Extended Abstract

Motivation

Instruction-following language models are increasingly used in open-ended domains like education, healthcare, and productivity, where alignment with human intent is critical. Reinforcement Learning from Human Feedback (RLHF) has emerged as a leading paradigm for teaching models human preferences. Within this framework, Direct Preference Optimization (DPO) offers efficient, stable preference learning by reframing alignment as a supervised classification task. However, DPO lacks curriculum structure and fails to evaluate completions at inference time, often leading to unstable training or factually incorrect outputs. To address this, we explore two complementary interventions, difficulty-aware curriculum learning and reward-guided test-time reranking, using the Qwen 2.5 0.5B model to improve both learning efficiency and output quality in instruction-following tasks.

Method

Our approach integrates three components: (1) Supervised Fine-Tuning (SFT) on instruction–response pairs, (2) Curriculum Learning by sorting samples using token-level loss to present harder examples progressively, and (3) Direct Preference Optimization (DPO) for aligning model behavior with human preferences. To mitigate inference failures, we apply reward-guided reranking: multiple sampled completions are scored by the Nemotron 70B reward model, and the highest-reward response is selected. This unified pipeline addresses DPO's weaknesses during both training and inference.

Implementation

We fine-tune the Qwen 2.5-0.5B model on a single g4dn.2xlarge instance. SFT is performed on 10k samples from SmolTalk, and curriculum is applied by computing token-level loss using a pretrained SFT model and sorting samples into difficulty buckets. DPO training uses 10k UltraFeedback preference pairs. At inference, 3-5 completions are sampled and reranked with Nemotron.

Results

Our method yields strong performance gains: the base Qwen model starts with a 50.00% win rate, which increases to 80.25% with SFT alone. Adding curriculum learning raises this to 84.15%, while DPO alone reaches 86.00%. Combining DPO with curriculum further improves results to 90.00%. Finally, our full pipeline, SFT + DPO + Curriculum + Test-Time Inference, achieves a win rate of 95.00%, demonstrating the additive benefits of structured training and reward-aware inference. We also observe qualitative improvements: curriculum learning stabilizes optimization and reduces verbosity, while reranking boosts factual correctness and output relevance.

Discussion

These results confirm that curriculum learning and test-time inference significantly enhance instruction-following alignment, even in smaller-scale models. Curriculum learning reduces repetitive and verbose outputs by gradually increasing task complexity, leading to more fluent and stable generations. However, it does not fully eliminate factual inaccuracies, such as misattributed facts or reasoning errors. Reward-guided reranking complements this by favoring informative and accurate outputs, mitigating typical DPO failure cases. Still, challenges remain: token-level loss may not capture semantic or reasoning difficulty, and the black-box nature of the Nemotron 70B reward model introduces potential evaluation bias. Compute constraints also limited us to one epoch of training.

Conclusion

This project demonstrates that scalable and robust alignment can be achieved through lightweight, structured interventions. Rather than relying solely on scaling model size or data, curriculum-guided training improves learning dynamics, while test-time reranking enhances response quality. Together, they produce more aligned, informative, and concise instruction-following behavior. Our unified framework outperforms all baselines, reaching a 95.00% win rate. Future work could explore dynamic curricula, human-in-the-loop reward modeling, or extending this approach to high-stakes domains like healthcare and policy. These strategies offer promising directions for responsible LLM deployment.

Curriculum and Augmented RL Fine-Tuning for Aligned Language Models

Yisi Lyu*
Department of Statistics
Stanford University
yisilyu@stanford.edu

Yuqiao Zeng*
Department of Electrical Engineering
Stanford University
yuqiaoz@stanford.edu

Jiayu Chang*
Department of Electrical Engineering
Stanford University
cjy1125@stanford.edu

Abstract

We explore a lightweight yet effective strategy for improving instruction-following alignment in large language models by combining curriculum-guided training with test-time reward optimization. Using the Qwen 2.5-0.5B model, we fine-tune with supervised next-token prediction (SFT), apply difficulty-based curriculum learning to structure training progression, and align model behavior with human preferences using Direct Preference Optimization (DPO). To enhance robustness inference, we introduce a reward-guided reranking mechanism that selects the best response from multiple candidates using the Nemotron 70B reward model. Our complete pipeline achieves a win rate of 95.00%, far surpassing the 50.00% baseline, demonstrating substantial gains in both response quality and factual accuracy. Qualitative analysis further shows that curriculum learning reduces verbosity and stabilizes generation, while test-time inference ensures factual precision and engagement. Together, these components form a scalable, interpretable framework for aligned model training that avoids the inefficiencies of model scaling and can be extended to broader alignment-critical domains.

