Extended Abstract

Motivation Large language models are used in contexts where user preferences vary widely across dimensions of interest. Traditional alignment techniques, including Few-Shot Preference Optimization (FSPO), typically reduce these nuanced preferences to a single scalar reward function,which oversimplifies user intent. My project addresses this limitation by exploring scalarization methods that preserve and balance multiple objectives in alignment training. This enables better personalization and more robust model behavior across diverse user values.

Method I extend the FSPO framework to handle multi-objective preference modeling by replacing its scalar reward signal with pareto. Instead of training on a single-axis reward, I construct multi-dimensional preference scores that represent distinct objectives such as helpfulness, correctness, harmlessness, and coherence. The design enables smoother access to interpreting the results and control over value alignment, directly addressing ethical concerns around bias and value collapse in existing alignment techniques.

Implementation I adapt FSPO's synthetic preference generation pipeline to produce labeled pairs across multiple axes and apply scalarization at the reward construction stage. The scalarized objective is then used to fine-tune a pre-trained LLM (e.g., Llama 3B). The training loop integrates custom scalarization functions and allows for dynamic weighting based on user input or pre-specified configurations. All components are implemented in PyTorch, and training is performed using distributed infrastructure on AWS GPUs.

Results Unfortunately my experments were not able to finish running on AWS for reasons that I had contacted the teaching via email.

Discussion Although I have not yet completed the experimental phase, prior research strongly suggests that integrating scalarization into preference optimization should lead to improved alignment across multiple user objectives. Scalarization methods are well-established in multi-objective optimization for their ability to encode trade-offs, and their application to language model alignment is supported by theoretical and empirical work. I expect that models trained using scalarized objectives will better capture nuanced user preferences. However, challenges remain in selecting appropriate scalarization strategies and balancing objectives effectively, particularly in the presence of noisy or conflicting preference data. These considerations will guide the design of future experiments and evaluation protocols.

Conclusion By integrating scalarization methods into FSPO, I enable large language models to align with multi-dimensional user preferences more effectively. This contributes both a novel technical enhancement and a step toward more pluralistic, ethically robust AI systems. Future work will focus on expanding preference axes, improving evaluation methodologies, and exploring applications in real-world alignment scenarios.

Beyond Single Rewards: Multi-Objective Scalarization in Few-Shot Preference Learning

Gabriel SantaCruz
Department of Computer Science
Stanford University
gsantac@stanford.edu

Abstract

As large language models (LLMs) are deployed in increasingly diverse settings, aligning them with nuanced user preferences across multiple dimensions—such as helpfulness, correctness, harmlessness, and coherence—has become essential. Existing techniques like Few-Shot Preference Optimization (FSPO) reduce these complex preferences to a single scalar reward, oversimplifying user intent. In this project, I extend FSPO by integrating scalarization methods, particularly Pareto-based approaches, to model and balance multi-dimensional objectives more effectively. I adapt FSPO's synthetic data pipeline to generate labeled comparisons across axes and implement a training loop using scalarized objectives to fine-tune a pre-trained model. While experiments could not be completed due to compute issues, prior research suggests that scalarization improves alignment fidelity, personalization, and interpretability. This work lays the foundation for future experiments and contributes a more pluralistic and transparent approach to preference-based LLM fine-tuning.

1 Introduction

Large language models (LLMs) are increasingly deployed in interactive settings where they must align with nuanced and diverse human preferences. To address this challenge, preference-based reinforcement learning approaches like Few-Shot Preference Optimization (FSPO) have been developed. These methods typically involve training LLMs using feedback derived from human or synthetic comparisons, with the goal of aligning model outputs to user preferences. However, a core limitation of current frameworks is their reliance on a scalar reward function that collapses multi-dimensional preferences—such as helpfulness, factuality, creativity, and ethical alignment—into a single numerical value. This reductionist approach can obscure important trade-offs, limit personalization, and introduce ethical concerns.

Reducing complex value structures to a single dimension introduces several problems. First, it limits the ability of alignment algorithms to reflect nuanced user trade-offs, leading to generic or misaligned behavior. Second, it hampers personalization by forcing all users into a one-size-fits-all optimization path, even when their priorities differ. Third, it risks over-optimizing toward dominant values while neglecting minority or underrepresented perspectives, raising fairness and inclusivity concerns. Finally, scalar reward formulations hinder transparency, making it unclear which specific values or behaviors are being optimized.

This project proposes an extension to the FSPO framework by introducing scalarization methods that preserve and balance multiple preference dimensions during training. Instead of using a fixed scalar reward, I represent each axis—helpfulness, creativity, factuality, and coherence—explicitly and apply scalarization techniques such as weighted sums, Pareto front approximations, and learned

combinations to generate a more informative training signal. This design enables interpretable, adaptable, and ethically robust alignment across diverse user preferences.

