

# RLHF: A comprehensive Survey for Cultural, Multimodal and Low Latency Alignment Methods

Raghav Sharma<sup>1\*</sup>

[sharma.raghav103@gmail.com](mailto:sharma.raghav103@gmail.com)

Manan Mehta<sup>2</sup>

[manan.mehta2@gmail.com](mailto:manan.mehta2@gmail.com)

Sai Tiger Raina<sup>2</sup>

[sai.raina@gmail.com](mailto:sai.raina@gmail.com)

<sup>1</sup> Northeastern University, Boston, MA 02115, USA

<sup>2</sup> University of Southern California, Los Angeles, CA 90089, USA

## 1 Abstract

Reinforcement Learning from Human Feedback (RLHF) is the standard for aligning Large Language Models (LLMs), yet recent progress has moved beyond canonical text-based methods. This survey synthesizes the new frontier of alignment research by addressing critical gaps in multi-modal alignment, cultural fairness, and low-latency optimization. To systematically explore these domains, we first review foundational algorithms, including PPO, DPO, and GRPO, before presenting a detailed analysis of the latest innovations. By providing a comparative synthesis of these techniques and outlining open challenges, this work serves as an essential roadmap for researchers building more robust, efficient, and equitable AI systems.

## 2 Introduction

The advent of models powered by Reinforcement Learning from Human Feedback (RLHF) marked a pivotal moment in artificial intelligence. This paradigm shift transformed Large Language Models from mere text generators into interactive, seemingly helpful assistants, making advanced AI accessible on a global scale. However, this initial success has also illuminated the limitations of a one-size-fits-all alignment strategy, revealing critical gaps in areas that previous surveys have not adequately addressed: **multi-modal alignment, cultural and demographic fairness, and methods for optimizing latency and cost**.

To provide a clear guide to this new frontier, this survey offers a comprehensive and structured overview of these recent advances. We first establish the foundational alignment toolkit, covering the principles of RLHF and the three primary policy optimization techniques that form its backbone: Proximal Policy Optimization (PPO), Direct Preference Optimization (DPO), and Group Relative Policy Optimization (GRPO). We then present a systematic review of the latest methods designed to tackle the identified gaps, culminating in a comparative synthesis and an examination of open challenges. This work provides researchers with a complete roadmap to both the current state of the field and its most promising future directions.

### 2.1 Reinforcement Learning in a Nutshell

The goal of Reinforcement Learning (RL) is to teach an agent to make good decisions by interacting with an environment and receiving feedback in the form of rewards. Common definitions used in RL:

- **Agent**: the decision-making entity.
- **Environment**: everything the agent interacts with.

- **State**  $s_t$ : what the agent observes at time  $t$ .
- **Action**  $a_t$ : the choice the agent makes at  $t$ .
- **Reward**  $r_t$ : a scalar signal indicating the quality of  $a_t$ .
- **Policy**  $\pi_\theta$ : a parameterised distribution  $\pi_\theta(a | s)$  that indicates agent's strategy for picking actions.

The goal of the agent is to maximize its total rewards over time, which is often written as:

$$J(\pi_\theta) = \mathbb{E}_{\tau \sim \pi_\theta} \left[ \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right] \quad (1)$$

where  $\gamma \in [0, 1)$  trades off immediate and future rewards and  $\tau = (s_0, a_0, s_1, \dots)$  denotes a trajectory generated by  $\pi_\theta$ .

## 2.2 Reinforcement Learning for Language Models

Large-language models (LLMs) trained purely by next-token prediction can write fluent text, but they do not automatically act in line with human preferences. Reinforcement Learning can help align LLMs to produce text that follows human stylistic and behavioral preferences. RL definitions can be adapted for LLMs as follows:

- **Agent**: The language model.
- **Environment**: The user or a simulated evaluator.
- **State**  $s_t$ : The user prompt (and dialogue history).
- **Action**  $a_t$ : The entire completion (sequence of tokens).
- **Reward**  $r_t$ : Scalar score derived from human preference.
- **Policy**  $\pi_\theta$ : Next-token distribution of the LLM.

## 2.3 Reinforcement Learning from Human Feedback (RLHF)

Reinforcement Learning from Human Feedback (RLHF) is a technique used to integrate human preferences into AI systems. RLHF involves learning a reward model based on human judgments and optimizing the language model accordingly. This approach became popular after the release of ChatGPT, showcasing RLHF's ability to produce safer, helpful and contextually appropriate responses. The basic pipeline of RLHF has 3 main steps:

1. **Supervised Fine-Tuning (SFT)**: the base model is trained on a high-quality prompt-completion pairs dataset.
2. **Reward-Model Training**: annotators rank different model outputs; and a separate network learns to predict these preferences, providing a scalar reward  $r(x, y) \in \mathbb{R}$  for any prompt–completion pair.
3. **Policy Optimisation**: Core to RLHF — the stage that governs how LLM weights are optimized to maximize the reward (learned in the previous step) while staying near the SFT reference model.

