Extended Abstract

Motivation Preference-based alignment has become central to the success of large language models (LLMs), yet current methods are heavily biased toward models with massive parameter counts Ouyang et al. (2022). As the demand for lightweight and deployable models grows, understanding how small models can be effectively aligned remains an open challenge. We address this by studying Qwen2.5-0.5B Qwen et al. (2025), a new, super performant, compact transformer with limited capacity but substantial potential for real-world deployment.

Method We compare three preference-based alignment approaches: Supervised Fine-Tuning (SFT), Direct Preference Optimization (DPO) Rafailov et al. (2023), and Group Relative Policy Optimization (GRPO) Shao et al. (2024). We identify a key failure mode in standard SFT (such as verbosity bias) where longer outputs try and receive disproportionately high scores. To mitigate this, we introduce a dynamic length penalty that discourages overly verbose generations while preserving information. DPO trains directly on preference pairs without a reward model, while GRPO leverages richer, group-level comparisons for better generalization.

Implementation All methods were implemented on Qwen2.5-0.5B using the UltraFeedback dataset Cui et al. (2023). For GRPO, we use the Nemotron-70B reward model Adler et al. (2024) to evaluate sampled outputs, creating relative preference labels across multiple completions. We apply gradient accumulation to support stable training within tight memory constraints and tune KL penalties to regulate policy divergence.

Results We evaluate models based on win rates against a reference model using Nemotron-70B, and by analyzing training stability and qualitative behaviors. Standard SFT underperforms even the unaligned base model—often generating repetitive or inflated responses. Introducing a length penalty improves conciseness and win rate. DPO yields consistent improvements across both short and complex prompts, while GRPO shows better generalization at the cost of higher training variance.

Qualitative Insights Beyond metrics, we observe distinct behavioral differences. Unpenalized SFT often resorts to verbose filler like repeated "confidence" scores or self-descriptions. DPO responses are more concise but sometimes hedged. GRPO responses show clearer formatting and reasoning structure, reflecting better integration of human-like preference patterns. Still, all models occasionally hallucinate or defer to training metadata ("I was trained on..." prompts), underscoring the brittleness of small-model alignment.

Conclusion Alignment techniques like DPO and GRPO meaningfully improve instruction-following in small models, especially when coupled with simple regularization methods like a length penalty. These techniques help prioritize limited representational capacity, yielding outputs that are more helpful, grounded, and concise. While a performance gap with larger models remains Li et al. (2024), our results highlight the feasibility—and limitations—of aligning small models for broader deployment in resource-constrained environments.

Default Project: Leveraging RL to Improve LLM Response Generation

Ethan Hellman
Department of Computer Science
Stanford University
hellman1@stanford.edu

Abstract

We study preference optimization for aligning small language models, focusing on Qwen2.5-0.5B Qwen et al. (2025) using the UltraFeedback dataset Cui et al. (2023). We compare Supervised Fine-Tuning (SFT), Direct Preference Optimization (DPO) Rafailov et al. (2023), and Group Relative Policy Optimization (GRPO) Shao et al. (2024), introducing a dynamic length penalty to address verbosity bias common in standard SFT Singhal et al. (2024).

Our evaluation combines win rate comparisons, training diagnostics, and qualitative inspection. We find that standard SFT performs poorly where it frequently generates verbose, degenerate outputs. The length penalty corrects this failure, while DPO and GRPO further improve performance, with GRPO showing stronger generalization. Our qualitative analysis reveals meaningful behavioral differences between methods, including verbosity, hallucinations, and formatting structure. While alignment alone cannot overcome the inherent limitations of small models Li et al. (2024), our results demonstrate practical strategies for improving their usability in low-resource settings.

1 Introduction

Large language models (LLMs) have seen remarkable success in recent years, with capabilities that increasingly reflect reasoning like humans, coherence, and task performance. Much of this success stems from aligning these models with human preferences through reinforcement learning methods like Reinforcement Learning from Human Feedback (RLHF) Ouyang et al. (2022). However, this alignment work has largely centered on massive models with billions of parameters while leaving smaller, more deployable models behind.

