Abstract

Motivation. At the core of today's large language models' success lies their ability to understand and reliably act on user instructions. These models are expected to adapt seamlessly to a wide variety of topics and tasks, consistently responding in a relevant, helpful, and coherent way. Meeting the needs of users also involves meeting each of these criteria and more, often including stylistic formatting preferences, factual accuracy, and even the "personality" or confidence of the response. To continuously improve a model's performance along these criteria, traditional reinforcement learning methods like Direct Preference Optimization (DPO) rely on binary user preferences between two comparable outputs, in order to assign a singular scalar reward for preferred outputs.

Incorporating many aspects of users' preferred or preferred responses into a singular scalar reward naturally comes at the cost of granularity in the specific aspects that may contribute to a better or worse response. To address this limitation, multi-objective reinforcement learning uses each desired response criteria as a unique reward signal, enabling optimization across multiple dimensions simultaneously rather than utilizing a single collapsed reward value. Recent work has shown that this multi-objective approach can preserve diversity of outputs and increased pluralistic alignments with a variety of human values (Sorensen et al. (2024), Meister et al. (2024), Singh et al. (2025)).

Method and Implementation. In this paper, we begin with the open-weight Qwen2.5-0.5B as our base model for all finetuning and preference optimization tasks. We first perform supervised fine-tuning (SFT) on the model using the Smoltalk dataset, in order to further solidify the model's understanding of conversational speech patterns. We then optimize this checkpoint via DPO on the Ultrafeedback dataset, which includes prompts and "better" or "worse" responses that reflect the relative ability of the response to follow the prompt's instructions. Finally, we use the SFT and DPO checkpoint as the baseline for our multi-objective experiments, where we aim to balance fluency and helpfulness. Fluency is scored by a dis-fluency reward model (4i/ai/BERT_disfluency_cls) and helpfulness is scored by a reward model designed to align with human preferences (OpenAssistant/reward-model-deberta-v3-large-v2). With these two separate reward scores, we use a weighted aggregation of these two reward heads to scale the loss between "better" and "worse" responses accordingly. This approach incentivizes the model to generate responses that represent an overall qualitative improvement of our model's responses along these criteria.

Results. We evaluated our models using the Llama 3.1 Nemotron 70B reward model across 200 held-out test prompts. The SFT model performed comparably to the Qwen2.5 baseline, with a win rate of 48.50%, but showed signs of overfitting to QA-style outputs. DPO yielded the strongest improvement, achieving a win rate of 60.50% and the highest average reward. Our multi-objective model, which jointly optimized for helpfulness and fluency, achieved a moderate improvement with a 52.00% win rate. These results demonstrate that while SFT alone does not yield significant improvements, both DPO and multi-objective reinforcement learning can enhance instruction-following ability—with DPO achieving the most consistent gains, and multi-objective optimization offering a promising path toward more balanced response generation.

Discussion & Conclusion. Although we had to alter our initial implementation, our method for implementing a multi-objective reward function provided modest results. Future improvements could be through implementing more robust models, such as RLOO or GRPO. In all, these results show the potential of multi-objective methods for advancing instruction-following capabilities in LLMs.

Instruction Following with Qwen2.5: Multi-Objective Reinforcement Learning

Team Members: Gabrielle Belanger, George Dimopoulos, Nahome Hagos
Emails: gcbel@stanford.edu, gdimop@stanford.edu, nhagos@stanford.edu

Abstract

We explore the use of multi-objective reinforcement learning to improve instruction-following behavior in Qwen2.5-0.5B by explicitly optimizing for helpfulness and fluency. Starting with supervised fine-tuning on the Smoltalk dataset, we apply preference-based optimization using the Ultrafeedback dataset. We then combine separate reward models to score each response on helpfulness and fluency into a single reward to guide model updates. Our method enables more targeted improvements across multiple behavioral dimensions rather than collapsing preferences into a single scalar value.