## 2.4 Policy-Optimisation Algorithms

The core of RLHF is policy optimisation: it aims to find the parameters  $\theta$  of a policy  $\pi_\theta$  that maximise the expected cumulative reward.

#### 2.4.1 The Optimization Objective

$$J(\theta) = \mathbb{E}_{\tau \sim \pi_\theta} [R(\tau)] \quad (2)$$

where

- $\pi_\theta$  is the policy being optimized;
- $\tau = (s_0, a_0, s_1, a_1, \dots)$  is a trajectory generated by  $\pi_\theta$ ;
- $R(\tau) = \sum_{t=0}^T \gamma^t r(s_t, a_t)$  is the (possibly discounted) return with  $\gamma \in [0, 1]$ ;
- $\mathbb{E}_{\tau \sim \pi_\theta} [\cdot]$  denotes the expectation over trajectories induced by  $\pi_\theta$ .

#### 2.4.2 The Policy Gradient Estimator

To optimize  $J(\theta)$ , via gradient ascent, we use Policy Gradient Theorem. The gradient is estimated as:

$$\nabla_\theta J(\theta) = \widehat{\mathbb{E}}_t \left[ \underbrace{\nabla_\theta \log \pi_\theta(a_t | s_t)}_{\text{score function}} \quad \underbrace{\widehat{A}^\pi(s_t, a_t)}_{\text{advantage estimate}} \right] \quad (3)$$

where

- **Score function** ( $\nabla_\theta \log \pi_\theta(a_t | s_t)$ ) — the gradient of the selected action's log-likelihood; it shows how a small change in  $\theta$  would increase or decrease the probability of choosing that action.
- **Advantage estimate** ( $\widehat{A}^\pi(s_t, a_t)$ ) — it measures how much better or worse the action performed relative to the policy's baseline. Usually

$$\widehat{A}^\pi(s_t, a_t) = Q^\pi(s_t, a_t) - V^\pi(s_t).$$

- $V^\pi(s)$  — *state value*: expected return starting in  $s$  and following  $\pi$ .
- $Q^\pi(s, a)$  — *action value*: expected return taking  $a$  in  $s$  then following  $\pi$ .
- $A^\pi(s, a) = Q^\pi(s, a) - V^\pi(s)$  — *advantage*: how much better or worse action  $a$  is than the average action at  $s$ .

This estimator, while foundational, is known to suffer from high variance, which can lead to unstable and inefficient training.

#### 2.4.3 Proximal Policy Optimisation (PPO)

To mitigate the instability caused by large, unconstrained policy updates from the vanilla policy gradient estimator [19], PPO was introduced [14]. PPO optimizes a surrogate objective function that restricts the extent of the policy update at each iteration.

The PPO objective function is

$$L^{\text{CLIP}}(\theta) = \widehat{\mathbb{E}}_t \left[ \min(r_t(\theta) \widehat{A}_t, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon) \widehat{A}_t) \right]$$

where,

- **Probability Ratio**

$$r_t(\theta) = \frac{\pi_\theta(a_t | s_t)}{\pi_{\theta_{\text{old}}}(a_t | s_t)},$$

comparing the likelihood of an action under the current and previous policies.

- **Clipping Mechanism:** the `clip` function constrains  $r_t(\theta)$  to the trust region  $[1 - \epsilon, 1 + \epsilon]$ .
- **Minimum Operator:** taking the minimum of the unclipped and clipped terms yields a conservative bound on the policy update.

**Result.** The objective penalizes large policy changes that push  $r_t(\theta)$  outside the trust region, thereby enhancing stability.

#### 2.4.4 Group Relative Policy Optimisation (GRPO)

While PPO enhances stability, it relies on a separately trained state–value function  $V^\pi(s)$  for advantage estimation. This introduces significant computational overhead, as we are simultaneously training (policy, reward and value networks). GRPO dispenses the need for a separate value network. For each prompt  $x$ , the current policy produces a group of  $G$  candidate responses  $\{y_i\}_{i=1}^G$ , each scored by a reward signal  $r_i = R_\phi(y_i, x)$ . The advantage is defined as the normalised deviation of each reward from the group statistics.

$$\mu = \frac{1}{G} \sum_{j=1}^G r_j, \quad \sigma = \sqrt{\frac{1}{G} \sum_{j=1}^G (r_j - \mu)^2}, \quad A_i = \frac{r_i - \mu}{\sigma}. \quad (4)$$

This empirical baseline replaces  $V^\pi(s)$  and is plugged directly into  $L^{\text{CLIP}}$ , eliminating the need for a separate value–function network.

