DPO, ORPO, GRPO, and PPO - Theory (Part-1)

Part-1 covers theoretical background. Refer part-2 for comparison of techniques and part-3 for the real world scenario.

1 Theoretical Background

Core Concept

Preference optimization methods align language models with human preferences by training models to prefer better responses over worse ones. These techniques emerged to simplify and improve upon traditional Reinforcement Learning from Human Feedback (RLHF).

1.1 PPO (Proximal Policy Optimization)

PPO is the **original RL algorithm used for RLHF** and the foundation of LLM alignment. It works by:

- Using a **reward model** trained on human preference data to score responses
- Optimizing the policy through **constrained policy updates** using a clipped surrogate objective
- Employing a **value function (critic network)** to estimate advantages and reduce variance
- Using **KL-divergence penalties** relative to the reference policy to prevent drastic changes

The key innovation: PPO constrains policy updates to a "trust region" through clipping, ensuring training stability while maintaining sample efficiency.

1.2 DPO (Direct Preference Optimization)

DPO reformulates the RLHF objective to eliminate the need for explicit reward models and complex RL training. It works by:

- Reparameterizing the RLHF objective to create an **implicit reward function** derived from the policy itself and a reference model
- Framing preference learning as a **classification problem** between chosen and rejected responses
- Using a static preference dataset with paired responses (chosen vs rejected)

The key insight: DPO indirectly solves the RLHF objective by optimizing this implicit reward, avoiding the instability of RL training.

1.3 ORPO (Odds Ratio Preference Optimization)

ORPO combines SFT and preference optimization into a **single monolithic training stage**. Its innovations include:

- Eliminating the reference model entirely, reducing memory by 20-30%
- Using the **odds ratio** to contrast preferred vs dispreferred responses during training
- Penalizing disfavored responses during SFT rather than treating all examples equally

ORPO measures the **absolute odds ratio within the current policy**, whereas DPO measures preference change relative to a frozen reference model.

1.4 GRPO (Group Relative Policy Optimization)

GRPO represents a paradigm shift introduced by DeepSeek in January 2025. It eliminates both reward models and critic networks by using:

- **Group-relative scoring**: For each prompt, generate 8-16 responses and score them against verifiable rewards (e.g., code correctness, math accuracy)
- **Advantage estimation**: Each response's advantage is computed relative to the group mean
- **Policy updates**: Increase probability of above-average responses, decrease probability of below-average ones

Recent research shows GRPO and DPO optimize fundamentally similar contrastive objectives—the difference lies in online vs offline training regimes.

---

2 Comparison Tables

2.1 Comprehensive Method Comparison

Criterion	PPO	DPO	ORPO	GRPO
Reference Model	Required	Required	Not required	Not required
Reward Model	Required (separate training)	Not required	Not required	Optional
Critic Network	Required (value function)	Not required	Not required	Not required
Training Stages	3 stages (SFT → RM → PPO)	2 stages (SFT → DPO)	1 stage	2 stages (SFT → GRPO)
Data Requirements	Preference pairs for RM, then online generation	Paired preferences	Paired preferences	Prompts + verifiable rewards
Compute Cost	Highest (4 models in memory)	Medium	Low	High
Memory Requirements	3-4× base model	2× base model	1× base model	1-2× base model

Learning Type	Online RL	Offline contrastive	Offline hybrid	Online RL
Implementation Complexity	Very high (37+ details)	Low	Low	Medium
Training Stability	Moderate (prone to instability)	Good	Very good	Excellent
Hyperparameter Sensitivity	Very high	Low	Low	Medium
Sample Efficiency	Low (on-policy)	High (offline)	High	Moderate
Best Performance	Complex reasoning, code generation	Safety, style, dialogue	Resource-c onstrained	Math, code with verifiable outputs
Catastrophic Forgetting Risk	High	Low	Low	Low
Fine-grained Control	Excellent	Limited	Limited	Good
Theoretical Foundation	Trust region optimization	Implicit reward maximization	Odds ratio during SFT	Group-relative advantage