However, our limited set of objectives—restricted to just helpfulness and fluency—may have constrained the model's ability to generalize, and likely contributed to its underwhelming quantitative performance compared to DPO, which indirectly captured a broader notion of preference via a single learned signal. Additionally, while we used a simple scalarized loss function to integrate our reward signals, there is significant potential for more advanced reinforcement learning methods—such as Pareto front optimization or constrained policy updates—to better leverage multi-objective reward structures and achieve improved alignment across competing dimensions.

1 Introduction

For Large Language Models, being able to follow instructions across a variety of tasks is crucial in ensuring models are widely helpful and applicable to diverse use cases. Users increasingly expect coherent, helpful responses that accurately capture the task at hand. In improving model performance in tasks like instruction following, traditional reinforcement learning approaches typically use a singular reward head – a metric that is often a measure overall user preference. While this approach has been shown to be successful (Rafailov et al. (2023)), our team was interested in exploring a more granular approach to improving model outputs: multi-objective reinforcement learning.

In this paper, we explore existing research and an end-to-end approach for improving response helpfulness and fluency through dedicated reward heads. In this exploration, we set out to determine whether replacing the aggregate reward during optimization with a multi-objective paradigm with two reward heads would more closely model how humans may judge response quality.

To explore this question, we optimize Qwen2.5-0.5B, a 0.5B parameter open-weight model through supervised fine-tuning, Direct Preference Optimization (DPO), and explore our primary extension of multi-objective reinforcement learning. From the original four parameters we selected to comprise our multi-objective reward function (helpfulness, factuality, fluency, and format clarity), we narrowed our scope to two. In particular, we selected helpfulness and fluency, as both contribute to user satisfaction with a response and identifying a reward model for measuring factuality proved to be challenging.

2 Related Work

This project builds on the work of Sorensen et al. (2024), Meister et al. (2024), and Singh et al. (2025), three foundational papers that helped motivate and inform our interest in multi-objective reinforcement learning.

Sorensen et al. (2024) argue for the importance of pluralistic alignment of model outputs with the diversity of human thought. The paper argues that existing single-objective models tend to reduce variation of output and that multi-objective RL is crucial in capturing a broader spectrum of human values, perspectives, and preferences. This work laid crucial groundwork for our extension, but did not explicitly define a training method for how to implement optimization of several objectives simultaneously in practice.

Meister et al. (2024) describe distributional alignment, which involves a model's ability to accurately represent a diverse range of opinions. The authors find that existing models primarily succeed at describing a distribution of opinions, rather than recreating and simulating a variety of opinions, indicating that current RLHF approaches reduce diversity across responses. Meister et al. primarily focus on evaluation of existing methods via prompt steering, and does not focus explicitly on optimizing multiple objectives during training. In this way, this paper works towards a different goal than we are aiming for: the authors focus on the ability of models to recreate specific opinions, whereas our project seeks to research multi-objective optimization to emphasize multiple desirable traits (like fluency and factual correctness) in output quality.

Singh et al. (2025) present Few-Shot Preference Optimization (FSPO), a novel technique that allows an LLM to be personalized for a user based on labeled preference pairs. FSPO uses these labeled pairs to generate a unique scalar reward for each user, enabling the model to better represent minority users preferences in output responses. This work aims to align responses with human values, but in a fundamentally different way than our proposal: FSPO focuses on scalar rewards tied to an individual user, and our approach optimizes via several aggregated unique reward sources. Thus, this paper does not address the key focus of our extension in rewarding multiple metrics simultaneously to improve model output.

Our extension builds on this motivating groundwork and expands on how the values described in these papers could be implemented in practice to create a holistic scoring process. In particular, this paper defines an end-to-end approach to align specific desired qualities (for our purposes, helpfulness and fluency) with the output of a model's responses. In an alternate setting, this training method could be applied with other reward functions that measure different qualities (diversity of opinion, factuality, or other aspects of model preference). Therefore, this extension aims to propose one method of training that could be applicable with aligning outputs with specific user or researcher desires.