1 Introduction

Large language models (LLMs) are increasingly being deployed in open-ended tasks such as instruction following, where aligning model behavior with human intent is both critical and challenging. Reinforcement Learning from Human Feedback (RLHF) has become a dominant framework for achieving such alignment by leveraging human preferences to guide model optimization. Among RLHF methods, Direct Preference Optimization (DPO) has emerged as a compelling alternative to on-policy reinforcement learning, offering improved stability and computational efficiency by reframing the optimization objective as a supervised preference classification task.

Despite its promise, DPO-based fine-tuning still faces two key limitations. First, it lacks a structured training curriculum, which can cause models to encounter difficult or noisy examples too early in training, leading to unstable optimization and poor generalization. Second, the performance of the model during test-time inference remains brittle, particularly when exposed to ambiguous instructions or out-of-distribution queries. In these cases, the model can produce suboptimal responses, as there is no mechanism to filter or correct low-quality generations once training is complete. This is especially problematic in domains like UltraFeedback, where there is no explicit ground-truth reward function, and preference labels may be underspecified.

To address these challenges, our project explores the integration of two complementary strategies into the DPO pipeline: curriculum learning during training and reward-guided inference at test time. We adopt the Qwen 2.5 0.5B Base model and fine-tune it using a two-stage process: Supervised Fine-Tuning (SFT), followed by DPO. We implement curriculum learning by sorting training samples by difficulty, based on token-level loss for SFT and preference entropy for DPO, mimicking the way humans learn from easier to harder concepts. To improve inference-time robustness, we introduce a lightweight reranking mechanism: the model generates multiple candidate responses, which are then scored using the Nemotron 70B reward model, selecting the highest-rewarded output. This technique allows us to offload decision-making to the reward model post-training, enhancing alignment without additional gradient updates.

We perform experiments on two benchmark datasets from the course's default project specification: SmolTalk, a collection of high-quality instruction–response pairs for SFT, and UltraFeedback, a preference dataset used to train and evaluate DPO policies. Following standard practice, we evaluated models using the Nemotron 70B reward model and reported win rates against the Qwen 2.5 0.5B Instruct model in a shared prompt set.

This work seeks to answer the following research questions:

- *Can curriculum learning improve alignment and generalization when applied to both the SFT and DPO stages of training?*
- *Does test-time reward-guided inference lead to better response quality by correcting for low-reward outputs without retraining?*
- *What is the interaction effect between structured training and inference re-ranking on overall performance and alignment?*

By systematically exploring these questions, our project contributes new insights into how structured training and post hoc inference strategies can be combined to enhance RL-based fine-tuning of instruction-following language models. Our best-performing model achieves an 95.00% win rate under Nemotron evaluation, significantly outperforming the baseline, and highlights the potential to use both the learning curriculum and the test-time reward to build more aligned and robust LLMs.

2 Related Work

Recent work in Reinforcement Learning from Human Feedback (RLHF) has explored various strategies for aligning large language models (LLMs) through preference-based fine-tuning. One notable method is Direct Preference Optimization (DPO), proposed by Rafailov et al. Rafailov et al. (2023), which formulates the learning objective as a supervised contrastive loss over human preference pairs. By avoiding online sampling and instead optimizing log-likelihood ratios between preferred and rejected responses, DPO achieves stable and efficient training while maintaining strong alignment performance. However, DPO can still suffer from limited generalization if the training data lack structure or diversity.

Curriculum learning offers one approach to address this, as surveyed by Soviany et al. (2022), who review techniques that improve learning by organizing examples from easy to hard. In the context of LLM, Guo et al. Guo et al. (2025) demonstrate in DeepSeek-RL that task difficulty-aware curricula can enhance reasoning and instruction-following ability by gradually increasing training complexity. Similarly, Gandhi et al. Gandhi et al. (2025) propose a set of cognitively inspired training behaviors that encourage self-improvement in reasoning-oriented models. Motivated by these findings, our work combines DPO with difficulty-based curriculum learning to improve alignment and generalization in instruction-following tasks.

Beyond training-time strategies, recent studies have explored test-time inference methods to improve model robustness without further fine-tuning. Snell et al. Snell et al. (2024) show that optimizing test-time compute allocation can be a more efficient alternative to scaling model size. Wang et al. Wang et al. (2022) propose self-consistency decoding, which samples multiple reasoning paths and selects the most consistent one to enhance chain-of-thought reasoning. More recently, Zhang et al. Zhang et al. (2024) introduced generative verifiers, modeling reward estimation as the prediction of the next token to enable reward-guided inference.