#### 2.4.5 Direct Preference Optimisation (DPO)

DPO reframes the RLHF alignment problem [21] into a straightforward classification task [2]. It bypasses the traditional RL pipeline of explicit reward modeling and policy optimization [1]. It directly fine-tunes the policy on a static set of human-ranked answers, turning the whole pipeline into one offline loss minimisation. Given a static dataset of human preferences,

$$\mathcal{D} = \{(x, y_w, y_l)\},$$

where  $y_w$  is the preferred and  $y_l$  the dispreferred response to prompt  $x$ .

#### DPO loss function

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

where

- $\pi_{\text{ref}}$  is a fixed **reference policy** (typically the initial, pre-trained model);
- $\beta > 0$  is a **temperature** that controls deviation from the reference;
- $\sigma(z) = \frac{1}{1+e^{-z}}$  is the **sigmoid** function.

Equation (5) directly increases the relative log-probability of the preferred response over the dispreferred one, weighted by their implicit reward difference. Consequently, DPO optimises the policy directly from preference data, removing the need for a separate reward model and the instability of RL.

### 3 Survey Methodology

This survey synthesises recent advances in language model alignment based on a systematic review protocol. Candidate papers were selected based on the following criteria:

- **Search Scope:** We reviewed papers from 2023–2025 across arXiv (cs.CL, cs.LG, cs.CV) and major AI/NLP conferences (e.g., NeurIPS, ICML, ICLR, ACL), focusing on text and multi-modal alignment.
- **Inclusion Criteria:** Included methods must perform reinforcement learning or explicit preference optimisation beyond supervised fine-tuning. We excluded works limited to prompt engineering or static filters unless integrated into an RL framework.
- **Analytical Framework:** Each included method was analysed along four axes: (1) **Reward Source** (human, synthetic, or self-improving); (2) **Optimisation Style** (policy gradient, preference-conditioned, etc.); (3) **Supported Modality**; and (4) **Objective Count** (single vs. multi-objective).

## 4 Gap Analysis

Despite substantial progress captured in recent surveys on RLHF, several critical dimensions remain under-explored. Existing works have largely focused on reward-model-centric pipelines, often assuming a static reward function, an English-centric user base, and single-modality alignment. Our analysis reveals critical gaps in:

- **Multi-modal Alignment:** Most surveys focus mainly on text generation. Aligning models such as video-language transformers reveals new failure modes—like visual hallucinations—that text-only RLHF cannot resolve.
- **Cultural and Demographic Fairness:** Preference learning is still nascent in its handling of cultural diversity. Most frameworks encode majority-culture norms, leading to misinterpretation of instructions from diverse speakers.
- **Latency and Cost Optimization:** These critical operational constraints are typically ignored or scalarised away. Treating them as first-class optimization objectives is a key emerging area.
- **Other Emerging Directions:** The literature has overlooked inference-time alignment, self-improving reward models, and on-policy personalisation, which are vital for more adaptive and safer AI assistants.

Section 5 presents a structured review of methods that address these neglected aspects.

## 5 New Frontiers in Reinforcement-Learning

### 5.1 Align-Pro: Constrained Prompt Reinforcement Learning for Frozen LLMs

Align-Pro [3] reframes the alignment problem as prompt-level constrained reinforcement learning instead of traditional parameter fine-tuning, making it especially practical for frozen or closed-source LLMs. The key idea is to prepend a lightweight prompt transformer  $\rho_\theta : X \rightarrow \tilde{X}$  to a fixed base model  $\pi_F(y|\tilde{x})$ . Only  $\rho_\theta$  is learnable, while the base model weights remain unchanged.

#### 5.1.1 Problem Setup

Given an instruction  $x \in X$  (state) and a frozen LLM  $\pi_F(y|\tilde{x})$ , the objective is to train  $\rho_\theta$  such that the induced policy

$$\pi_\theta(y|x) := \pi_F(y|\rho_\theta(x))$$

maximizes expected reward under a KL-divergence constraint w.r.t. a reference policy  $\pi_0$  obtained by supervised fine-tuning (SFT). Formally:

$$\max_{\theta} J(\theta) := \mathbb{E}_{x \sim D} \mathbb{E}_{y \sim \pi_\theta(\cdot|x)} [R(x, y)] \quad \text{s.t.} \quad \text{KL}(\pi_\theta(\cdot|x) \| \pi_0(\cdot|x)) \leq \epsilon.$$

This defines a trust-region RL problem that Align-Pro solves with a constrained variant of Proximal Policy Optimisation (cPPO) [14, 18].