2.2 Performance Comparison by Task

Task Type	PPO	DPO	ORPO	GRPO
Code Generation	⭐⭐⭐⭐⭐ Excellent	⭐⭐⭐ Good	⭐⭐ Fair	⭐⭐⭐⭐ Very Good
Mathematical Reasoning	⭐⭐⭐⭐⭐ Excellent	⭐⭐⭐ Good	⭐⭐ Fair	⭐⭐⭐⭐⭐ Excellent
Dialogue/Chat	⭐⭐⭐⭐ Very Good	⭐⭐⭐⭐⭐ Excellent	⭐⭐⭐ Good	⭐⭐⭐ Good
Safety Alignment	⭐⭐⭐⭐ Very Good	⭐⭐⭐⭐⭐ Excellent	⭐⭐⭐ Good	⭐⭐⭐ Good

Summarization	⭐⭐⭐⭐ Very Good	⭐⭐⭐⭐⭐ Excellent	⭐⭐⭐⭐ Very Good	⭐⭐⭐ Good
Small Models (<7B)	⭐⭐⭐ Good	⭐⭐⭐⭐ Very Good	⭐⭐⭐⭐⭐ Excellent	⭐⭐ Fair
Large Models (7B+)	⭐⭐⭐⭐⭐ Excellent	⭐⭐⭐⭐ Very Good	⭐⭐⭐ Good	⭐⭐⭐⭐⭐ Excellent

2.3 Resource Requirements Comparison

Resource	PPO	DPO	ORPO	GRPO
GPU Memory	3-4× base model	2× base model	1× base model	1-2× base model
Training Time	Longest	Medium	Shortest	Long
Implementation Effort	Very High (37+ details)	Low	Very Low	Medium
Infrastructure Needs	Complex (4 models)	Moderate (2 models)	Simple (1 model)	Moderate (vLLM server)
Hyperparameter Tuning	Extensive	Minimal	Minimal	Moderate
Data Requirements	Preference pairs + RM training	Preference pairs	Preference pairs	Prompts + verifiable rewards
Sample Efficiency	Low (on-policy)	High (offline)	High (offline)	Moderate (online)

2.4 Training Characteristics Comparison

Characteristic	PPO	DPO	ORPO	GRPO
Training Stability	⚠️ Moderate (unstable)	✅ Good	✅✅ Very Good	✅✅ Excellent

Convergence Speed	Slow	Fast	Fastest	Moderate
Catastrophic Forgetting	⚠️ High Risk	✅ Low Risk	✅ Low Risk	✅ Low Risk
Reward Signal Quality	Flexible (learned)	Fixed (implicit)	Fixed (odds ratio)	Verifiable (ground truth)
Online Learning	✅ Yes	❌ No	❌ No	✅ Yes
Exploration Capability	✅ High	❌ None	❌ None	✅ Moderate
KL Divergence Control	Explicit penalty	Implicit via reference	None needed	Implicit

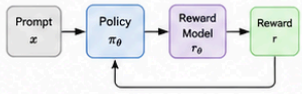
2.5 Pipeline Components by Method

Component	PPO	DPO	ORPO	GRPO
Policy Model	✓	✓	✓	✓
Reference Model	✓	✓	✗	✗
Reward Model	✓ (trained)	✗	✗	✓ (optional)
Value/Critic Model	✓	✗	✗	✗

3 At a Glance

PPO (Proximal Policy Optimization)

SETUP



OBJECTIVE

Maximize reward while keeping policy close to reference policy π_{ref}

$$\max_\theta \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\theta}\left[r_\theta(x,y)\right]$$

$$s.t. \quad \mathrm{KL}(\pi_\theta \,\|\, \pi_{ref}) \le \epsilon$$

PPO CLIPPED OBJECTIVE

$$L_{PPO}(\theta) = \mathbb{E}\left[\min\left(r_t(\theta)\hat{A}_t,\ \mathrm{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right] - \beta\ \mathrm{KL}(\pi_\theta \,\|\, \pi_{ref})$$