The central research question is: *Can scalarization improve the alignment of large language models with multi-dimensional user preferences by accurately modeling and balancing competing objectives within the FSPO framework?* Through this investigation, I aim to enhance alignment quality and personalization while contributing to a more transparent and pluralistic approach to LLM fine-tuning.

2 Related Work

Efforts to align large language models (LLMs) with user preferences have largely centered on scalar reward modeling. One notable approach is Few-Shot Preference Optimization (FSPO) (1), which uses synthetic comparisons to fine-tune models toward individual users. While effective in practice, FSPO reduces rich, multi-dimensional preferences to a single reward signal. This simplification can obscure important value trade-offs and hinder personalization, especially when users prioritize different objectives. Moreover, the reliance on synthetic data raises concerns about fairness and potential bias amplification, especially in the absence of robust human oversight.

Complementary work on pluralistic alignment (2) highlights the need to preserve diverse perspectives in aligned models. By introducing a taxonomy of alignment types and advocating for value pluralism, this line of research underscores the limitations of one-size-fits-all optimization. However, it stops short of offering concrete tools for managing competing objectives during training. This project addresses that gap by applying scalarization techniques—borrowed from multi-objective optimization—to represent and balance multiple preference dimensions, offering a more flexible and interpretable alternative to traditional scalar reward modeling.

3 Method

In this section, I present Few-Shot Preference Optimization (FSPO). I first introduce the background and motivation, then formalize the mathematical framework, and finally detail my implementation.

3.1 Problem Formulation

Let π_θ denote my policy model with parameters θ, and π_{ref} denote a reference model (typically a supervised fine-tuned model). For each training example, I have:

- A prompt x
- A chosen response y_c and a rejected response y_r
- A set of dimension-specific preference scores $\{s_d\}_{d \in D}$ where D represents my set of evaluation dimensions

Each score $s_d \in [0, 1]$ quantifies the strength of preference between y_c and y_r along dimension d. A score of 1 indicates strong preference for y_c over y_r, while a score of 0 indicates no preference.

3.2 FSPO Loss Function

The standard DPO objective for a single preference pair (x, y_c, y_r) is:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta, \pi_{\text{ref}}) = -\log \sigma \left(\beta \cdot \left(\log \frac{\pi_\theta(y_c|x)}{\pi_{\text{ref}}(y_c|x)} - \log \frac{\pi_\theta(y_r|x)}{\pi_{\text{ref}}(y_r|x)} \right) \right) \tag{1}$$

where β is a temperature parameter and σ is the sigmoid function.

I extend this to incorporate dimension-specific preference signals:

$$\mathcal{L}_d(\pi_\theta, \pi_{\text{ref}}) = -\log \sigma \left(\beta \cdot s_d \cdot \left(\log \frac{\pi_\theta(y_c|x)}{\pi_{\text{ref}}(y_c|x)} - \log \frac{\pi_\theta(y_r|x)}{\pi_{\text{ref}}(y_r|x)} \right) \right) \tag{2}$$

Here, s_d scales the log probability ratio based on the strength of preference in dimension d. This formulation has several advantages:

- It naturally handles varying preference strengths
- It reduces to standard DPO when $s_d = 1$ for all dimensions
- It allows for dimension-specific optimization

The overall FSPO loss is then a scalarization of the dimension-specific losses:

$$\mathcal{L}_{\text{FSPO}}(\pi_\theta, \pi_{\text{ref}}) = \mathcal{S}(\{\mathcal{L}_d\}_{d \in D}) \tag{3}$$

where $\mathcal{S}$ is a scalarization function that aggregates the dimension-specific losses.

3.3 Scalarization Methods

3.3.1 Weighted Sum

The simplest approach linearly combines the dimension-specific losses:

$$\mathcal{S}_{\text{weighted}}(\{\mathcal{L}_d\}_{d \in D}) = \sum_{d \in D} w_d \cdot \mathcal{L}_d \tag{4}$$

where $w_d \geq 0$ is the weight for dimension d and $\sum_{d \in D} w_d = 1$. While straightforward, this approach may not preserve Pareto optimality when the Pareto front is non-convex.

3.3.2 Chebyshev Scalarization

This approach focuses on minimizing the worst-case performance across dimensions:

$$\mathcal{S}_{\text{chebyshev}}(\{\mathcal{L}_d\}_{d \in D}) = \max_{d \in D} \{w_d \cdot (z_d^* - \mathcal{L}_d)\} \tag{5}$$

where z_d^* is a reference point (typically the ideal point) and w_d is the weight for dimension d. This method is particularly useful when I want to ensure that no single dimension is severely compromised.

