of domains. Inspired by 2D-Curri-DPO, which uses dual metrics for prompt and response difficulty, we simplify this approach by evaluating several scalar difficulty measures independently.

We define three difficulty metrics:

- **Reward Margin (RM):** $|r(y^+) - r(y^-)|$, computed from the Nemotron-70B reward model.
- **Response Length Difference (RDL):** $||y^+| - |y^-||$, to capture stylistic or verbosity preferences.
- **Prompt Complexity (PC):** $\text{StdDev}_{i=1}^{k} \text{PPL}_{\text{PLM}}(y^{(i)}|x)$, using the standard deviation of perplexities across $k = 5$ completions generated by GPT-2. The distribution of these prompt complexities is shown in Figure 4.

Each of these metrics is computed over the full training set prior to DPO. The data is then partitioned into difficulty bins, and we define a per-epoch curriculum schedule to gradually introduce harder samples. In our prompt complexity curriculum, for example, training begins with examples with low perplexity variance and transitions to samples with higher syntactic or semantic complexity.

Our curriculum sampling logic supports multiple modes: epoch-based scheduling, anti-curriculum (hard-to-easy), and additive annealing where difficult examples are injected progressively. We further generalize this to support dynamic β scheduling, with $\beta = \beta_0 - \Delta\beta \cdot t$, where t is the epoch number. This allows the model to adopt a more conservative policy in early epochs and progressively emphasize divergence from the reference model in later ones.

3.4 Training Configuration and Implementation Details

All models are trained on either NVIDIA T4 or H100 GPUs using PyTorch and Hugging Face Transformers. The tokenizer is loaded from Qwen2.5-0.5B with right padding and truncation. Pad tokens are aligned to the model's EOS token, and attention masks are constructed accordingly for log-likelihood computation.

Our DPO training loop supports both standard and gradient-accumulated variants. For our final experiments, we used gradient accumulation over 20 accumulations each with a batch size of 6, which simulates a batch size of 120. We train for 4–4 epochs depending on the configuration. For fair comparison, we validate all models on the same held-out UltraFeedback split (2K examples) and score completions using the LLaMA 3.1 Nemotron-70B reward model. Final results are reported as win rate fold change compared to the SFT-only baseline.

3.5 Summary of Experimental Variants

Across our experiments, we test the following configurations:

- **Vanilla DPO:** $\beta = 0.1$, learning rate=$1e^{-6}$, no curriculum.
- **Hyperparameter Sweep:** Varying $\beta \in \{0.1, 0.2, 0.3\}$, learning rate $\in \{5e^{-7}, \ldots, 1e^{-7}\}$, effective batch size $\in \{60, 120\}$.
- **Curriculum-DPO (Reward Margin):** Sorted data using reward margin, batch size 120.
- **Curriculum-DPO (Prompt Complexity):** Sorted data using perplexity-guided prompt complexity, $\beta_0 = 0.35$, $\Delta\beta = 0.1$, epochs = 3-4.

Our best-performing model used prompt complexity-based curriculum with $\beta_0 = 0.45$, $\Delta\beta = 0.1$, trained for 4 epochs with a batch size of 120. It achieved a win rate fold change of 1.150 over SFT, outperforming all other DPO variants and confirming the benefit of difficulty-aware training schedules.

4 Results

Supervised fine-tuning converged after four epochs of training, with validation loss and perplexity roughly matching training metrics, indicating minimal overfitting.

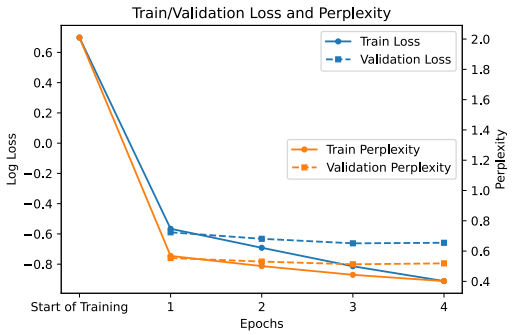

Figure 2: SFT Learning Curves

4.1 Quantitative Evaluation

To evaluate the impact of curriculum learning and hyperparameter tuning on preference optimization, we compare multiple DPO variants against a supervised fine-tuning (SFT) baseline. The primary evaluation metric is win rate fold change, computed using the LLaMA 3.1 Nemotron-70B reward model on held-out UltraFeedback validation prompts. Table 1 summarizes performance across seven configurations, including vanilla DPO, hyperparameter sweeps, and curriculum-guided methods using reward margin and prompt complexity.