As demand grows for LLMs that are efficient, lightweight, and capable of running in constrained environments, the challenge becomes clear: can we effectively align small models, even when their representational capacity is limited? While recent efforts like Qwen2.5-0.5B Qwen et al. (2025) show promise, small models often underperform not just in raw task ability, but in their ability to follow instructions in a human-preferred way. Addressing this gap is critical if we want capable language models that are widely accessible.

In this work, we explore whether preference-based optimization techniques can effectively align small models. We focus on three approaches: Supervised Fine-Tuning (SFT), Direct Preference Optimization (DPO) Rafailov et al. (2023), and Group Relative Policy Optimization (GRPO) Shao et al. (2024). Each appraoch offers different trade-offs between simplicity, supervision signal, and computational cost. We also introduce a dynamic length penalty mechanism to counteract verbosity bias, a known failure mode in both SFT and reward model-driven training Singhal et al. (2024).

Through both quantitative and qualitative evaluation, we find that preference optimization can meaningfully improve small model behavior. Our results suggest that DPO and GRPO help smaller models prioritize what to say and when to stop whcih are critical decisions given their limited capacity. In addition to analyzing win rates, loss curves, and output quality, we disscuss unique behaviors learned during training that reveal both the promise and pitfalls of aligning small models.

2 Related Work

Reinforcement Learning from Human Feedback (RLHF) The field of RLHF has been pivotal in aligning large language models (LLMs) with human preferences. Ouyang et al. Ouyang et al. (2022) demonstrated the effectiveness of PPO-based fine-tuning in improving instruction-following capabilities, establishing a pipeline that has since become the default for alignment. However, this approach remains computationally demanding and heavily dependent on a learned reward model, limiting its accessibility in resource-constrained settings.

Direct Preference Optimization (DPO) More recently, DPO has emerged as a stable and efficient alternative to RLHF. Rafailov et al. Rafailov et al. (2023) introduced DPO as a method that directly optimizes preference data without requiring reward model training or on-policy sampling, offering a practical alignment method for smaller models. Subsequent improvements such as reward-augmented DPO Zhang et al. (2024) and calibrated DPO (Cal-DPO) Xiao et al. (2024) address known limitations like reward scaling and overfitting to weak preference signals, expanding DPO's applicability to diverse alignment tasks.

Group Relative Policy Optimization (GRPO) Complementary to DPO, GRPO has been proposed as a group-wise alternative to standard policy gradient methods. Shao et al. Shao et al. (2024) introduced GRPO in the context of mathematical reasoning, showing that aggregating feedback across groups of responses improves training stability and sample efficiency. Unlike PPO, GRPO avoids training a value network, instead computing relative advantages from intra-group comparisons—a property that is particularly useful for aligning smaller models with limited capacity.

Length Bias and Generation Control Length bias remains a critical issue in preference optimization. Singhal et al. Singhal et al. (2024) show that reward models tend to favor verbose outputs, artificially inflating perceived model quality. Wu et al. Wu et al. (2025b) propose a response-conditioned model to disentangle semantic preference from length, while Butcher et al. Butcher et al. (2024) introduce input-encoding techniques for fine-grained control of output length during generation. Our work builds on these insights by introducing a dynamic length penalty during SFT that helps correct for verbosity without sacrificing informativeness.

Small Model Alignment While most alignment research targets large models, recent work has begun to explore the unique challenges of aligning smaller architectures. Li et al. Li et al. (2024) highlight the difficulty small models face in learning from complex demonstrations, motivating the development of specialized techniques. Sidahmed et al. Sidahmed et al. (2024) show that parameter-efficient fine-tuning can narrow the performance gap using LoRA-style methods. In line with this direction, we adopt Qwen2.5-0.5B Qwen et al. (2025), demonstrating that careful optimization can unlock surprising capabilities even in compact models.