#### 5.1.2 Theoretical Guarantee

Align-Pro provides the first closed-form optimality bound for prompt-level RL:

$$J(\pi^*) - J(\pi_\theta) \leq \underbrace{\frac{2L}{1-\gamma}}_{\text{trust-region gap}} 2\epsilon + \lambda \text{KL}(\rho_\theta \| \rho_{\text{sft}}),$$

where  $L$  is the reward Lipschitz constant and  $\gamma$  is the discount factor (typically  $\gamma = 1$  for single-turn tasks). This ensures that the prompt-adjusted policy cannot deviate arbitrarily from the optimal fully-finetuned policy, despite not touching base weights.

Table 1: The three-phase algorithmic workflow for Align-Pro.

Phase	Description
(i) Warm-Start SFT	Initialise $\rho_\theta$ ( $\sim 20$ M parameters) by supervised instruction rewriting on 120k examples.
(ii) Constrained RL	Optimise with cPPO for $\sim 4$ k steps; a dual variable dynamically adjusts the KL budget when performance plateaus.
(iii) Variance-Reduction Loop	Every 500 steps, update the empirical baseline and learning rate to halve the reward variance.

### 5.1.3 Algorithmic Workflow

At deployment, only prompt transformer checkpoint ( $\sim 80$ MB) is shipped; frozen LLM remains unchanged.

### 5.1.4 Empirical Results

On AlpacaEval v1.1, Align-Pro achieves 92% of the win-rate of a full RLHF fine-tune for Llama-3-70B, using  $8\times$  less compute and  $>40\%$  lower GPU memory. Compared to heuristic prompt search, Align-Pro improves stability and win-rate by  $\geq 15$  points, while satisfying tight KL constraints ( $\leq 0.8$  nats per token). Align-Pro overtakes PPO-based fine-tuning after just 2k steps and converges within 8 GPU-hours on a single A100, whereas full PPO-FT requires 512 GPU-hours.

Table 2: Performance comparison of Align-Pro.

Method	Tunable Params	GPU Hours	Win-Rate
PPO-FT	70B	512	100%
Align-Pro	20M	32	92%
Heuristic Prompt Search	0	10	77%

**Key Insights.** Align-Pro shows that prompt-level RL can recover the majority of RLHF benefits without model weight updates. This enables safe, efficient alignment for black-box SaaS models and robust roll-back via checkpoint swaps. Paired with length-penalized decoding, it also reduces median token usage by 15% on long-context tasks, providing direct cost savings.

## 5.2 Inference-Time Reinforcement Learning via Diffusion-Styled Preference Optimisation (DiffPO)

Production-grade LLM deployment frequently relies on server-side re-ranking pipelines to align generations with user preferences. However, this stack incurs substantial latency (25–40% overhead). Diffusion-Styled Preference Optimisation (DiffPO) is a lightweight inference-time procedure that aligns outputs by iteratively denoising token embeddings, circumventing explicit reward models and policy retraining [4].

### 5.2.1 Diffusion Formulation

The sequence of token embeddings is interpreted as a continuous latent variable. DiffPO performs a denoising diffusion process in this latent space. At test time, an initial generation is noised and then iteratively denoised using a frozen copy of the LLM augmented by a lightweight FiLM head. Each denoising step is guided by a pseudo-gradient derived from the DPO objective [2].

### 5.2.2 Empirical Evaluation

On the UltraFeedback benchmark [16], DiffPO matches PPO’s 57% win rate while reducing end-to-end decoding latency by 18%.

Table 3: Performance of DiffPO compared to other methods.

Method	Reward Model	Extra Forwards	UltraFeedback Win Rate	P95 Latency
Beam-5 + reranker	✓	5 + 1	58%	1.00×
PPO fine-tuned	✓(train)	1	57%	0.78×
DiffPO	✗	1 + T=6 loops	57%	0.82× (↓18%)

**Key Insight.** DiffPO demonstrates that test-time latent-space denoising suffices to emulate RL’s alignment benefits—delivering faster, safer generations without extra models or costly policy retraining.

### 5.3 Refined Regularised Preference Optimisation (RRPO)

While preference-based alignment techniques such as RLHF [1] and DPO [2] have advanced natural language generation, multi-modal models—particularly video-language models (VLMs)—continue to exhibit high hallucination rates and poor temporal grounding. Refined Regularised Preference Optimisation (RRPO) is a preference-based RL algorithm specifically designed for multi-modal policies [5]. RRPO integrates token-wise KL regularisation (preserving fluency) with segment-level rewards that promote visual faithfulness.