where $r_t(\theta) = \dfrac{\pi_\theta(y_t|x_t)}{\pi_{\theta_{old}}(y_t|x_t)}$

ADVANTAGE (GAE)

$$\hat{A}_t = \sum_{l=0}^{\infty} (\gamma\lambda)^l \quad \delta_{t+l}$$

e.g., $\delta_t = r_t + \gamma V(x_{t+1}) - V(x_t)$

VALUE LOSS

$$L_V(\theta) = \mathbb{E}\left[(V_\theta(x_t) - R_t)^2\right]$$

TOTAL LOSS

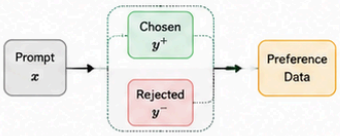
$$L_{PPO\ total} = -L_{PPO} + c_1 L_V - c_2 \mathcal{H}(\pi_\theta)$$

NOTES / KEY POINTS

- On-policy
- Uses reward model for scalar reward
- Stable with clipping + KL penalty
- Requires value function (critic)

DPO (Direct Preference Optimization)

SETUP



OBJECTIVE

Maximize log-prob of chosen over rejected relative to reference policy π_{ref}

$$\max_\theta \mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}}\left[\log\sigma\left(\beta\,\Delta_\theta(x,y^+,y^-)\right)\right]$$

DPO LOSS

$$L_{DPO}(\theta) = -\mathbb{E}_{(x,y^+,y^-)}\left[\log\sigma\left(\beta\,\Delta_\theta(x,y^+,y^-)\right)\right]$$

where

$$\Delta_\theta(x,y^+,y^-) = \log\frac{\pi_\theta(y^+|x)}{\pi_{ref}(y^+|x)} - \log\frac{\pi_\theta(y^-|x)}{\pi_{ref}(y^-|x)}$$

$$\sigma(z) = \frac{1}{1+e^{-z}}$$

OPTIONAL REGULARIZATION

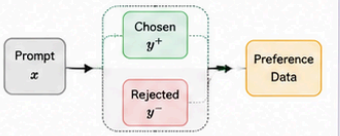
$$L = L_{DPO} + \lambda\ \mathrm{KL}(\pi_\theta \,\|\, \pi_{ref})$$

NOTES / KEY POINTS

- Supervised on preference pairs
- No reward model or critic needed
- Equivalent to Bradley–Terry model
- Simple, efficient, stable

ORPO (Odds Ratio Preference Optimization)

SETUP



OBJECTIVE

Optimize odds ratio with explicit normalization (improves stability)

$$\max_\theta \mathbb{E}_{(x,y^+,y^-)}\left[\log\frac{\pi_\theta(y^+|x)}{\pi_\theta(y^-|x)} - \log\frac{\pi_{ref}(y^+|x)}{\pi_{ref}(y^-|x)}\right]$$

ORPO LOSS

$$L_{ORPO}(\theta) = -\mathbb{E}_{(x,y^+,y^-)}\left[\log\sigma\left(\beta\,\Gamma_\theta(x,y^+,y^-)\right)\right]$$

where

$$\Gamma_\theta(x,y^+,y^-) = \log\frac{\pi_\theta(y^+|x)}{\pi_\theta(y^-|x)}\,\frac{\pi_{ref}(y^-|x)}{\pi_{ref}(y^+|x)}$$

OPTIONAL REGULARIZATION

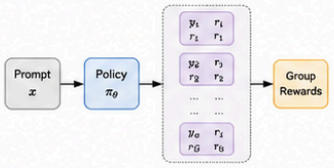
$$L = L_{ORPO} + \lambda\ \mathrm{KL}(\pi_\theta \,\|\, \pi_{ref})$$

NOTES / KEY POINTS

- Odds ratio formulation (symmetric)
- Better calibrated gradients than DPO
- No reward model or critic
- Simple and efficient

GRPO (Group Relative Policy Optimization)

SETUP



OBJECTIVE

Maximize group-relative advantage (no value function required)

$$\max_\theta \mathbb{E}_{x\sim\mathcal{D}}\ \mathbb{E}_{y_{1:G}\sim\pi_\theta}\left[\frac{1}{G}\sum_{i=1}^{G}\hat{A}_i\log\pi_\theta(y_i|x)\right]$$