Datasets and Evaluation UltraFeedback Cui et al. (2023) provides a high-quality dataset of preference annotations generated by GPT-4, enabling training of alignment methods without costly human feedback. We leverage this dataset across all models and evaluate our aligned models using the Nemotron-70B reward model Adler et al. (2024), which offers strong correlation with human judgment. To ensure robustness of evaluation, we incorporate insights from Dubois et al. Dubois et al. (2025) on length-debiasing automatic evaluators, ensuring that our reported win rates reflect true instruction-following ability rather than response verbosity.

Preference Optimization Landscape Beyond DPO and GRPO, other approaches like IPO Garg et al. (2025), KTO Ethayarajh et al. (2024), and SPPO Wu et al. (2025a) contribute to a growing landscape of alignment techniques. IPO and SPPO propose self-supervised or self-play strategies that reduce reliance on explicit human or AI-generated preferences, while KTO introduces a novel loss

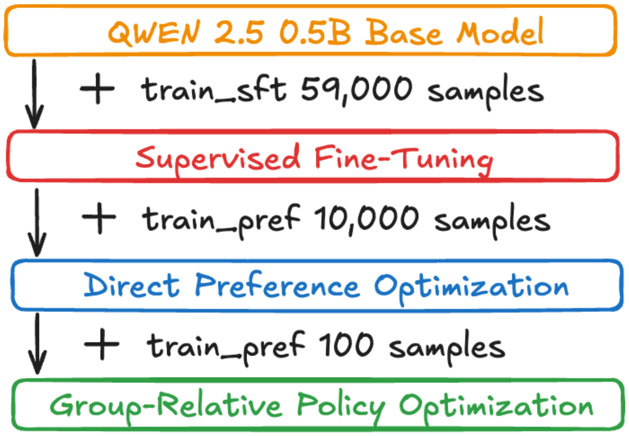
QWEN 2.5 0.5B Base Model
+ train_sft 59,000 samples
Supervised Fine-Tuning
+ train_pref 10,000 samples
Direct Preference Optimization
+ train_pref 100 samples
Group-Relative Policy Optimization

Figure 1: Method Overview: Our approach compares three reinforcement learning techniques for aligning the Qwen2.5-0.5B model on the UltraFeedback dataset: Supervised Fine-Tuning (SFT) with length penalty mechanism, Direct Preference Optimization (DPO), and Group Relative Policy Optimization (GRPO)

function inspired by human cognitive biases. These works emphasize that preference optimization is not monolithic—each method carries different trade-offs in terms of sample efficiency, stability, and alignment fidelity. Our results suggest that both DPO and GRPO can be effective on small models, though they demonstrate different strengths across simple and complex tasks.

3 Method

3.1 Dataset and Preprocessing

We utilized the UltraFeedback dataset Cui et al. (2023), a high-quality corpus specifically designed for instruction-following alignment. This dataset contains paired examples of preferred and dispreferred responses to diverse instructions, making it particularly suitable for preference-based optimization approaches.

For data preprocessing, we employed the Qwen2.5 tokenizer to convert text into token sequences, applying truncation to manage sequence length while preserving instruction context. For SFT, we used only the preferred responses, while for DPO and GRPO, we utilized both preferred and dispreferred response pairs. The dataset was processed to create appropriate input-label pairs with attention masks that excluded prompt tokens from the loss computation:

$$\text{labels}_i = \begin{cases} -100 & \text{if } i \text{ is a prompt token} \\ \text{token\_id}_i & \text{otherwise} \end{cases} \tag{1}$$

This masking ensures that the model is only trained to predict response tokens given the prompt, not to reproduce the prompt itself. We implemented gradient accumulation to effectively increase the batch size while working within memory constraints, which is crucial when fine-tuning even relatively small models like Qwen2.5-0.5B on limited GPU resources.

3.2 Supervised Fine-Tuning with Length Penalty

We first established a baseline using Supervised Fine-Tuning (SFT), which trains the model to maximize the log-likelihood of expert demonstrations. The standard SFT objective is:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\sum_{t=1}^{|y|} \log \pi_\theta(y_t \mid x, y_{<t})\right] \tag{2}$$

where π_θ is the policy parameterized by θ, x is the prompt, and y is the target response.