3 Method

3.1 Supervised Fine-Tuning

We first finetuned Qwen2.5 using the Smoltalk dataset, which contains dialog-style instruction-following samples designed to simulate conversational speech patterns. The goal of this stage was to adapt the base model to produce more natural, user-aligned language outputs. We adapted hyperparameters from existing literature on finetuning Qwen models and used a learning rate of 2e-5 with weight decay set to 0.01, and optimized using Adam with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e{-}8$. We trained for 3 epochs using a batch size of 4 and gradient accumulation steps of 8 (yielding an effective batch size of 32).

We observed that training loss dropped rapidly early in training, which led us to incorporate a linear warmup over the first 500 steps and an exponential decay schedule to stabilize convergence. Flash attention was disabled due to compatibility issues with the model architecture. We used mixed-precision training cautiously and opted out of AMP to avoid instability in early runs. Checkpoints were saved at the end of every epoch, and training logs were recorded every 50 steps. While the model showed clear adaptation to conversational tone, later qualitative analysis revealed overfitting patterns in its tendency to answer all prompts in a QA format, regardless of instruction complexity or intent—likely due to dataset bias.

3.2 Direct Preference Optimization (DPO)

To further improve Qwen2.5's performance on the instruction following task, we optimized our SFT checkpoint with Direct Preference Optimization (DPO) on the UltraFeedback binarized dataset Rafailov et al. (2023). We use our SFT Qwen2.5 checkpoint as:

- A frozen reference model p_{ref} to provide baseline log-likelihood.
- A trainable model p_θ, which we train and compute updates on to incentivize better completions.

We obtain the pairwise reference data for the instruction following task, in the form of "better" and "worse" labeled responses from the Ultrafeedback binarized dataset.

At each training step we compute the log likelihood of the better and worse responses (y_{better} and y_{worse} respectively) with respect to a given prompt x on both models p_θ and p_{ref}. We then calculate the difference in the log-likelihood ratios:

$$-\log \sigma \left([\log p_\theta(y_{\text{better}}|x) - \log p_{\text{ref}}(y_{\text{better}}|x)] - [\log p_\theta(y_{\text{worse}}|x) - \log p_{\text{ref}}(y_{\text{worse}}|x)] \right).$$

We then use this to incentivize p_θ the p_{ref} to prefer the better output, and perform a gradient update on p_θ.

In this training, we use a batch size of 4, a beta of 0.1, learning rate of 5e-6 for 3 epochs with 3000 batches selected from the dataset, which is shuffled between epochs.

3.3 Multi-Objective Reinforcement Learning

We further train Qwen 2.5-0.5B to enhance the model's performance on the instruction following task via a multi-objective optimization framework that explicitly balances two behavioral dimensions: helpfulness and fluency.

This approach involves the construction of separate reward signals for each objective. For each generated output, we compute a per-objective score s_i and define the gap g_i relative to a predefined ideal $\hat{s}_i$ as:

$$g_i = |s_i - \hat{s}_i|,$$

where i indexes the behavioral objectives.

The reward sources are as follows:

- **Helpfulness:** Scored using `OpenAssistant/reward-model-deberta-v3-large-v2`, a preference reward model trained on human feedback that evaluates how well a model's response aligns with helpful intent.
- **Fluency:** Scored using `4i/ai/BERT_disfluency_cls`, a BERT classifier that predicts the fluency of a response by identifying patterns in generated text which are not fluent.

To reflect the varying importance of each objective, we apply criticality weighting to each gap:

$$g_i^{\text{eff}} = w_i \cdot g_i,$$

where w_i represents the criticality weight assigned to objective i (which, for our experiments, involved emphasizing helpfulness and fluency equally).

Rather than aggregating these effective gaps into a scalarized loss via a Soft Chebyshev formulation as initially proposed, our final implementation directly converts the weighted sum of gaps into a scalar reward:

$$r = -\sum_i w_i \cdot |s_i - \hat{s}_i|$$

This reward then modulates the log-likelihood of the generated response under the current model policy. Specifically, we optimize the following loss:

$$\mathcal{L} = -\log p_\theta(y|x) \cdot r,$$

where $p_\theta(y|x)$ is the probability of the generated output y given input x, and r is the reward computed from the behavioral scores. This approach treats high-reward outputs as more desirable and updates the model parameters to increase their likelihood.