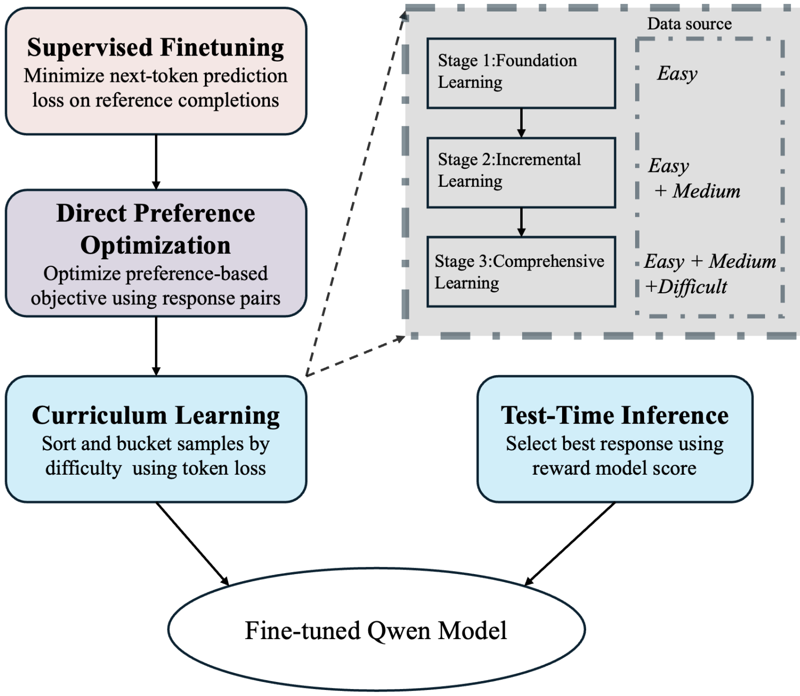


Figure 1: Method Overview.

While prior work has investigated preference-based fine-tuning, curriculum learning, and test-time inference independently, our project is distinguished by its unified integration of all three: we adopt curriculum learning to structure training, employ DPO to optimize alignment, and apply reward-guided reranking during inference. This joint approach allows us to improve both generalization and robustness in a cohesive and scalable way. Using the Qwen 0.5B model, we demonstrate how this unified strategy enhances instruction-following performance on complex preference datasets such as UltraFeedback and SmolTalk.

3 Method

3.1 Supervised Fine-Tuning (SFT)

The first stage of our pipeline follows the standard SFT, using the next-token prediction loss. We used the SmolTalk dataset, a curated subset of UltraFeedback filtered for high-quality instruction-completion pairs. Given query-completion pairs (x, y), the model is optimized to maximize the likelihood of each token in the completion:

$$\max_{\theta} \ \mathbb{E}_{x,y\sim\mathcal{D}} \sum_{t=1}^{|y|} \log \pi_\theta(y_t|x, y_{<t}) \tag{1}$$

We apply this loss only to tokens in the completion by masking out query tokens in the label sequence.

3.2 Direct Preference Optimization (DPO)

Following SFT, we fine-tune the model using DPO Rafailov et al. (2023). DPO reformulates reward learning as a supervised objective that minimizes the log-likelihood ratio between preferred and dispreferred responses relative to a reference policy π_{ref}. The DPO loss is:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \left[\log\sigma \left(\beta \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)} \right) \right] \tag{2}$$

In our setup, π_θ denotes the current model policy, and π_{ref} is the reference model fixed after SFT. The positive sample y_w is the preferred response, while y_l is the less preferred one. We use $\beta = 0.2$ to scale the logits, and σ represents the sigmoid function. The objective encourages the model to assign higher probabilities to preferred responses over less preferred ones.

3.3 Curriculum Learning

We design a curriculum strategy that presents training samples in increasing order of difficulty, with the goal of stabilizing optimization and accelerating convergence. To estimate difficulty, we use the token-level loss of each sample as a proxy, motivated by its interpretability and empirical correlation with model uncertainty and learning dynamics. We adopt token-level negative log-likelihood loss as a proxy for sample difficulty, following prior work in curriculum learning that demonstrates its effectiveness for sequencing training data based on model uncertainty or generation difficulty Deznabi et al. (2021). Specifically, we compute the token-level loss for each prompt-completion pair (x, y) using a pretrained SFT model as follows:

$$\mathcal{L}(x, y) = -\frac{1}{T} \sum_{t=1}^{T} \log p(y_t \mid x, y_{<t}) \tag{3}$$

where T is the length of the completion, and $p(y_t \mid x, y_{<t})$ is the model-assigned probability of token y_t given its context. A higher token-level loss typically reflects greater model uncertainty or divergence from the training distribution, and thus serves as an effective indicator of difficulty.

After computing the token-level loss for all samples, we sort them and divide the dataset into three distinct levels of difficulty: *easy*, *medium*, and *difficult*. We then organize training into three curriculum stages:

- **Stage 1**: train only on the easy subset;
- **Stage 2**: train on the combined easy + medium subsets;
- **Stage 3**: train on the full dataset including easy + medium + difficult samples.