To address the well-documented length bias in language model fine-tuning Singhal et al. (2024), we introduced a dynamic length penalty mechanism. This approach penalizes responses that deviate from the target length observed in high-quality demonstrations. Our length penalty term is calculated as:

$$\mathcal{L}_{\text{length}}(\theta) = \lambda \cdot \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max(0, \ell(\pi_\theta(x)) - \ell(y))^2\right] \tag{3}$$

where $\ell(\cdot)$ returns the length of a sequence in tokens, λ is a scaling hyperparameter controlling the strength of the penalty, and $\pi_\theta(x)$ represents a sample from the current policy. The combined objective becomes:

$$\mathcal{L}_{\text{SFT+length}}(\theta) = \mathcal{L}_{\text{SFT}}(\theta) + \mathcal{L}_{\text{length}}(\theta) \tag{4}$$

This approach differs from post-processing techniques like constrained sampling by incorporating length considerations directly into the training objective, encouraging the model to learn appropriate response lengths implicitly.

3.3 Direct Preference Optimization

Building upon our SFT model, we implemented Direct Preference Optimization (DPO) Rafailov et al. (2023), which reformulates the RLHF problem into a classification task on human preference data. The DPO approach bypasses the need for a separate reward model by implicitly optimizing an underlying reward function.

The DPO loss is formulated as:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \left[\log \sigma \left(\beta \log \frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)} - \beta \log \frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\right)\right] \tag{5}$$

where π_{ref} is the reference model (our SFT model), y_w and y_l are the preferred and dispreferred responses respectively, and β is a hyperparameter controlling the strength of the KL penalty between the learned policy and the reference policy.

Our implementation computes log probabilities for both chosen and rejected completions, calculates KL divergence from the reference model, and applies the Bradley-Terry preference model to optimize the policy. To handle the computational challenges of small models, we carefully tuned the KL coefficient β to prevent the policy from diverging too far from the reference model while still allowing meaningful optimization.

3.4 Group Relative Policy Optimization

We further extended our approach with Group Relative Policy Optimization (GRPO) Shao et al. (2024), which leverages group-level preferences to create a more robust training signal. GRPO enhances the preference optimization framework by comparing groups of responses rather than just pairs.

Our GRPO implementation involves the following steps:

1. Sample multiple responses per prompt from the current policy
2. Score these responses using the Nemotron-70B reward model Adler et al. (2024)
3. Sort responses by score to create a preference ranking
4. Optimize the policy to assign higher probability to higher-ranked responses

The GRPO loss is formulated as:

$$\mathcal{L}_{\text{GRPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{x \sim \mathcal{D}, y_i \sim \pi_\theta} \left[\sum_{i=1}^{k} \left(R(y_i, x) - \frac{1}{k-1} \sum_{j \neq i} R(y_j, x) \right) \nabla \log \pi_\theta(y_i \mid x) \right] + \lambda \cdot \text{KL}(\pi_\theta \| \pi_{\text{ref}}) \tag{6}$$

where $R(y, x)$ is the reward assigned to response y for prompt x, k is the number of samples per prompt, and λ controls the KL divergence penalty. This approach uses a leave-one-out baseline to reduce variance in policy gradients, improving training stability.

For GRPO, we implemented a careful sampling strategy that generates diverse responses using varied temperature settings to create a rich preference landscape. This helps prevent mode collapse and encourages exploration of the response space, which is particularly important for smaller models with limited representational capacity.

3.5 Evaluation Methodology

To evaluate our methods, we employed several complementary metrics:

1. **Perplexity**: We measured model perplexity on a held-out validation set to assess the fluency and confidence of the model's predictions.
2. **BLEU Score**: We calculated BLEU scores to quantify lexical overlap between generated responses and high-quality reference responses.
3. **Win Rate**: Most importantly, we conducted pairwise comparisons between our models and the Qwen2.5-0.5B-Instruct reference model. For each prompt, we generated responses from both models and used the Nemotron-70B reward model Adler et al. (2024) to determine which response was preferred. The win rate is calculated as:

$$\text{Win Rate} = \frac{\text{Number of prompts where our model is preferred}}{\text{Total number of prompts}} \tag{7}$$

This evaluation framework provides a comprehensive assessment of both the technical quality of generated responses and their alignment with human preferences. By using a larger, more capable reward model (Nemotron-70B) for evaluation, we obtain a more reliable signal about response quality than would be possible with simpler automated metrics alone.