While our original intent was to implement Direct Preference Optimization, we encountered practical difficulties with DPO's implementation and data requirements. Consequently, we opted for this simpler reward-weighted policy gradient approach to directly optimize for helpfulness and fluency.

Furthermore, we omit brevity as an explicit optimization objective in initial experiments, due to the difficulty of defining a static ideal brevity target across diverse prompts. Future extensions may incorporate brevity optimization conditioned on prompt type or use an LLM-based evaluator for more flexible brevity assessment.

4 Experimental Setup

4.1 Data

For our experiments, we used two datasets:

- The Smoltalk dataset (Allal et al. (2025)), which was used for our supervised finetuning, includes over 460,000 conversations between a user and an LLM, of which we used 10% as a held-out validation set. For instance:

  User: "You are a gamekeeper at a medieval village in 14th century England. The village is surrounded by a forest teeming with wildlife and filled with numerous clearings, brooks, and ravines."
  Assistant: "Good morrow to thee, stranger. 'Tis a grand day to be outdoors, ain't it? The sun shines bright, and the forest is full of life. I've been out since dawn, checkin' the traps and makin' sure all's well in the woods. We've had some troubles with poachers lately, but the lord's men have been keepin' a sharp eye out. What brings thee to our humble village? Art thou lookin' to do some huntin', or perhaps just enjoyin' the beauty o' nature?"

- The binarized Ultrafeedback dataset (Cui et al. (2023)), which was used for DPO and our multi-objective extension, contains over 60,000 prompts with annotated better and worse responses. For instance:

  User: "List five web frameworks."
  Response 1: "As an AI language model, I don't have a subjective opinion but here are five popular web frameworks: 1. React 2. Angular 3. Laravel 4. Django 5. Ruby on Rails"
  Response 2: "1. Django 2. Ruby on Rails 3. Laravel 4. AngularJS 5. ReactJS"

4.2 Task Description

Instruction following involves ensuring that a model can produce coherent, helpful responses across many types of questions and instructions from users. Our baseline, Qwen2.5-0.5B was specifically pre-trained to well across various instruction following tasks, making it well suited to this exploration. This task-specific improvement will be centered around reinforcement learning methods on the Ultrafeedback dataset. For instance, here are two sample instruction following prompts we expect our model to perform well on:

- Convert this HTML document to a LaTeX document, ensuring all formatting and mathematical expressions are properly translated. Write your response in a code block.
- Create a function to sort an array of floating-point numbers in descending order.

5 Results

5.1 Quantitative Evaluation

To test the performance of our respective efforts, we used the Llama 3.1 Nemotron 70B Reward Model to measure whether the baseline Qwen or our optimized models performed better across 200 test prompts.

Model	Avg Reward	Avg Reward Diff from Qwen2.5	Win Rate against Qwen2.5
Qwen2.5-0.5B	-24.07	–	–
SFT	-25.00	-0.93	48.50
DPO	-22.91	+1.16	60.50
Multi-Obj	-24.53	-0.46	52.00

Note: Due to the difficulties processing leaderboard submissions, all above numbers reflect the results of local evaluation scripts.

Model	Leaderboard Submissions
SFT	gng_sft_final_instruction_following_1749537926
DPO	gng_dpo_final_instruction_following_1749537965
Multi-Obj	gng_mutliobj_instruction_following_1749538006

Note: We also submitted .jsons via Gradescope as we were awaiting leaderboard scores

The SFT model won 194 out of 400 response comparisons, yielding a win rate of 48.50%. While close to the Qwen2.5's baseline performance, this highlights the need for more targeted optimization. We also observed signs of overfitting in "best" epochs, seen through qualitative analysis that shows the model's shift towards question and answer-style responses indicative of the Smoltalk dataset.