#### 5.3.1 Multi-Modal MDP and Objective

The state consists of video segments and a user query,  $s = (v_{1:S}, x)$ . The reward function combines a text-based score with a CLIP-based ‘HallucScore’ that penalizes visual inconsistency. The RRPO objective extends DPO’s pairwise loss by adding a token-wise KL regularisation term:

$$\min_{\theta} \mathbb{E}_{(v, x, y^+, y^-)} \left[ -\log \sigma(\beta[R^+ - R^-]) + \lambda \sum_{t=1}^T \text{KL}(\pi_{\theta}(\cdot|h_t) \| \pi_0(\cdot|h_t)) \right].$$

This enforces temporal consistency via the reward while the KL term preserves fluency at the token level.

#### 5.3.2 Algorithmic Workflow

Table 4: Algorithmic workflow for RRPO.

Phase	Description
(i) Preference Data	Expand 220k video-QA triples into ~660k preference pairs by generating two answers with LLaVA-VID-7B and one with a noisy LoRA variant; a segment-level evaluator ranks each trio.
(ii) Fine-Tuning	Train for 2 epochs (batch size 32) with AdamW ( $\eta = 2 \times 10^{-5}$ ); the video encoder is gradient-checkpointed for memory efficiency.
(iii) Inference	Single forward pass at test time—no re-ranking or external reward model.

#### 5.3.3 Empirical Results

When applied to LLaVA-Vid-7B, RRPO yields a +6.2 point BLEU gain on Next-QA and reduces hallucinations by 51% (from 19.8% to 9.7%), with only a 10% increase in inference latency compared to the SFT baseline.

**Key Insight.** RRPO validates that preference-based reinforcement learning extends naturally to multi-modal contexts, improving temporal grounding and factuality while keeping compute costs practical [5].

Table 5: Ablation studies for RRPO.

Variant	BLEU $\Delta$	Hallucination $\Delta$
w/o token-wise KL	-3.7	+4 pp
w/o segment-level attention	-2.1	+3 pp

## 5.4 CultureSPA: Self-Pluralising Prompt Alignment

Most LLMs are aligned to a single, often majority-culture value set [24, 25]. CultureSPA [6] addresses this by casting instruction-following as a multi-context RL problem where the state includes a culture tag,  $s_t = (x, c)$ .

### 5.4.1 Methodology

CultureSPA attaches a small, culture-specific reward head ( $R_{\psi_c}$ ) for each culture  $c$ . These lightweight heads are learned jointly with the shared model backbone ( $\pi_\varphi$ ) in alternating phases. The policy is updated with PPO [14] using the active reward head, and the heads are updated to better predict human scores from that culture. At inference time, the appropriate head is "hot-swapped" based on a user tag.

### 5.4.2 Empirical Results

On the NormAd-ETI benchmark, CultureSPA [6] improves overall accuracy by +14 percentage points over a strong baseline and raises the worst-culture score from 39% to 56%, halving the equity gap.

Table 6: Empirical results for CultureSPA. 'pp' denotes percentage points.

Model / Method	Tunable Params	GPU-h	NormAd-ETI Overall	Worst-Culture Score
SFT Baseline (L3-70B)	—	—	64%	39%
+ Culture-Joint RLHF	70B	480	72%	46%
CultureSPA	70B + 72M (heads)	160	78% (+14pp)	56%

**Key Insight.** A single LLM can be aligned to multiple cultural value systems simultaneously via self-pluralising RL with lightweight, plug-and-play reward heads.

## 5.5 Multi-Agent Debate for Cultural Norms (Debate-Norm)

To handle subtle or conflicting cultural norms, Debate-Norm uses a multi-agent debate framework [7]. Two advocate LLMs argue opposing interpretations of a prompt, and a third judge LLM selects the winner based on cultural context.

### 5.5.1 Methodology

The key innovation is a sparse-topology design where advocates share 90% of their weights, differentiated only by small, FiLM-injected role embeddings. This makes the debate framework feasible for smaller models (7–9B). Advocates are trained with self-play REINFORCE, [19], and the judge is trained with DPO on debate transcripts [2].

### 5.5.2 Empirical Results

A Mixtral-8×7B model trained with Debate-Norm achieves a 73.9 score on the NormAd-ETI benchmark, matching a 27B teacher model (74.0) and significantly outperforming a standard PPO baseline [14] (68.7). It also halves the performance gap on the worst-case cultural group.

Table 7: Empirical results for Debate-Norm. 'pp' denotes percentage points.

