Week 4

Advisor: Dr. Yue Wang

Student: David Tang

Project Overview

Motivation

- RLHF, DPO: Typically learn a scalar preference signal, compress tradeoffs
- NLHF: Models nontransitive preferences via a game formulation



Shared Limitation

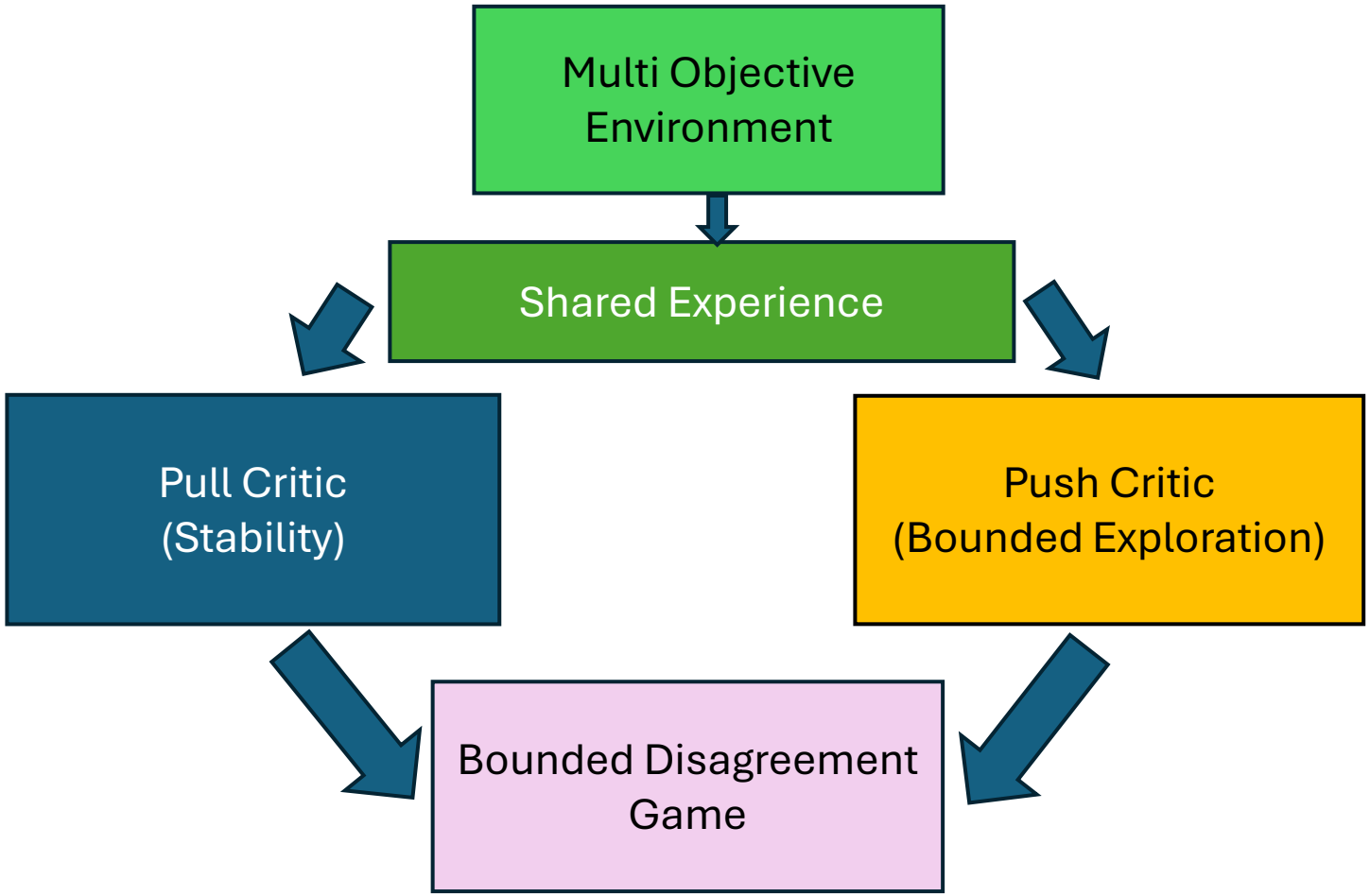
- Contextual Bandit Assumption (no long-horizon structure)
- Feedback on isolated samples rather than sequential trajectories
- Limited modeling of state-dependent change sover future returns



This motivates sequential, state-dependent vector-valued preference modeling in multi objective environments.

Reinforcement Learning from Human Feedback (RLHF); Direct Preference Optimization (DPO), Nash Learning from Human Feedback (NLHF)

Our Method: Nash Regularized Vector Critics



Multi-Objective Sequential Decisions

State: s_t

Action: a_t

Vector Reward: $\mathbf{r_t} = [r_1, r_2, ..., r_K]^T$ represents multiple objectives

Vector Value Function: $\mathbf{V(s)} = [V_1(s), V_2(s), ..., V_K(s)]^T$ estimates future return for each objective

Unlike Scalar RL:

- Objective dimensions are preserved throughout critic learning
- Scalarized only during policy optimization

Dual Vector Critics

- Both estimate multi-objective value functions under shared environment dynamics but optimize different objectives.

$$V_{\phi_{\text{pull}}}(s), \quad V_{\phi_{\text{push}}}(s) \in \mathbb{R}^K$$

Pull Critic	Push Critic
TD Learning	TD Learning
Consensus Pressure	Bounded Divergence Pressure
Value Stabilization	Preference Tradeoff Exploration