Vanilla DPO (Run 1) serves as our base configuration, using $\beta = 0.1$, a learning rate of $1e^{-6}$, and no gradient accumulation. It achieves a win rate fold change of 1.000. Adjusting the learning rate (Run 2) and batch size (Run 3) provides incremental improvements, with Run 3 reaching 1.125. However, these gains taper off without structural changes to data presentation. Runs 5–7 explore curriculum-guided DPO, with prompt complexity emerging as the most robust and stable strategy. Our best model (Run 7) uses $\beta_0 = 0.45$, $\Delta\beta = 0.1$, and a four-epoch curriculum, resulting in a 1.150× win rate gain over the baseline.

These results confirm that curriculum learning can improve alignment even in low-capacity models, and that prompt complexity is an effective signal for structuring preference data. While loss minimization alone is not always predictive of generalization, win rate trends consistently favor models that balance KL penalty growth with progressive exposure to difficult prompts.

Table 1: Performance Comparison Across DPO Variants

DPO Run	Train Loss	Val. Loss	Win Rate Fold Change
1 (Vanilla DPO)	0.840	0.827	1.000
2 ($\beta = 0.1$, lr=$5e^{-7}$)	0.779	0.665	0.964
3 ($\beta = 0.3$, lr=$1e^{-7}$, bs=120)	0.485	0.607	1.125
4 ($\beta = 0.2$, lr=$1e^{-7}$)	0.467	0.619	0.973
5 (Curriculum: RM)	0.394	0.685	1.081
6 (Curriculum: PC, epochs=3)	0.499	0.594	1.081
7 (Curriculum: PC, epochs=4)	0.586	0.604	**1.150**

Overall, we find that while standard DPO responds well to learning rate and batch size adjustments, curriculum-based strategies offer the most reliable path to downstream improvements. Future directions may involve integrating curriculum learning with multi-objective preference optimization and exploring automatic difficulty estimators beyond static metrics.

4.2 Qualitative Analysis

To better understand the behavioral differences between models, we conducted a qualitative comparison on instruction-following tasks requiring semantic fidelity and rephrasing ability. One such prompt asked the model to translate a complex legal sentence into plain, everyday English without losing its original meaning. The example highlights how different training methods affect both literal understanding and communicative clarity (see Figure 2).

The SFT model failed to meaningfully rephrase the sentence, merely echoing the original legalese with no structural or lexical simplification. While technically accurate, this response fails to fulfill the user instruction and reflects the limitations of supervised fine-tuning without preference-based alignment. The DPO model improves marginally by trimming unnecessary phrasing, but its response omits key clauses like "claims, damages, liabilities, costs, and expenses," compromising semantic completeness. Furthermore, the phrasing it retains is syntactically broken ("will covenants and agrees to"), indicating that DPO without a complexity-aware curriculum may over-optimize for brevity at the cost of grammar and precision.

In contrast, the Curriculum-DPO model produces a response that is both human-readable and faithful to the original content. It paraphrases legal terms into plain English constructs ("sign a promise," "protect," "fair and reasonable relationship"), while retaining the legal intent. Additionally, it includes a self-reflective explanation of its simplification choices, demonstrating an emergent behavior of clarity justification. This kind of metacognitive commentary is rare in standard SFT and DPO outputs,

and likely emerges due to structured exposure to varying difficulty prompts during training. While not all added interpretation is strictly accurate ("fair and reasonable relationship" is an inferred clause), the response offers a meaningful trade-off between literal precision and user comprehension.

Across the data set, similar patterns emerged. Curriculum-DPO excelled at handling ambiguous prompts, generating longer-form completions with appropriate structure and tone. It often provided step-by-step breakdowns or rewordings aligned with instruction intent. SFT tended to produce exact restatements of the input or unstructured outputs, while DPO fell somewhere in between – often correct but rather abrupt or underexplained. These trends suggest that curriculum-guided sampling not only stabilizes preference optimization but also encourages more context-aware and user-aligned generation. This example serves as an illustrative case of how curriculum-informed DPO can enhance both the interpretability and quality of model outputs beyond what is visible through numerical metrics alone.