The DPO model won 242 out of 400 response comparisons, yielding an increased win rate of 60.50% over Qwen2.5. This gain over the baseline SFT model this optimization was trained on shows that the model was able to generally produce responses that were more preferred by Nemotron. Though this optimization was done with a single epoch and a small number of batches (3000, with a batch size of two), result shows that even a modest amount of optimization via DPO was able to significantly improve the percieved helpfulness of the model on the instruction finetuning task.

The multi-objective model shows a modest increase in wins over Qwen2.5, with 208 wins out of 400 comparisons for a win rate of 52.00%. While the multi-objective model slightly outperforms SFT, its gains are less pronounced than those achieved by DPO. This outcome suggests that, although optimizing for multiple distinct criteria like helpfulness and fluency can promote more diverse and engaging responses, our current approach may not have fully leveraged the potential of multi-objective reinforcement learning. Limitations such as a small number of reward dimensions and a simple scalarized loss function likely contributed to the model's relatively underwhelming performance. In addition, it is possible that the selected reward models do not align perfectly with Nemotron's scoring preferences, leading this method to incentivize outputs that are misaligned with desired target outputs.

5.2 Qualitative Analysis

5.2.1 Examples

One sample result with a prompt from the held out test set and the corresponding responses from each model:

- **Prompt:** "How is augmented reality being used to enhance museum experiences and engage visitors with interactive exhibits?"
- **SFT Model response:** "Augmented reality (AR) is being used to significantly enhance museum experiences by creating immersive and interactive exhibits that blur the lines between the physical and digital worlds. These AR experiences can be used to bring exhibits to life, allowing visitors to explore and interact with the information in a more engaging and meaningful way. For instance, virtual tours of historical artifacts can be augmented with 3D models and descriptions, providing visitors with a more personalized and interactive experience... This can be especially useful for museums that want to reach a wider audience, or for museums that want to create interactive exhibits that can be accessed from a variety of devices., AR is being used to create immersive and interactive exhibits that can enhance the museum experience and engage visitors in meaningful ways. By bringing exhibits to life in a more interactive and dynamic way, museums can create a more memorable and impactful experience for visitors.: Is there any current technology being developed that can handle large amounts of data at very high speeds that would be able to support immersive experiences?, there are several technologies currently being developed that can handle large amounts of data at very high speeds and support immersive experiences. Some of these technologies include high-definition cameras, 4K video recorders, and sensors that can collect and"
- **DPO Model response:** "Augmented reality (AR) is being integrated into museum experiences to create immersive, interactive, and engaging experiences for visitors. One way AR is being used is through augmented tours, where visitors can scan virtual exhibits to see 3D reconstructions, hear historical recordings, or see virtual artifacts in a more immersive environment. For example, a museum might use AR to bring ancient civilizations to life,

allowing visitors to explore ancient ruins and learn about historical events through interactive AR experiences... This personalized approach to museum visits helps visitors to engage more deeply with the content and culture of the institution they visit.: What are some of the benefits of incorporating augmented reality into museum experiences?augmented reality (AR) into museum experiences can bring numerous benefits to visitors and the museum itself. Some of the key benefits include:accessibility and"

- **Multi-objective Model response:** "Augmented reality (AR) technology is revolutionizing the way museums experience visitors. AR allows visitors to immerse themselves in interactive exhibits by overlaying digital content onto the physical space. This technique can be applied to various exhibits, from historical artifacts to contemporary art installations. For instance, museums can create immersive 3D models of ancient artifacts, providing visitors with a more authentic and engaging experience... Augmented reality (AR) brings a treasure trove of possibilities to the historical exhibit world, making it more accessible and enjoyable for younger audiences. One approach is to create immersive, interactive exhibits that transport visitors to specific eras and locations. For instance, museums can use AR to create an immersive environment where visitors can engage with historical artifacts, hear stories, and even taste the flavors of different periods. This can be achieved through AR glasses or mobile devices, allowing visitors to immerse themselves in the exhibit.strategy is to incorporate AR into virtual tours, allowing younger visitors to explore historical events and cultural heritage sites in entirely new ways. This can include AR-driven historical recreations, where visitors can step into historical settings, interact with artifacts, and even participate in hands-on activities. AR can also be used to create interactive storytelling, where visitors can explore historical events and cultural heritage sites through AR-generated narratives, complete with audio"

Sentences have been reduced in length. We were unable to set a lower token limit for responses when we ran the evaluating code.