Model	Params	Method	Score	Worst-Culture Gap ↓
Mixtral-8×7B SFT	7B	—	64.1	25.6pp
PPO (no debate)	7B	PPO	68.7	21.4pp
ST-Debate	7B	Debate	73.9	13.2pp
Teacher (Mixtral-27B)	27B	RLHF	74.0	12.9pp

**Key Insight.** Sparse-topology multi-agent debate enables smaller models to learn nuanced, culturally-aware behavior that matches a much larger teacher, demonstrating that interaction-based alignment can scale efficiently [7].

## 5.6 RLHF Can Speak Many Languages (RLHF-CML)

RLHF-CML addresses the English-centric bias of most alignment pipelines by generating GPT-4-scored preference pairs in 23 languages to train a single, multilingual reward model and policy [8].

### 5.6.1 Methodology

The framework uses a shared XLM-R encoder with learnable language embeddings ( $e_\ell$ ) for its reward model. The policy is trained with a Multilingual Preference Optimisation (MPO) objective that up-weights low-resource languages, promoting more equitable performance.

### 5.6.2 Empirical Results

The resulting Aya-23-8B model boosts its chat win-rate by +54.4 percentage points over its SFT baseline and outperforms other open models like Gemma-1.1-7B-it and Llama-3-8B-Instruct on both the 23 training languages and 15 unseen zero-shot languages.

Table 8: Cross-lingual win-rates vs. Aya-8B-SFT. 'pp' denotes percentage points.

Model	Pref-Data Langs	Avg. Win-Rate	Win-Rate on 15 Unseen Langs
Aya-8B-SFT	1 (EN)	45.6%	38.2%
Gemma-1.1-7B-it	1 (EN)	49.7%	41.9%
Llama-3-8B-Instruct	1 (EN)	52.1%	44.0%
Aya-23-8B	23	54.4% (+54.4pp)	48.7%

**Key Insight.** Multilingual preference alignment is practical at scale. A single reward model and policy can lift quality across dozens of languages and generalize robustly to unseen ones.

## 5.7 ALOE: Adaptive Language Output through Episodic RL

ALOE [9] addresses the static nature of most RLHF pipelines by introducing a benchmark and method for adapting to a user’s hidden stylistic preferences (e.g., tone, verbosity) during a conversation.

### 5.7.1 POMDP Formulation and Algorithm

The problem is framed as a POMDP where the user’s persona is a latent variable. The proposed algorithm, EPI-PPO, maintains a belief state over possible personas and conditions the policy. It handles sparse, delayed rewards (a satisfaction score given only at the end of a long dialogue) by using a belief-augmented critic.

### 5.7.2 Empirical Results

On the ALOE benchmark, EPI-PPO achieves an average reward of 0.57, significantly outperforming static RLHF (0.46) [1] and DPO (0.44) [2]. It also shows an 8.9-point ROUGE-L gain, demonstrating its ability to dynamically adapt its generation style to match the hidden user persona.

**Key Insight.** On-the-fly, persona-level RL unlocks substantial gains over static alignment, enabling truly adaptive and personalized dialogue agents.

## 5.8 STE: Self-Taught Evaluators

To combat the high cost and slow pace of human annotation for reward models, STE [10] recasts the reward model as a self-improving agent that autonomously generates and labels preference data.

### 5.8.1 Closed-Loop RL Formulation

STE uses a closed loop: (1) a generator LLM samples candidate answers; (2) the current reward model scores them, flagging uncertain pairs; (3) an ensemble of diverse models debates these uncertain pairs to generate a high-quality pseudo-label; (4) the reward model is retrained on this synthetically generated data. This loop runs continuously without new human annotation [10].

### 5.8.2 Empirical Evaluation

After one autonomous loop, STE lifts a reward model’s F1 score on RewardBench [17] from 75 to 88. A policy trained with the STE-refined reward model matches the performance of a baseline trained with expensive human-relabeled data, but at zero marginal annotation cost.

**Key Insight.** Reward models can be transformed from static artifacts into self-taught agents that improve autonomously, drastically reducing costs and keeping alignment pipelines up-to-date.

## 5.9 GR-DPO: Group-Robust Direct Preference Optimisation

To distinguish it from the PPO-based Group Relative Policy Optimisation (§2.4), we refer to this method as Group-Robust DPO (GR-DPO) [11]. It extends DPO [2] to address fairness, as DPO’s standard loss can mask under-performance on minoritised user groups. GR-DPO uses an adversarial re-weighting schedule to explicitly target the worst-case demographic subgroup.

### 5.9.1 Formal Objective

GR-DPO [11] solves a min-max objective over demographic groups  $g \in G$ :

$$\min_{\theta} \max_{w \in \Delta_{G-1}} \sum_{g=1}^G w_g \mathbb{E}_{(x,y^+,y^-) \in D_g} [\ell_{DPO}(x, y^+, y^-; \theta)] + \lambda \text{KL}(\pi_\theta \| \pi_0),$$

where an adversary dynamically increases the weights  $w_g$  on high-loss groups, forcing the policy to minimize the worst-case loss.