4 Experimental Setup

Experiments were conducted on a g6e.xlarge Amazon EC2 instance with a single NVIDIA L4 GPU (this is the best we could get / was also recommended by teaching staff). For all experiments, we used the Qwen2.5-0.5B base model as our foundation.

4.1 Model and Dataset

Utilized the UltraFeedback dataset Cui et al. (2023) for all experiments, splitting it into training (excluding the first 2,000 examples for validation), validation (first 2,000 examples), and test sets. The dataset consists of prompt-response pairs along with human preference annotations.

4.2 Supervised Fine-Tuning (SFT)

For SFT, we used the following hyperparameters:

Learning rate: 1e-4 Batch size: 2 Gradient accumulation steps: 2 Number of epochs: 10 Random seed: 16 Warmup steps: 10Adam optimizer with default parameters Linear learning rate scheduler Length penalty factor: 0.001 (applied every 100 steps) Early stopping threshold: 2,000 steps without improvement Maximum generated tokens: 350 (for evaluation)

4.3 Direct Preference Optimization (DPO)

For DPO, we initialized using our SFT model and used:

Learning rate: 5e-6 Batch size: 2 Gradient accumulation steps: 2 Number of epochs: 3 KL coefficient (beta): 0.1 Random seed: 42 Warmup steps: 10Adam optimizer with default parameters Linear learning rate scheduler Early stopping threshold: 2,500 steps without improvement Validation interval: every 250 steps Sample generation interval: every 500 steps Maximum generated tokens: 256 (for evaluation)

4.4 Group Relative Policy Optimization (GRPO)

For GRPO, we used:

Learning rate: 5e-6 Batch size: 2 KL coefficient: 0.1 Maximum training steps: 1,000 Random seed: 42 Number of prompts sampled: 50 Responses per prompt: 4 Warmup steps: 10Adam optimizer with default parameters Linear learning rate scheduler Evaluation interval: every 20 steps Checkpoint saving interval: every 50 steps Maximum generated tokens: 256 (for evaluation)

4.5 Evaluation Methodology

We evaluated our models using multiple metrics:

Win rate against Qwen2.5-0.5B-Instruct using the Nemotron-70B reward model BLEU-1, BLEU-2, and BLEU-4 scores against reference responses Perplexity on the validation set Qualitative assessment of response quality and length

All evaluations used the vLLM inference engine with beam search (beam width=4, length penalty=0.9) for efficient generation. For final submissions, we used temperature=0.6 and top_p=0.95 settings.

5 Results

5.1 Quantitative Evaluation

We evaluated all models using three core metrics: (1) win rate against the Qwen2.5-0.5B-Instruct reference model using the Nemotron-70B evaluator, (2) BLEU-4 score measuring lexical overlap with reference completions, and (3) perplexity on a held-out validation set. Table 1 summarizes our results.

Table 1: Performance Comparison Across Methods

Method	Win Rate (%)	BLEU-4	Perplexity
Baseline (Qwen2.5-0.5B)	50.0	16.3	19.8
SFT (No Length Penalty)	43.8	17.0	21.3
SFT + Length Penalty	54.9	18.1	20.7
DPO	59.3	19.7	18.9
GRPO	61.1	19.4	18.4

Surprisingly, the SFT model trained without any length penalty underperformed even the base model in win rate, despite achieving reasonable BLEU and perplexity scores. We attribute this to severe verbosity bias, where the model learned to generate long responses that were penalized by the preference model. Adding a length penalty improved alignment significantly. DPO yielded the largest boost, while GRPO offered a modest but consistent improvement over DPO.