5.2.2 Insight

The SFT model gives a broad but detailed answer. It touches on key uses of AR in museums like interactivity and personalization. However, repeats itself unnecessarily and is verbose, with phrases like “interactive and immersive” used multiple times. Additionally, its shift into a QA format mid-response reflects overfitting to instruction-following datasets like Smoltalk.

The DPO response is clear, focused, and well-structured. It stays tightly aligned with the prompt, offering concrete examples like AR tours and virtual reconstructions that enhance visitor engagement. The follow-up answer is concise and directly addresses the question about AR benefits.

The multi-objective model shines with creativity and vivid descriptions, including unique elements like “tasting the flavors of different periods” and immersive storytelling. It expands the educational and emotional impact of AR in museums, especially for younger visitors. The grammar occasionally falters (“museums experience visitors”), and the structure is a bit scattered, jumping between ideas without clear transitions.

6 Discussion

6.1 Implications

Our findings highlight key trade-offs between traditional and multi-objective fine-tuning methods for instruction-following models. DPO demonstrates that even a relatively small number of preference optimization steps can yield significant gains in reward model scores and preferred output quality. This suggests DPO remains a strong default method for improving response helpfulness and alignment under limited resource constraints.

However, the multi-objective approach shows promise for capturing richer response qualities. While our model only showed modest quantitative improvements, qualitative inspection revealed a more dynamic and engaging response style. That said, our performance may have been constrained by the limited number of reward objectives used—namely, helpfulness and fluency. Reducing the set of tracked qualities to just two dimensions may have introduced blind spots, where the model failed to capture other important attributes like factuality, coherence, or politeness. The lack of broader multi-faceted feedback could lead to instability or inconsistent alignment across varied prompts. Future work could address this by integrating additional reward heads and exploring how the dimensionality of supervision affects generalization, trade-offs, and controllability. We also highlight the potential implementation of other reinforcment learning methods with our reward heads.

6.2 Limitations

Although we worked extensively with the intent to implement our extension through DPO, we encountered substantial issues with our design process. We believe that these issues are derived from DPO's reliance on pairwise preference between responses, rather than supporting multiple independent reward signals directly.

Looking at DPO's loss function:

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

we can see that this method uses the log-likelihood difference between the preferred and less preferred responses under the current policy to directly optimize with human preferences. DPO cannot easily utilize multi-objective rewards (in our case, helpfulness and fluency) without collapsing them into a singular scalar beforehand. However, this defeats the purpose of multi-objective optimization in the first place, as this obscures how individual objectives contribute to the model's learning.

Thus, due to these limitations and the constraints of a ten-week class, we opted to implement a simpler scalarized reward aggregation method.

6.3 Future Work

In continuing our analysis of methods that may improve upon baseline Qwen2.5 performance on the instruction following task, we plan to further explore two main alternatives that could support multi-objective preference optimization: Reward-Weighted Likelihood Optimization (RLOO), and Generalized Preference Optimization (GRPO).

6.3.1 RLOO

Instead of using preferences between outputs, RLOO operates directly on scalar reward scores assigned to individual generated responses.

The RLOO objective updates the model by weighting the log-likelihood gradient with the scalar reward:

$$\nabla_\theta\mathcal{L}_{\text{RLOO}} = -\mathbb{E}_{x,y\sim\pi_\theta}\left[R(x,y)\nabla_\theta\log\pi_\theta(y \mid x)\right]$$

This is generalizable to any $R(x,y)$ which are scalarized functions of multiple reward components of any number. For example, a linear combination of helpfulness and fluency might look like so:

$$R(x,y) = \alpha\cdot r_{\text{helpfulness}}(x,y) + (1-\alpha)\cdot r_{\text{fluency}}(x,y)$$

Therefore, RLOO is better-suited for multi-objective reinforcement learning settings than DPO, as it supports continuous-values reward signals and does not require preference data.