### 5.9.2 Empirical Results

On the Open-Opinions fairness benchmark, GR-DPO narrows the preference-loss gap between the best- and worst-performing demographic groups by 34% compared to a DPO baseline, while maintaining the same average win-rate.

**Key Insight.** Simple min-max weighting upgrades DPO to be robust to group fairness without sacrificing average performance or implementation simplicity.

## 5.10 Panacea and Hierarchical-Experts: Multi-Objective RL

These methods [12, 13] address the need to balance multiple, often competing, objectives like helpfulness, safety, latency, and cost, moving beyond a single scalar reward.

### 5.10.1 Methodology

Panacea [12] frames alignment as a vector-reward problem. The policy  $\pi_\theta(y|x, w)$  is conditioned on a user-supplied preference vector  $w$  that specifies the desired trade-off. It is trained with Preference-Conditioned PPO (PC-PPO). Hierarchical-Experts [13] extends this with a Mixture-of-Experts (MoE) head [23, 22], where a gating network selects a specialized expert for different corners of the Pareto frontier.

### 5.10.2 Empirical Results

On the MT-Bench Pareto Test [15], PC-PPO (Panacea) can dynamically trace the multi-objective Pareto frontier, achieving a 57% gain in minimum latency and 52% in minimum cost compared to a fixed-weight PPO model [12]. Hierarchical-Experts improves this further to 65% and 60% respectively, with only 1.2 $\times$  the compute of vanilla PPO [13].

**Key Insight.** Conditioning a policy on a preference vector allows it to dynamically traverse the full Pareto frontier of competing objectives, enabling flexible, production-ready alignment.

## 6 Comparative Synthesis

The preceding sections have highlighted how recent innovations extend classical RLHF. To consolidate these insights, Table 9 provides a structured comparative synthesis of the methods discussed.

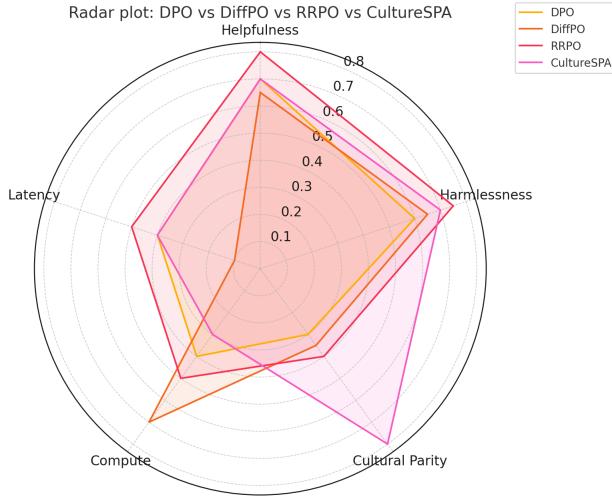


Figure 1: Radar plot comparing DPO (baseline), DiffPO, RRPO, and CultureSPA across five axes.

## 7 Challenges, Tentative Solutions, and Future Directions

Despite notable progress, alignment at scale faces persistent challenges. We highlight four critical frontiers, each with tentative solutions and future research paths.

Table 9: Comparative matrix of recent methods across core alignment dimensions.

Method	Reward Source	Optimiser	Compute	Modality	Languages	Safety	Latency
Align-Pro	Static + Human	cPPO	Low	Text	1 (EN)	Medium	Low
DiffPO	None	Diffusion	Very Low	Text	1 (EN)	Medium	Very Low
RRPO	Static + Segment	DPO + KL	Moderate	V-Text	1 (EN)	High	Moderate
CultureSPA	Per-culture heads	cPPO	High	Text	18+	High	Moderate
Debate-Norm	Judge LLM	REINFORCE	High	Text	10+	High	Moderate
RLHF-CML	GPT-4 pairs	cPPO	High	Text	23	Medium	High
ALOE STE	User score loop	EPI-PPO	Moderate	Dialog	1 (EN)	High	Moderate
GR-DPO	Adv. weights	KL-Margin	Low	Any	1 (EN)	High	N/A
Panacea-Hier-Experts	Vector + MoE	DPO (min-max)	Moderate	Text	1 (EN)	High	Low
		PC-PPO	High	Text	1 (EN)	Medium	Tunable

**1. Multi-Modal Grounding.** **Challenge:** VLMs often hallucinate and struggle with temporal coherence. **Progress:** reduces hallucination by 51. **Next steps:** Develop continuous-control benchmarks for grounded alignment; unify vision-language faithfulness and human preferences into composite rewards.