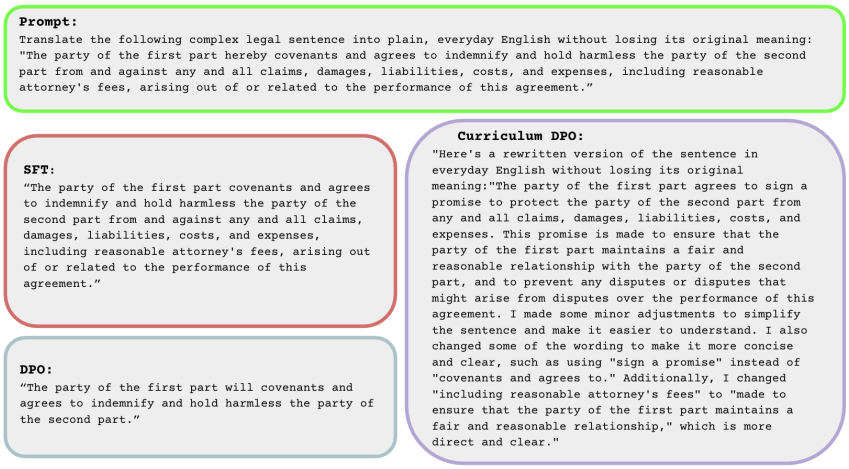

Figure 3: Sample generations on held-out prompt data set

5 Discussion

Our experiments show that curriculum learning offers a consistent and interpretable improvement to Direct Preference Optimization. By structuring training examples from easy to hard, we mitigated early-stage instability and over-regularization, especially in low-learning-rate settings. In our best-performing configuration (Run 7), a prompt complexity-based curriculum with $\beta_0 = 0.45$, $\Delta\beta = 0.1$, and a batch size of 120 achieved a 1.150-fold win rate improvement over the SFT baseline, demonstrating that even simple difficulty heuristics can yield strong gains. Compared to vanilla DPO (Run 1), which achieved a fold change of 1.000, curriculum-guided models converged more efficiently and generalized better to harder prompts. Interestingly, while reward margin–based sorting also helped (Run 5), prompt complexity proved to be the most robust across epochs. This supports the view that syntactic and semantic complexity, as captured by perplexity variance, meaningfully influences alignment behavior. These results suggest that incorporating curriculum into preference optimization is not only effective but also scalable, especially for smaller models with limited capacity to absorb heterogeneous training signals.

6 Conclusion

We show that curriculum learning can significantly improve the stability and performance of Direct Preference Optimization (DPO) by ordering examples based on difficulty. While DPO improves over SFT, it is highly sensitive to hyperparameters. Among several curriculum strategies tested, prompt complexity was the most effective, leading to our best win rate gain of 1.150 times over the vanilla DPO baseline. These results highlight the importance of training dynamics in preference optimization and show that even simple curriculum heuristics can yield meaningful improvements without added architectural complexity.

7 Team Contributions

- **Christopher Sun**: Designed experiments, wrote code, wrote report
- **Abishek Satish**: Designed experiments, wrote code, wrote report

Changes from Proposal We initially attempted to pursue a custom project on tokenization for long-context imitation learning, but then decided to switch to the default project a few weeks in.

References

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*. 41–48.

Wei Li and Jie Zhang. 2025. 2D-Curri-DPO: Dual-Difficulty Curriculum for Preference Optimization. *arXiv preprint arXiv:2504.07856* (2025).

Yining Liu, Zihang Yu, Omer Levy, and Mike Lewis. 2022. Curriculum learning works for pretraining autoregressive transformers. *arXiv preprint arXiv:2212.10561* (2022).

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155* (2022).

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Timo Schick, Oyvind Tafjord, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-Instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560* (2022).

A Additional Experiments

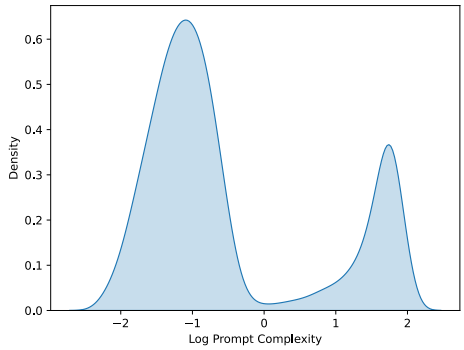

Figure 4: Distribution of log prompt complexities calculated using the methods described in the report