6.3.2 GRPO

GRPO is a framework that extends the core ideas of DPO to multi-objective reward settings. While DPO assumes access to binary preferences or scalar values over outputs, GRPO is designed to handle multiple reward signals when modeling preference.

Here, each response is evaluated by a reward vector:

$$\vec{r}(x,y) = [r_1(x,y), r_2(x,y), \ldots, r_k(x,y)] \in \mathbb{R}^k,$$

where each $r_i(x,y)$ represents the reward for a specific objective—in our case, helpfulness or fluency. Those vectors are then transformed into scalar scores through a learned preference function, such as using a neural network:

$$f(\vec{r}) = \mathbf{w}^\top \cdot \phi(\vec{r}).$$

Next, GRPO compares two responses in a contrastive manner akin to DPO:

$$\mathcal{L}_{\text{GRPO}} = -\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}}\left[\log\sigma\left(f(\vec{r}(x,y^+)) - f(\vec{r}(x,y^-))\right)\right],$$

Therefore, GRPO is better able to utilize multi-objective rewards in its computation than DPO.

7 Conclusion

Our project explored improving instruction-following behavior in Qwen2.5-0.5B by utilizing a multi-objective reinforcement learning mechanism, optimizing for helpfulness and fluency. Starting from supervised fine-tuning and DPO, we introduced a scalarized reward aggregation method to support this task. While our approach provided modest improvements, it demonstrated that optimizing for multiple behavioral criteria can produce more nuanced and coherent responses. With future advancements in our work, in particular through RLOO or GRPO, we hope to see even greater improvements. In all, these results show the potential of multi-objective methods for advancing instruction-following capabilities in LLMs.

8 Team Contributions

Gabrielle Belanger: Implemented the DPO training pipeline, helped in writing the evaluation script used across all models, and contributed significantly to the writing and organization of the final report and poster.

George Dimopoulos: Contributed to the implementation of the multi-objective extension (w/ Nahome). Contributed to the final report's sections concerning the project extension, discussion, and conclusion. Contributed to filling out the project poster.

Nahome Hagos: Developed training scripts for the SFT and multi-objective (w/ George) models, helped in writing the evaluation script used across all models, ran most experiments on his local instance, and contributed to portions of the final report.

These breakdowns map on fairly well to initial projected work breakdown, with slight adjustments do to schedule alignment.

References

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, Colin Raffel, Leandro von Werra, and Thomas Wolf. 2025. SmolLM2: When Smol Goes Big – Data-Centric Training of a Small Language Model. arXiv:2502.02737 [cs.CL] `https://arxiv.org/abs/2502.02737`

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. 2023. UltraFeedback: Boosting Language Models with High-quality Feedback. arXiv:2310.01377 [cs.CL]

Nicole Meister, Carlos Guestrin, and Tatsunori Hashimoto. 2024. Benchmarking Distributional Alignment of Large Language Models. (2024). `https://arxiv.org/pdf/2411.05403`

Ryan Rafailov, Jason Li, Haotian Zhang, Pavel Izmailov, Polina Kirichenko, and Andrew Gordon Wilson. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. *arXiv preprint arXiv:2305.18290* (2023). https://arxiv.org/abs/2305.18290

Anikait Singh, Sheryl Hsu, Kyle Hsu, Eric Mitchell, Stefano Ermon, Tatsunori Hashimoto, Archit Sharma, and Chelsea Finn. 2025. FSPO: Few-Shot Preference Optimization of Synthetic Preference Data in LLMs Elicits Effective Personalization to Real Users. (2025). https://arxiv.org/abs/2502.19312

Taylor Sorensen, Jared Moore, Jillian Fisher Fisher, Mitchell Gordon, Niloofar Mireshghallah, Christopher Michael Rytting, Andre Ye, Liwei Jiang, Ximing Lu, Nouha Dziri, Tim Althodd, and Yejin Choi. 2024. A Roadmap to Pluralistic Alignment. (2024). https://arxiv.org/pdf/2402.05070