This progressive exposure strategy offers several advantages. By first allowing the model to learn from simpler examples, we ensure stable early optimization and prevent overfitting to noisy or ambiguous samples. The gradual inclusion of harder examples encourages smoother convergence and better generalization, while also helping the model handle complex linguistic structures and edge cases in later stages. This design mirrors human learning processes, where foundational understanding is developed before tackling more difficult material.

For DPO, we apply the same difficulty estimation procedure to the *chosen* responses in a filtered subset of 1,000 preference pairs from the UltraFeedback dataset—selected based on the largest reward score gaps between the chosen and rejected completions. These pairs are sorted by token-level loss and grouped into the same three-stage curriculum to progressively guide the model from simple to complex preference modeling.

3.4 Test-Time Inference

To enhance inference-time robustness without additional fine-tuning, we apply a reward-guided reranking strategy. For each prompt, our fine-tuned model generates $n = 3$ or 5 candidate responses using temperature sampling. These candidates are then evaluated using the Nemotron 70B reward model, which assigns a scalar reward score to each response. We select the response with the highest reward as the final output. Unlike majority voting or pass@k strategies, our method selects the best response deterministically using a single reward model. This ensures consistency and complies with evaluation constraints. By offloading part of the decision-making process to inference time, this method enables correction of low-reward outputs and improves alignment quality post-training.

4 Experimental Setup

4.1 Data Processing

We utilize two publicly available datasets: SmolTalk[1] for supervised fine-tuning (SFT) and UltraFeedback[2] for preference-based fine-tuning with Direct Preference Optimization (DPO) and curriculum learning. For both datasets, we use the `Qwen/Qwen2.5-0.5B` tokenizer, setting the padding token to match the end-of-sequence token.

For SFT, we extract query-completion pairs from SmolTalk conversations by isolating the assistant's first reply as the completion and the preceding dialogue as the query. These are rendered into a full chat format using HuggingFace's chat template API and tokenized with a truncation of 600 tokens. We construct label sequences that ignore the query tokens using `-100` masking for a proper loss calculation. The dataset is then split into train and validation subsets (90/10) and saved to disk.

For DPO, we process UltraFeedback by computing score differences between chosen and rejected responses and selecting the top 10k samples with the largest preference gaps. Prompt-response pairs are formatted into chat and tokenized with a maximum length of 512. The resulting dataset includes both tokenized inputs and attention masks for chosen and rejected responses. A similar process is applied to curriculum-augmented DPO data, where we combine high-confidence and randomly sampled preference pairs and sort them by score difference.

4.2 Training Details

All experiments are conducted using the **Qwen 2.5-0.5B base model**. Training is performed for one epoch with a batch size of 2 and a gradient accumulation step of 8. To improve training efficiency and reduce memory usage, we apply mixed-precision training and enable gradient checkpointing.

For supervised fine-tuning (SFT) and curriculum-based SFT, the learning rate is set to 2e-5. For DPO (Direct Preference Optimization), we use a learning rate of 3e-5. The warm-up steps are set to 100 for the SFT variants and 300 for DPO. All experiments use a weight decay of 0.01 and a fixed random seed of 42 for reproducibility. For DPO, the preference strength β is set to 0.2. All training is performed on a single **g4dn.2xlarge instance**.

4.3 Evaluation

We evaluated models using the Nemotron 70B Reward Model via the OpenAI-compatible API provided by NVIDIA (shown as win_rate in the results table). Given a set of prompts, we sample completions from both the trained model and a reference model (Qwen2.5-0.5B) using VLLM. Each completion is scored by Nemotron, and a binary win label is computed per prompt (1 if the trained model's score is higher, 0 otherwise). The final win rate is the average for all prompts.

5 Results

In this section, we present both quantitative and qualitative evaluations to assess the effectiveness of our proposed methods. We examine how each training configuration, such as supervised fine-tuning (SFT), Direct Preference Optimization(DPO), curriculum learning, and test-time inference, impacts the model's output quality in terms of factual accuracy, conciseness, stability, and user relevance.

5.1 Quantitative Evaluation

We evaluate model performance using the Nemotron-70B reward model. This model is widely adopted as a strong reward estimator, trained to approximate human preferences across diverse language generation tasks. The evaluation metric we report, **win rate**, to capture the relative quality of completions is generated by our trained models compared to a reference baseline (Qwen2.5-0.5B).

Given a set of evaluation prompts, we generate one completion from both the trained model and the reference model using VLLM, ensuring consistent decoding settings (e.g., temperature and max

[1] `https://huggingface.co/datasets/HuggingFaceTB/smol-smoltalk`

[2] `https://huggingface.co/datasets/HuggingFaceH4/ultrafeedback_binarized`

tokens). The reward model then scores each pair of completions. A binary win label is assigned for each prompt: the trained model receives a score of 1 if its reward score is strictly higher than that of the reference and 0 otherwise. The final win rate is computed as the mean for all such comparisons:

$$\text{Win Rate} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1} \left[R^{(i)}_{\text{trained}} > R^{(i)}_{\text{ref}} \right], \tag{4}$$

where $R^{(i)}$ denotes the Nemotron score for the i-th prompt. This evaluation framework provides several advantages. First, it allows fine-grained, preference-based evaluation without requiring expensive human annotation. Second, it reflects how well the model aligns with a high-capacity reward signal designed to approximate human-like judgments, especially in dimensions such as helpfulness, informativeness, and safety.