GROUP ADVANTAGE

$$A_i = r_i - \frac{1}{G}\sum_{j=1}^{G} r_j$$

$$\hat{A}_i = \frac{A_i - \mu_A}{\sigma_A + e}$$

$$\mu_A = \frac{1}{G}\sum_{j=1}^{G} A_j, \quad \sigma_A^2 = \frac{1}{G}\sum_{j=1}^{G}(A_j - \mu_A)^2$$

GRPO LOSS

$$L_{GRPO}(\theta) = -\mathbb{E}_x\ \mathbb{E}_{y_{1:G}}\left[\frac{1}{G}\sum_{i=1}^{G}\hat{A}_i\log\pi_\theta(y_i|x)\right] - \beta\ \mathrm{KL}(\pi_\theta \,\|\, \pi_{ref})$$

NOTES / KEY POINTS

- Sample G responses per prompt
- Uses group-relative advantage
- No critic / value model
- Efficient and scalable

AT A GLANCE	On-policy RL Reward model + Critic Clip + KL for stability	Offline preference learning Pairwise (chosen vs rejected) Simple logistic objective	Odds ratio between policies Improved gradient calibration No critic, no reward model	Group sampling + relative advantage No critic, no reward model Scalable and efficient

Symbols: π_θ : current policy π_{ref} : reference policy r_θ : reward model x : prompt y : response y^+, y^- : chosen / rejected r : reward KL : Kullback–Leibler divergence $\sigma(z) = \frac{1}{1+e^{-z}}$: sigmoid β : inverse temperature / regularization strength ϵ : clip range G : group size $\mathbb{E}$: expectation

3.1 PPO (Proximal Policy Optimization)

Pipeline: Base Model → SFT → Reward Model Training → PPO Optimization

Key Equation:

$$L^{PPO}(\theta) = \mathbb{E}_t \left[\min \left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t \right) \right] - \beta D_{KL}(\pi_\theta || \pi_{ref})$$

Where: $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{old}(a_t|s_t)}$

Components:

- Policy: π_θ
- Reference Policy: π_{ref}
- Reward Model: $r_\phi(x, y)$
- Value Function: $V_\psi(x)$
- Advantage: $\hat{A}_t = r(x, y) + V(x') - V(x)$

Objective:

$$\max_\theta \mathbb{E}_{x \sim \mathcal{D}, y \sim \pi_\theta(y|x)}[r_\phi(x, y)] - \beta D_{KL}(\pi_\theta || \pi_{ref})$$

3.2 DPO (Direct Preference Optimization)

Pipeline: Base Model → SFT → DPO

Key Equation:

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

Implicit Reward:

$$r(x, y) = \beta \log \frac{\pi_\theta(y|x)}{\pi_{ref}(y|x)}$$

Simplified Form:

$$L_{DPO} = -\log \sigma \left(\beta \left[\log \frac{\pi_\theta(y_w|x)}{\pi_{ref}(y_w|x)} - \log \frac{\pi_\theta(y_l|x)}{\pi_{ref}(y_l|x)} \right] \right)$$

Components:

- Policy: π_θ
- Reference Policy: π_{ref} (frozen)
- Preference pairs: (x, y_w, y_l) where $y_w \succ y_l$

Probability Interpretation:

$$P(y_w \succ y_l) = \sigma(r(x, y_w) - r(x, y_l))$$

3.3 ORPO (Odds Ratio Preference Optimization)

Pipeline: Base Model → ORPO (single stage)

Key Equation:

$$L_{ORPO} = L_{SFT} + \lambda \cdot L_{OR}$$

SFT Loss:

$$L_{SFT} = -\mathbb{E}_{(x,y)}[\log \pi_\theta(y|x)]$$

Odds Ratio Loss:

$$L_{OR} = -\mathbb{E}_{(x,y_w,y_l)} \left[\log \sigma \left(\log \frac{\text{odds}_\theta(y_w|x)}{\text{odds}_\theta(y_l|x)} \right) \right]$$