Regularized Two-Critic Minimax Game

- Our disagreement is defined as:

$$D(\phi_{\text{pull}}, \phi_{\text{push}} \mid s) = \|\mathbf{V}_{\phi_{\text{pull}}}(s) - \mathbf{V}_{\phi_{\text{push}}}(s)\|_2$$

- Our weighting is defined as:

$$w(D) = \text{clip}\left(\frac{D(\phi_{\text{pull}}, \phi_{\text{push}} \mid s)}{c}, 0, 1\right)$$

- Our minimax objective defined as:

$$\min_{\phi_{\text{pull}}} \max_{\phi_{\text{push}}} \mathbb{E}_{s\sim\mathcal{D}}\left[w(D) \cdot D(\phi_{\text{pull}}, \phi_{\text{push}} \mid s)^2\right]$$

$\mathcal{D}$ is the policy-induced state distribution.

Critic Losses

- The pull critic seeks a consensus (stabilization)

$$L(\phi_{\text{pull}}) = \mathbb{E}_{s\sim\mathcal{D}}\left[\|V_{\phi_{\text{pull}}}(s) - \mathcal{R}_{\text{pull}}\|_2^2 + \beta\, w(D)\, \|V_{\phi_{\text{pull}}}(s) - \text{sg}(V_{\phi_{\text{push}}}(s))\|_2^2\right]$$

- The push critic induces bounded divergence (structured exploration)

$$L(\phi_{\text{push}}) = \mathbb{E}_{s\sim\mathcal{D}}\left[\|V_{\phi_{\text{push}}}(s) - \mathcal{R}_{\text{push}}\|_2^2 - \beta\, w(D)\, \|V_{\phi_{\text{push}}}(s) - \text{sg}(V_{\phi_{\text{pull}}}(s))\|_2^2 + \zeta \max\left(0, \|V_{\phi_{\text{push}}}(s) - \text{sg}(V_{\phi_{\text{pull}}}(s))\|_2^2 - \delta_{\max}^2\right)\right]$$

$\mathcal{R}$: vector return targets

Scalarization Quick Recap

Let $\mathbf{x} = [x_1, x_2, \ldots, x_K]^T$ be the multi-dimensional metric vector

$$S_{\text{linear}}(\mathbf{x}) = \sum_{k=1}^{K} w_k x_k \qquad \sum_{k=1}^{K} w_k = 1$$

$$S_{\text{product}}(\mathbf{x}) = \prod_{k=1}^{K} x_k$$

Linear Scalarization	Product (Nash Social Welfare)
"OR-Like" tradeoff behavior	"Soft AND-Like" behavior
High performance in one objective can offset another	Objectives remain coupled; weakness in one reduces overall utility
Requires choosing objective weights	Encourages balanced solutions without explicit weight tuning

Actor Optimization

- Actor is optimized using stabilized consensus of the coupled critics' value estimates
- While PPO is used in this example, it is not limited to only PPO.

$$\mathbf{A}_{\text{pull}},\ \mathbf{A}_{\text{push}} \in \mathbb{R}^K$$

$$\mathbf{A}_{\text{fused}} = \frac{\mathbf{A}_{\text{pull}} + \mathbf{A}_{\text{push}}}{2}$$

$$\hat{A} = \text{Normalize}\left(S(\mathbf{A}_{\text{fused}}) \right)$$

$$L_{\text{actor}}(\theta) = L_{\text{PPO}}(\theta; \hat{A})$$

$S(\cdot) : \mathbb{R}^K \to \mathbb{R}$ is a scalarization operator.

Timeline

- Week 1: Linear vs Nash Social Welfare + A2C baseline
- Week 2: PPO baseline
- Week 3: Nash Dual Critic Formulation/Initial Implementation
- Week 4: Nash Dual Critic Integration, Debugging, Stabilizing
- Week 5: Evaluation across additional environments
- Week 6: Hyperparameter sweeps and ablation experiments
- Week 7: Statistical Analysis / Report
- Week 8: Statistical Analysis / Report

Extras: Scalarization (Speed vs Safety Example)

- In Reinforcement Learning, we want to maximize our rewards

- Simple Average

$$R_{\text{avg}} = \frac{r_{\text{speed}} + r_{\text{safety}}}{2}$$

Action	Speed	Safety
Reckless	100	1
Cautious	40	20

R(Reckless)=0.5(100+1)=50.5

R(Cautious)=0.5(40+20)=30

Extras: What if we tried learning/tuning the weights

- Learned/Tuned weights

$$R = w_{\text{speed}} r_{\text{speed}} + w_{\text{safety}} r_{\text{safety}}$$

$$w_{\text{speed}} = 0.3$$

$$w_{\text{safety}} = 0.7$$

Action	Speed	Safety
Reckless	100	20
Cautious	60	80
Dangerous	300	20

R(Reckless)=0.3(100)+0.7(20)=44

R(Cautious)=0.3(60)+0.7(80)=74

R(Dangerous)=0.3(300)+0.7(20)=104

Linear acts as compensatory OR gate
Speed can compensate for Safety

Extras: Product Aggregation

Action	Speed	Safety
Reckless	100	1
Cautious	40	20

$$R_{\text{product}} = r_{\text{speed}} \cdot r_{\text{safety}}$$

R(Reckless)=100*1=100

R(Cautious)=40*20=800

The product acts as a Soft AND Gate
Objectives are coupled; neither objective can be ignored

Thank You

- Questions?