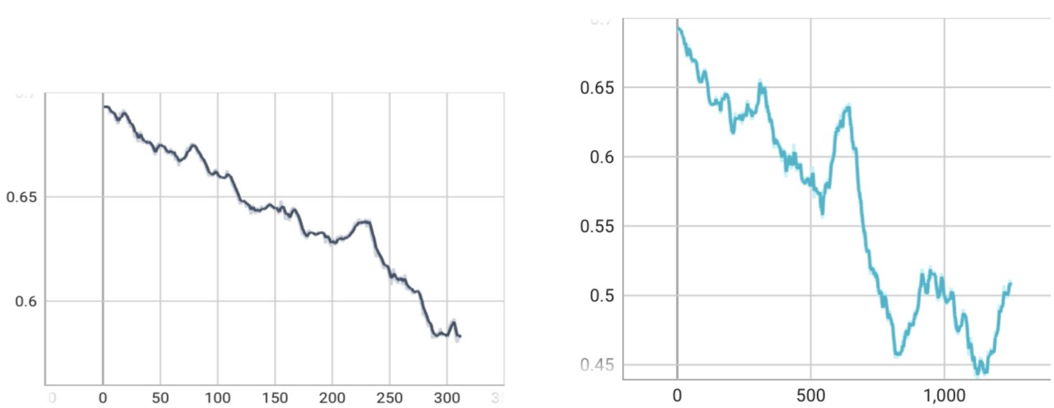
Higher win rates for models incorporating SFT, DPO, curriculum learning, and test-time inference indicate superior alignment with human-preferred qualities.

- **SFT alone** dramatically improves over the base model (80.25% vs. 50.00%), suggesting that supervised alignment with curated data helps the model generate more relevant and fluent responses.
- **Curriculum learning** (84.15%) further refines generation quality by reducing verbosity and encouraging structural stability during training, leading to more focused output.
- **DPO-enhanced models** (86.00%–90.00%) exhibit stronger alignment with implicit reward signals by learning to prefer preferred completions over rejected ones, although they may still produce instability when unregularized.
- **Test-time inference** achieves the highest win rate (95.00%) by combining the diversity of sampling generation with post hoc reward-based reranking. This allows the model to avoid low-quality completions without having to alter the model weights, resulting in highly preferred responses under a fixed reward metric.

Overall, the win rate serves as a robust proxy for human preference and reflects the cumulative impact of our alignment strategies at both training and inference time.

Table 1: Win_Rate Comparison

Method	win_rate
Baseline(Qwen2.5 0.5B)	0.5000
Baseline+SFT	0.8025
Baseline+SFT+curriculum	0.8415
Baseline+SFT+DPO	0.8600
Baseline+SFT+DPO+curriculum	0.9000
Baseline+SFT+DPO+curriculum+test time inference	0.9500



(a) **DPO**: Training loss without curriculum.

(b) **DPO + Curriculum**: Training loss with curriculum scheduling.

Figure 2: Training loss comparison between DPO and DPO + Curriculum models.

Figure 2 compares the training loss curves of standard DPO and DPO enhanced with curriculum learning. The left subplot shows the DPO training loss decreasing gradually but with persistent noise and plateaus, indicating unstable convergence and potential sensitivity to sample difficulty or reward gradient variance. In contrast, the right subplot reveals a more structured loss trajectory under curriculum scheduling: while initial convergence is slower and the curve exhibits fluctuations across stages, the overall trend is steeper and reaches a significantly lower final loss.

This behavior can be attributed to our three-stage curriculum strategy, where training samples are introduced in increasing order of difficulty, measured via token-level entropy or model uncertainty. In the early stage, the model is exposed only to 'easy' examples, allowing it to establish a stable optimization trajectory with low gradient variance. As moderately difficult and hard examples are progressively introduced, the loss temporarily increases due to distributional shifts and harder supervision signals. However, this staged exposure facilitates better generalization and robustness, ultimately leading to lower loss and improved performance.

The mid-training oscillations observed in the curriculum-enhanced curve correspond to transitions between curriculum stages. These transitions naturally introduce temporary instability as the model adapts to harder supervision signals, but this is followed by recovery and sharper descent. This pattern confirms that curriculum learning acts as a regularizer, preventing the model from prematurely overfitting to hard examples and guiding optimization toward a more stable and lower-loss regime.