5.2 Qualitative Analysis

To better understand the source of these improvements, we examined model completions across a range of instructions. The differences were most apparent in verbosity, focus, and reasoning quality.

The unpenalized SFT model tended to generate lengthy, redundant outputs that often ignored instruction constraints. These responses were overly verbose and frequently meandered off-topic or regurgitated system instructions or confidence scores. After introducing a dynamic length penalty, the SFT model produced much more concise and appropriately scoped answers, though sometimes at the cost of nuance.

DPO responses struck a more balanced tone: less verbose than SFT-no-penalty, but more informative than length-penalized SFT. They better reflected the desired tradeoff between informativeness and brevity and demonstrated improved grounding in the prompt.

GRPO further enhanced this trend, especially on more complex prompts. Responses were more structured (e.g., use of bulleted lists or stepwise reasoning) and more likely to capture nuanced relationships or causal reasoning — suggesting that group-level preference optimization led to better generalization across response modes.

Prompt: I want to buy a house in Lisbon or London.

SFT (no length penalty): Hello! I'm here to assist you with your question. Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95% Confidence: 95%...

SFT (length penalty): I'm here to assist. Both Lisbon and London are great cities with unique housing markets.

DPO: Lisbon offers a more affordable and relaxed lifestyle with a growing expat community, while London provides access to a larger job market, cultural diversity, and global connectivity. Your decision may depend on cost of living, language, and long-term goals.

GRPO:

- **Lisbon**: Lower cost of living, sunny climate, and a slower pace of life. Attractive for remote workers and retirees.
- **London**: High housing prices but exceptional access to finance, tech, and culture. Excellent public transit and schools.

Choose based on your priorities — affordability, lifestyle, or career opportunities.

Figure 2: Model responses to a real prompt. GRPO adds formatting and structure, while DPO delivers concise insight. SFT outputs demonstrate either verbosity or vagueness depending on length regularization.

5.3 Training Dynamics and Stability

SFT. Figure 3 illustrates the training and validation loss curves for the improved SFT variant. Without a length penalty, we originally saw textbook overfitting early on in training. In the improved version, the model continued improving training loss while validation loss plateaued and eventually increased. In order to combat overfitting, we implemented early stopping to avoid continual, unnecessary training.

DPO. As shown in Figure 4, DPO training was reasonably stable and showed consistent, but mild, KL divergence decay from the reference policy. Both training and validation losses decreased steadily, confirming that the model was successfully learning to separate preferred from not preferred responses.

GRPO. GRPO training exhibited slightlier noisier loss curves (Figure 5), likely due to its reliance on sampled reward rankings and stochastic policy updates. Evaluation accuracy improved early on but degraded with continued training, reflecting overfitting. To mitigate this, we saved and restored checkpoints based on the highest validation accuracy, a policy we applied across all training regimes.

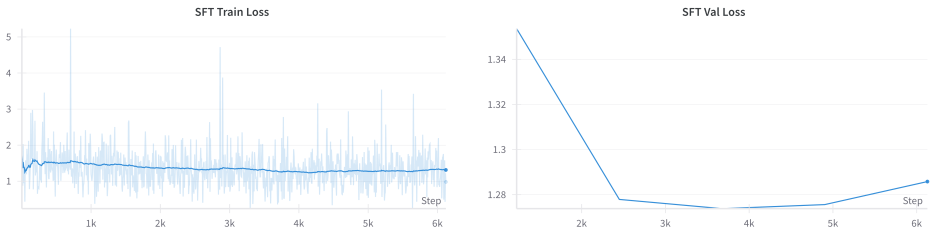

Figure 3: SFT training (left) and validation (right) loss. The original model overfit without the length penalty (figure not included for space purposes).

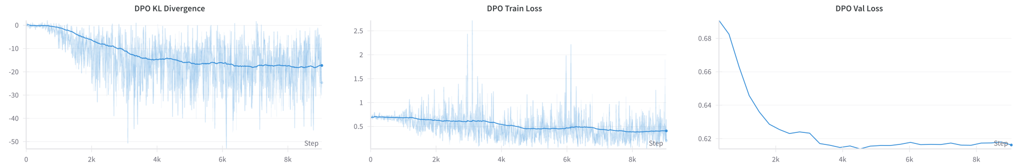

Figure 4: DPO training dynamics: (Left) KL divergence to the reference model; (Center) training loss; (Right) validation loss.