Odds Definition:

$$\text{odds}_\theta(y|x) = \frac{\pi_\theta(y|x)}{1 - \pi_\theta(y|x)}$$

Combined Objective:

$$L_{ORPO} = -\log \pi_\theta(y_w|x) - \lambda \log \sigma \left(\log \frac{\text{odds}_\theta(y_w|x)}{\text{odds}_\theta(y_l|x)} \right)$$

Components:

- Policy: π_θ (no reference model needed)
- Preference pairs: (x, y_w, y_l)

3.4 GRPO (Group Relative Policy Optimization)

Pipeline: Base Model → SFT → GRPO

Key Equation:

$$L_{GRPO}(\theta) = \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y_i\sim\pi_\theta(y|x)}\left[A_i \log \pi_\theta(y_i|x)\right] - \beta D_{KL}(\pi_\theta||\pi_{ref})$$

Group Advantage:

$$A_i = r(x, y_i) - \frac{1}{G}\sum_{j=1}^{G} r(x, y_j)$$

Per-Sample Update:

$$\nabla_\theta L = \sum_{i=1}^{G}(r_i - \bar{r})\nabla_\theta \log \pi_\theta(y_i|x)$$

Components:

- Policy: π_θ
- Group size: G (typically 8-16 samples per prompt)
- Reward function: $r(x, y)$ (verifiable, e.g., code correctness)
- Group mean: $\bar{r} = \frac{1}{G}\sum_{i=1}^{G} r_i$

Contrastive Form:

$$L_{GRPO} \approx -\mathbb{E}\left[\log \sigma\left(r(x, y_+) - r(x, y_-)\right)\right]$$

Where y_+ has above-average reward, y_- has below-average reward

3.5 Method Comparison Matrix

Method	Models Required	Loss Type	Data Format	KL Constraint
PPO	4 (π, πref, r, V)	Clipped surrogate	Online generation	Explicit penalty
DPO	2 (π, πref)	Binary cross-entropy	Preference pairs	Implicit via reference
ORPO	1 (π)	SFT + Odds ratio	Preference pairs	None
GRPO	1-2 (π, optional πref)	Policy gradient	Prompts + rewards	Optional penalty

3.6 Mathematical Relationships

DPO ≈ GRPO (offline case):

$$\lim_{G \to \infty} L_{GRPO} \approx L_{DPO} \text{ when rewards are preference-based}$$

ORPO vs DPO:

$$L_{DPO} \text{ uses } \log \frac{\pi_\theta}{\pi_{ref}}, \quad L_{ORPO} \text{ uses } \log \frac{\text{odds}_\theta(y_w)}{\text{odds}_\theta(y_l)}$$

PPO → DPO simplification:

$$\text{DPO eliminates: } r_\phi, V_\psi \text{ by deriving closed-form optimal policy}$$

3.7 Optimization Targets

Method	What It Maximizes
PPO	$\mathbb{E}[r_\phi(x, y)] - \beta D_{KL}(\pi_\theta \Vert \pi_{ref})$
DPO	$\mathbb{E}[\log P(y_w \succ y_l)]$ via implicit reward
ORPO	$\text{Likelihood}(y_w) + \lambda \cdot \text{OddsRatio}(y_w, y_l)$
GRPO	$\mathbb{E}_{group}[(r_i - \bar{r}) \log \pi(y_i)]$

4 Emerging Trends (2025-2026)

Trend	Method	Application	Impact
Reasoning models boom	GRPO, RLVR	Math, coding, formal logic	GRPO democratizes reasoning model training
Process reward models	PPO variants	Step-by-step feedback	OpenAI o1/o3 style training
Hybrid pipelines	DPO → PPO	Initial safety → performance tuning	Best of both worlds
RLAIF (RL from AI Feedback)	DPO + synthetic data	Scalable preference generation	Reduces human labeling cost
Multi-objective optimization	PPO with multiple RMs	Safety + capability + style	Gemini, Claude approach