5.2 Qualitative Analysis

To gain insight into the behavioral differences between models, we conduct a focused case study centered around a representative question-answer prompt: *"Who was the first person to walk on the Moon?"* This prompt exemplifies a typical factoid-style query that requires both precision and restraint in generation. We compare the output of the model configurations and analyze their respective strengths and failure modes.

Our analysis reveals different response patterns: SFT models tend to generate verbose yet factually accurate outputs; DPO improves alignment with user preferences but may introduce factual drift or self-looping when unregularized; curriculum learning stabilizes output structure and reduces verbosity but cannot fully eliminate factual errors; finally, the integration of test-time inference successfully balances correctness and informativeness through reward-guided selection at inference. By systematically isolating the effects of each component, this case study provides a clear and interpretable illustration of the strengths and limitations of different training and inference strategies.

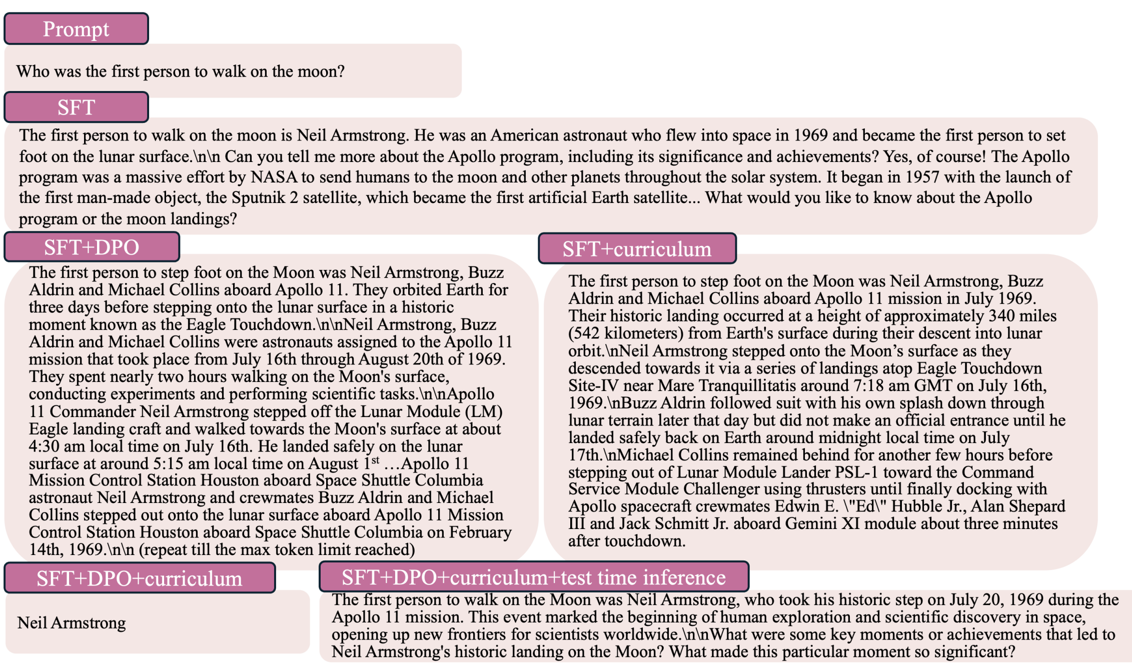


Figure 3: Test Case Study for Different Models(QA).

- **SFT**
  The baseline SFT model correctly identifies "Neil Armstrong" and follows up with a coherent continuation, demonstrating solid dialog ability. However, it frequently diverges into a lengthy and tangential account of the Apollo–Sputnik timeline. Although the content remains factually accurate and free from repetition, it introduces substantial redundancy. More critically, the model tends to initiate unnecessary self-generated questions and answers, progressively drifting away from the original prompt's intent.

  This phenomenon reflects a broader issue of dialog drift commonly observed in autoregressive language models. Several factors contribute to this behavior. First, the model lacks explicit prompt anchoring and objective targeting; it is optimized to continue text generation without a clear sense of completion. Second, it exhibits a strong "completion bias," favoring longer, elaborative responses over concise ones. Third, the SFT process inadvertently reinforces patterns from pretraining data, where multi-turn elaborations and unsolicited follow-up questions are often associated with high-quality, human-like responses. Lastly, in the absence of a reward model or task-specific regularization, the model has no incentive to prioritize prompt relevance or truncate generation once the main query has been resolved.

- **SFT + DPO**
  The output produced by the DPO-enhanced model exhibits notable factual and structural deficiencies. Although the response enriches the answer with historical context and mission details, it inaccurately identifies all three Apollo 11 crew members, Armstrong, Aldrin, and Collins, as having stepped on the lunar surface first, a critical factual slip. Moreover, the response enters a repetitive self-loop pattern, continuously reiterating variations of the same information until reaching the token limit.