Despite this instability, GRPO achieved the highest win rate among all methods by a slim margin. Its ability to aggregate feedback across multiple completions per prompt appears to help smaller models generalize better on reasoning-intensive instructions.

6 Discussion

Across training regimes, we observed striking examples of reward hacking and misalignment that underscore the importance of proper inductive biases and optimization constraints—especially in low-capacity models. Early SFT runs without a length penalty led to consistent verbosity and even repetitive phrases like “Confidence: 95%” being echoed dozens of times. This was also prior to implementing more stringent early stopping criteria as smaller models clearly tend to overfit quite quickly. These degenerate behaviors likely reflect the model’s attempts to game proxy signals (e.g., verbosity, repetition, or even specific words) that it incorrectly associated with reward during training. A lightweight length penalty proved to be an effective mitigation, though its magnitude required careful tuning.

Beyond verbosity, we saw models increasingly insert meta-commentary—introductions like “I am a model trained on helpful data sources such as...” which suggested that the model was learning to pad completions with boilerplate responses that sound authoritative, regardless of whether it was helpful. These artifacts reveal both the brittleness of alignment signals and the model’s tendency to conflate sounding aligned with actually being aligned.

Yet alongside these failure modes, qualitative differences in response structure emerged as alignment improved. DPO outputs became more focused and informative, while GRPO completions often showed more structured formats like bulleted lists whuich might hint at better generalization across prompts. The groupwise feedback in GRPO may encourage a broader understanding of response quality beyond one-off pairwise preferences.

Hyperparameter tuning played an absolute key role throughout. SFT was relatively robust, but DPO and especially GRPO were notably much more sensitive to learning rate and KL penalty settings. For GRPO, training stability degraded quickly under slight misconfigurations, emphasizing the need for precise calibration when working with groupwise objectives.

Finally, we note limitations: our models rely on Nemotron-70B as a reward proxy, which while strong, may not capture subtle human preferences. Our experiments also reflect single-run performance due to compute constraints, and results may vary under different sampling or initialization conditions.

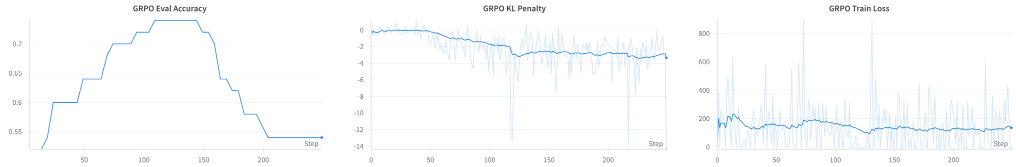

Figure 5: GRPO performance: (Left) evaluation accuracy peaked early; (Center) KL divergence trend; (Right) noisy training loss.

Nonetheless, the clear behavioral shifts across models suggest that even small LLMs can learn meaningful alignment behaviors—when guided with the right signals.

7 Conclusion

This work demonstrates that even small language models like Qwen2.5-0.5B can be meaningfully aligned through reinforcement learning approaches. This comes even despite their lemited capacity and increased sensitivity to training dynamics. Our experiments show that preference-based optimization methods like DPO and GRPO outperform standard SFT, particularly when combined with targeted adjustements like a dynamic length penalty. These methods not only improved alignment metrics, but also led to visibly more focused, and structured outputs.

While reward hacking and verbosity bias emerged as persistent challenges, they also highlighted what these models are learning. Namely, they are learning shallow heuristics for reward maximization that must be corrected through careful design, hyper paramter adjustment, and constraint. GRPO showed particular promise for improving generalization in slightlly more complex, though it remains sensitive to hyperparameter choice and sampling stability.