3.3.3 Pareto-Aware Aggregation

My most sophisticated approach explicitly maintains Pareto optimality:

Algorithm 1 Pareto-Aware Aggregation

ParetoAwareScalarization$\{\mathcal{L}_d\}_{d \in D}$, $\{w_d\}_{d \in D}$ Apply weights: $\mathcal{L}_d^w \leftarrow w_d \cdot \mathcal{L}_d$ for all $d \in D$ Identify Pareto-dominant solutions in the batch Compute reference point: $r_d \leftarrow \min_{\text{batch}} \mathcal{L}_d^w - 1$ for all $d \in D$ Compute hypervolume contributions for Pareto-optimal solutions Compute weighted sum: $\mathcal{W} \leftarrow \sum_{d \in D} \mathcal{L}_d^w$ For Pareto-optimal solutions, add hypervolume contribution scaled by temperature Scalarized values

This method ensures that:

- Solutions on the Pareto front receive higher scores
- The hypervolume contribution rewards solutions that dominate larger portions of the objective space
- Trade-offs between dimensions respect Pareto optimality

3.4 Training Procedure

Algorithm 2 outlines my training procedure:

Algorithm 2 FSPO Training Procedure

Policy model π_θ, reference model π_{ref}, dataset $\mathcal{D}$, dimensions D, scalarization method $\mathcal{S}$ Initialize optimizer with learning rate α each epoch each batch $(x, y_c, y_r, \{s_d\}_{d \in D})$ in $\mathcal{D}$ Compute policy log probabilities: $\log \pi_\theta(y_c|x)$, $\log \pi_\theta(y_r|x)$ Compute reference log probabilities: $\log \pi_{\text{ref}}(y_c|x)$, $\log \pi_{\text{ref}}(y_r|x)$ Compute dimension-specific losses $\mathcal{L}_d$ for all $d \in D$ Apply scalarization: $\mathcal{L}_{\text{FSPO}} = \mathcal{S}(\{\mathcal{L}_d\}_{d \in D})$ Update model: $\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}_{\text{FSPO}}$ Evaluate model on validation set Save checkpoint if performance improves Optimized policy model π_θ

3.5 Implementation Details

I implement FSPO using PyTorch and the Hugging Face Transformers library. My implementation includes several key components:

3.5.1 Parameter-Efficient Fine-Tuning

To efficiently adapt large language models, I employ Low-Rank Adaptation (LoRA) (**?**):

$$W = W_0 + \Delta W = W_0 + BA \tag{6}$$

where W_0 is the pre-trained weight matrix, $B \in \mathbb{R}^{d \times r}$, $A \in \mathbb{R}^{r \times k}$, and $r \ll \min(d, k)$. This reduces the number of trainable parameters while maintaining model quality.

3.5.2 Log Probability Computation

For efficient computation of log probabilities, I use the following approach:

$$\log p(y|x) = \frac{1}{|y|} \sum_{t=1}^{|y|} \log p(y_t|x, y_{<t}) \tag{7}$$

where $|y|$ is the length of the response and y_t is the token at position t.

3.5.3 Dimension Weighting

Users can specify custom weights for each dimension, allowing for application-specific optimization:

$$w_d = \frac{w'_d}{\sum_{d' \in D} w'_{d'}} \quad \forall d \in D \tag{8}$$

where w'_d is the user-specified weight for dimension d.

3.6 Rationale and Theoretical Justification

My FSPO approach is theoretically grounded in several key principles:

- **KL-Constrained Optimization**: Like DPO, FSPO can be derived as a solution to a constrained optimization problem that maximizes expected reward while limiting divergence from the reference model.
- **Multi-Objective Optimization**: By treating each dimension as a separate objective, I draw on established techniques from the multi-objective optimization literature.
- **Pareto Efficiency**: My Pareto-aware scalarization explicitly maintains the principle that an improvement in one dimension should not come at the expense of severe degradation in others.

The dimension-specific scaling factor s_d serves a crucial role in modulating the strength of the preference signal. When s_d is close to 1, the model receives a strong signal to prefer y_c over y_r along dimension d. Conversely, when s_d is close to 0, the model receives little guidance along that dimension. This allows for fine-grained control over the learning process and better alignment with nuanced human preferences.

4 Results

Unable to run the experiments so there are no results.

4.1 Quantitative Evaluation

No results.

4.2 Qualitative Analysis

No results.

5 Discussion

No results. The one barrier to finishing this project was my access to compute. AWS Spot instances made me restart my training many times and lost much of my progess. I used all of the credits we were given and could no longer run my experiments due to this.

6 Conclusion

Scalarization methods theoretically provide many benefits for alignment and model transparency. Given the proper resources and ability to train the models, this approach can be beneficial in improving FSPO.

7 Team Contributions

- I did this project individually

References

[1] Singh, A., Hsu, S., Hsu, K., Mitchell, E., Ermon, S., Hashimoto, T., Sharma, A., & Finn, C. (2025). FSPO: Few-shot preference optimization of synthetic preference data in LLMs elicits effective personalization to real users. *arXiv*.

[2] Sorensen, T., Moore, J., Fisher, J., Gordon, M., Mireshghallah, N., Rytting, C. M., Ye, A., Jiang, L., Lu, X., Dziri, N., Althoff, T., & Choi, Y. (2024). A roadmap to pluralistic alignment. *arXiv*, 1.