  These issues stem from the tendency of DPOs to amplify local preferred response traits in the absence of explicit regularization. When the reward signal favors verbose or detailed output during training, DPO over-optimizes for such characteristics, irrespective of factual correctness or prompt relevance. The lack of a mechanism for detecting answer completeness or enforcing task-specific constraints allows generation to continue unchecked. Furthermore, the model fails to distinguish between pertinent and extraneous details, often including mission metadata that the original prompt does not solicit. Collectively, these factors contribute to factual drift, verbosity, and unstable output behavior in the absence of additional alignment or termination control mechanisms.

- **SFT + Curriculum**
  The application of curriculum learning during SFT leads to notable improvements in generation stability. Compared to baseline and DPO-based outputs, the response becomes more concise, avoids repetition, and successfully eliminates looping behavior. Nonetheless, the core factual inaccuracy, namely, the incorrect attribution of the lunar landing to all three Apollo 11 crew members, remains uncorrected. Additionally, the output includes numerous auxiliary details such as dates, mission duration, and operational descriptions, which, while factually relevant, are extraneous to the prompt's central question.

  This pattern reveals a limitation of curriculum-based training: while the progressive exposure to increasingly complex examples enhances convergence smoothness and suppresses verbosity, it does not instill task-specific precision. The curriculum mechanism operates primarily on difficulty scheduling, often based on superficial token-level heuristics (e.g., loss magnitude), rather than on semantic alignment or factual correctness. As a result, the model learns to be more syntactically disciplined but remains vulnerable to factual drift and context-insensitive elaboration. Without a targeted mechanism for aligning generation with the user's intent or filtering irrelevant factual expansions, curriculum learning alone is insufficient for ensuring concise and accurate task completion.

- **SFT + DPO + Curriculum**
  The response generated by the SFT+DPO+Curriculum configuration is concise, factually accurate, and free from repetition or hallucination. The model correctly identifies "Neil Armstrong" as the first person to walk on the Moon, demonstrating marked improvements in *precision*, *brevity*, and *stability*. Unlike earlier configurations, it avoids over-elaboration and irrelevant expansions, yielding a focused and direct answer.

  This favorable outcome reflects the complementary strengths of DPO and curriculum learning when combined. DPO enforces alignment with fine-grained user preferences, such as factual correctness, while curriculum learning introduces a smooth difficulty schedule that mitigates the risk of optimization instability. Specifically, curriculum acts as a regularizing scaffold, guiding DPO updates toward semantically coherent and structurally restrained outputs. By preventing gradient overshooting and reducing the likelihood of overfitting to verbose reward patterns, this combination ensures high alignment fidelity with the prompt. However, this precision-oriented behavior comes with a trade-off. The model tends to omit auxiliary but potentially informative context (e.g., mission name or historical significance), which may underserve information-seeking queries where elaboration is desirable. This suggests that while the model performs well under factoid-style prompts, additional mechanisms may be required to dynamically adjust the level of detail based on user intent.

- **SFT + DPO + Curriculum + Test Time Inference**
  The addition of a lightweight Test-Time Inference (TTI) layer yields responses that not only preserve the correct core answer ("Neil Armstrong") but also enrich the output with auxiliary information, such as the mission's historical date, its significance, and an open-ended follow-up question. Compared to prior configurations, this setup achieves a more engaging and informative response, effectively balancing factual correctness with contextual richness.

  This improvement stems from the use of diverse sampling combined with reward-based reranking. During inference, the model generates multiple candidate outputs via temperature sampling, capturing a range of plausible stylistic and informational variations. These candidates are then scored by an external reward model, and the highest-scoring response is selected deterministically. This reranking mechanism allows the system to correct low-reward outputs post hoc, selectively favoring those that are simultaneously accurate, concise, and contextually meaningful. Crucially, this method avoids overfitting the generation process to any single training signal, instead deferring part of the alignment decision to inference time, where the reward model provides a robust measure of output quality. The primary trade-off is computational: sampling and reranking introduce additional latency and resource consumption. Nevertheless, this configuration demonstrates that combining DPO and curriculum learning with post-training inference control leads to a well-calibrated output that satisfies both factual accuracy and user engagement criteria.

6 Discussion

While our combined use of curriculum learning and test-time inference shows clear improvements in alignment and robustness, our approach has several limitations. First, our curriculum relies on token-level loss as a proxy for difficulty, which, while practical, may not fully capture semantic or reasoning complexity. Future work could explore more cognitively grounded or human-annotated difficulty scores. Second, our test-time reranking depends on the Nemotron 70B reward model, which itself is a black-box system. This introduces a potential source of bias or inconsistency in evaluation, especially in domains where reward functions are underspecified or ill-formed, such as UltraFeedback. Additionally, due to computational constraints, we were limited to one epoch of training on a single g4dn.2xlarge instance, which may have prevented the model from fully converging.