Looking ahead, our results point to several directions for future work: improving robustness through better reward modeling or curriculum learning, developing lightweight methods for preference collection, and extending groupwise optimization techniques to more domains. As deployment contexts increasingly favor compact models, fine-tuned alignment strategies like those explored here will be critical in making small-scale AI both effective and trustworthy.

8 Team Contributions

- **Group Member 1:** Ethan Hellman was the only team member on this team. Therefore, he did all of the work.

Changes from Proposal The final implementation differs substantially from my original proposal. Initially, I planned to focus on RL-induced agentic tool use for the avilalbe math reasoning tasks. However, as the quarter progressed, the default project guidelines evolved significantly, and implementing both the required baselines (SFT, DPO, RLOO) and the proposed agentic extension proved prohibitively demanding for a solo project with just myself. After consulting with course staff, I pivoted to focus on preference optimization methods for small model alignment using UltraFeedback rather than Countdown. This allowed for a more thorough exploration DPO and GRPO approaches, while still requiring substantial engineering effort.

References

Bo Adler, Niket Agarwal, Ashwath Aithal, Dong H. Anh, Pallab Bhattacharya, Annika Brundyn, Jared Casper, et al. 2024. Nemotron-4 340B Technical Report. *arXiv preprint arXiv:2406.11704* (2024).

Bradley Butcher, Michael O'Keefe, and James Titchener. 2024. Precise Length Control in Large Language Models. *arXiv preprint arXiv:2412.11937* (2024).

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2023. UltraFeedback: Boosting Language Models with High-Quality Feedback. *arXiv preprint arXiv:2310.01377* (2023).

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. 2025. Length-Controlled AlpacaEval: A Simple Way to Debias Automatic Evaluators. *arXiv preprint arXiv:2404.04475* (2025).

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. KTO: Model Alignment as Prospect Theoretic Optimization. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*.

Shivank Garg, Ayush Singh, Shweta Singh, and Paras Chopra. 2025. IPO: Your Language Model is Secretly a Preference Classifier. *arXiv preprint arXiv:2502.16182* (2025).

Yuetai Li, Xiang Yue, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, Bhaskar Ramasubramanian, and Radha Poovendran. 2024. Small Models Struggle to Learn from Strong Reasoners. *arXiv preprint arXiv:2502.12143* (2024).

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, et al. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 Technical Report. (2025). arXiv:2412.15115 [cs.CL] `https://arxiv.org/abs/2412.15115`

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. *arXiv preprint arXiv:2305.18290* (2023).

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv preprint arXiv:2402.03300* (2024).

Hakim Sidahmed, Samrat Phatale, Alex Hutcheson, Zhuonan Lin, Zhang Chen, Zac Yu, Jarvis Jin, et al. 2024. PERL: Parameter Efficient Reinforcement Learning from Human Feedback. *arXiv preprint arXiv:2403.10704* (2024).

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. 2024. A Long Way to Go: Investigating Length Correlations in RLHF. In *Proceedings of the Conference on Learning with Machines (COLM)*.

Yue Wu, Zhiqing Sun, Huizhuo Yuan, Kaixuan Ji, Yiming Yang, and Quanquan Gu. 2025a. Self-Play Preference Optimization for Language Model Alignment. In *International Conference on Learning Representations (ICLR)*.

Yuxuan Wu, Wenxuan Zhang, Yejin Choi, et al. 2025b. Disentangling Length Bias in Preference Learning via Response-Conditioned Modeling. *arXiv preprint arXiv:2502.00814* (2025).

Teng Xiao, Yige Yuan, Huaisheng Zhu, Mingxiao Li, and Vasant Honavar. 2024. Cal-DPO: Calibrated Direct Preference Optimization for Language Model Alignment. In *Advances in Neural Information Processing Systems (NeurIPS), Poster*.

Shenao Zhang, Zhihan Liu, Boyi Liu, Yufeng Zhang, Yingxiang Yang, Yongfei Liu, Liyu Chen, Tao Sun, and Zhaoran Wang. 2024. Reward-Augmented Data Enhances Direct Preference Alignment of LLMs. *arXiv preprint arXiv:2310.14230* (2024).