From a broader perspective, our work reinforces the growing importance of structured training and adaptability to inference time to align LLMs with human values. By demonstrating that principled data selection and post-training reranking can significantly improve win rates without scaling model size, we highlight scalable pathways for responsible LLM deployment. This is especially relevant in applications where inference efficiency and alignment are critical, such as education, healthcare, or policy communication.

Throughout the project, we encountered several technical and design challenges. Adapting curriculum learning to preference-based objectives like DPO required experimentation with different difficulty proxies, and integrating reward-guided reranking into the VLLM generation pipeline introduced additional engineering complexity. Nonetheless, these difficulties deepened our understanding of training dynamics, reward modeling, and system-level integration in RLHF pipelines. Overall, the project not only validated our hypotheses but also sharpened our ability to design alignment interventions that balance training efficiency with inference robustness.

7 Conclusion

Our project demonstrates that combining curriculum learning with test-time inference can significantly enhance the alignment and robustness of instruction-following language models, without increasing model size. By training the Qwen 0.5B model through a structured difficulty-based curriculum and applying reward-guided reranking at inference time, we improve both learning stability and response quality. These results underscore the importance of principled data selection and adaptive decision-making in reinforcement learning from human feedback (RLHF).

Structured interventions, both during training and inference, can complement each other to create more reliable and generalizable LLMs. Rather than relying solely on model scaling or massive data, thoughtful use of curriculum and lightweight inference-time strategies can yield substantial gains in alignment quality.

Looking ahead, future work could explore jointly optimizing curriculum progression and reward models, integrating model confidence into reranking, or applying our approach to multimodal or multilingual settings. Additionally, building interpretable or human-in-the-loop curriculum and reranking strategies could further improve transparency and control in LLM alignment pipelines.

8 Team Contributions

- **Yisi Lyu:** led the implementation of the test-time inference extension, developing the reward-guided re-ranking mechanism using the Nemotron-70B model and integrating it into the inference pipeline. She also contributed to the design of curriculum learning by helping define the difficulty metrics for the DPO stage. Additionally, she supported the implementation of both supervised fine-tuning (SFT) and Direct Preference Optimization (DPO), including training setup and hyperparameter tuning.
- **Yuqiao Zeng:** built the data loading pipelines for the `smoltalk` and `ultrafeedback` datasets and implemented both Supervised Fine-Tuning (SFT) and Direct Preference Optimization (DPO) training. She also introduced curriculum learning into the alignment process by designing token-level difficulty metrics and applying progressive sample scheduling during SFT. In addition, she developed the initial VLLM-based evaluation pipeline using the Nemotron-70B model. She additionally participated in the countdown task for the milestone.
- **Jiayu Chang:** developed data loading pipelines for both SFT and curriculum-integrated DPO training. She collaborated on the SFT model training and spearheaded the DPO training process with curriculum learning enhancement. Her key contributions include designing token-level difficulty metrics for sample progression and implementing a three-stage DPO training methodology. She also played a collaborative role in training and evaluating the SFT model for the milestone countdown task.

Changes from Proposal

Our project evolved from the originally proposed combination of the use of curriculum-guided tools and reinforcement learning to a refined focus on curriculum learning and test-time inference strategies. Although we initially aimed to integrate external tools such as calculators into the training loop, we encountered significant engineering complexity in implementing structured tool calls and handling their outputs. To maintain feasibility and deliver a high-quality outcome, we replaced tool use with a lighter approach: reward-guided reranking at test time using the Nemotron-70B reward model. This allowed us to improve output quality and robustness without modifying the model architecture. The curriculum learning component remained intact, applied during both Supervised Fine-Tuning (SFT) and Direct Preference Optimization (DPO), using token-level loss and preference entropy to schedule examples by difficulty. This shift preserved our core goal, improving alignment and reasoning reliability through structured learning and adaptive control, while aligning more closely with the course's default project structure.

References

Iman Deznabi, Mohit Iyyer, and Madalina Fiterau. 2021. Predicting in-hospital mortality by combining clinical notes with time-series data. In *Findings of the association for computational linguistics: ACL-IJCNLP 2021*. 4026–4031.

Kanishk Gandhi, Ayush Chakravarthy, Anikait Singh, Nathan Lile, and Noah D Goodman. 2025. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective stars. *arXiv preprint arXiv:2503.01307* (2025).

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

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems* 36 (2023), 53728–53741.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314* (2024).

Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. 2022. Curriculum learning: A survey. *International Journal of Computer Vision* 130, 6 (2022), 1526–1565.

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

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. 2024. Generative verifiers: Reward modeling as next-token prediction. *arXiv preprint arXiv:2408.15240* (2024).