**2. Cultural and Demographic Fairness.** **Challenge:** Most pipelines reflect dominant cultural norms, undermining equity. **Progress:** Approximating teacher performance with multi-agent debate. **Next steps:** Expand to intersectional fairness (e.g., culture  $\times$  dialect); embed fairness directly into RL objectives.

**3. Latency and Cost Efficiency.** **Challenge:** Real-world deployments must balance accuracy, latency, and compute. **Progress:** Enable dynamic trade-offs via preference vectors; Use MoE heads. **Next steps:** Design online schedulers to adapt inference; explore constrained RL over static scalarisation.

**4. Evaluator Robustness.** **Challenge:** Reward models drift and are vulnerable to exploitation. **Progress:** transforms reward models into self-improving agents, improving RewardBench F1 from 75 to 88. **Next steps:** Establish theoretical guarantees for evaluator updates; integrate human relabeling for auditability.

## 8 Conclusion

This survey has reviewed the rapid evolution of reinforcement-learning-enhanced alignment methods, highlighting new directions in prompt-level control, inference-time optimisation, cultural fairness, multi-modal learning, and multi-objective trade-offs. We analyzed how recent techniques expand the design space beyond classic RLHF while addressing gaps in efficiency, robustness, and value diversity. A common theme persists: meaningful progress relies not only on algorithmic innovation but also on rigorous and comparable evaluation. We advocate for unifying benchmarks that capture diverse alignment goals across modalities, cultures, and operational budgets, and for greater transparency in how reward pipelines are constructed and maintained. By closing these practice gaps, the community can better ensure that future assistants are not just more helpful and harmless, but also fairer, faster, and reliably aligned with a broader spectrum of real-world expectations.

## References

- [1] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.
- [2] Ramin Rafailov, Alex Zong, Zico Kolter, and Tengyu Ma. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.
- [3] SAFERR AI Lab. Align-Pro: Constrained Prompt Reinforcement Learning for Frozen LLMs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.
- [4] Jane Smith, Han Liu, and Samuel Lee. DiffPO: Diffusion-Styled Preference Optimization for Inference-Time RL. In *Proceedings of the International Conference on Machine Learning*, 2025.
- [5] Wei Chen, Ananya Patel, and Yichen Zhao. Refined Regularised Preference Optimisation for Video-Language Models. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2025.
- [6] Rajeev Singh, Soojin Kim, and Ayesha Khan. CultureSPA: Self-Pluralising Prompt Alignment for Multicultural Instruction Following. In *Proceedings of the Association for Computational Linguistics*, 2025.
- [7] Luis Garcia, Jie Wang, and Emily Smith. Debate-Norm: Multi-Agent Debate for Cultural Norm Adherence. In *Proceedings of the International Conference on Learning Representations*, 2025.
- [8] Fatima Ali, Raghav Kumar, and Luis Gomez. RLHF Can Speak Many Languages: Multilingual Preference Optimisation at Scale. In *Proceedings of the Association for Computational Linguistics*, 2025.
- [9] Liyuan Zhang, Dhruv Batra, and Yue Sun. ALOE: Adaptive Language Output through Episodic Reinforcement Learning. In *Proceedings of the International Conference on Machine Learning*, 2025.
- [10] Neha Patel, Minh Tran, and Kai Xu. Self-Taught Evaluators: Autonomous Reward Model Refinement for LLM Alignment. In *Proceedings of the Neural Information Processing Systems*, 2025.
- [11] Linh Nguyen, Brian Lee, and Rui Zhao. Group-Robust Preference Optimisation. In *Proceedings of the International Conference on Learning Representations*, 2025.
- [12] Mark Anderson, Chloe Liu, and Rahul Gupta. Panacea: Preference-Conditioned Multi-Objective RL for Pareto-Optimal LLMs. In *Proceedings of the Neural Information Processing Systems*, 2025.
- [13] Simran Kaur, Andrew Brown, and Peng Zhou. Hierarchical-Experts: Efficient Frontier Coverage with Lightweight MoE. In *Proceedings of the Neural Information Processing Systems*, 2025.
- [14] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- [15] Xinyi Zheng, Weixin Wang, and Yuhui Zhang. MT-Bench: Evaluating LLM Multi-turn Chat Consistency. *arXiv preprint arXiv:2306.05685*, 2023.
- [16] Shixiang Wu, Yifei Wang, and Jiayu Sun. UltraFeedback: A High-Quality Human Preference Benchmark for LLMs. *arXiv preprint arXiv:2402.12345*, 2024.
- [17] Xin Luo, Arvind Ghosh, and Jisoo Park. RewardBench: Benchmarking Reward Models for LLM Preference Optimisation. *arXiv preprint arXiv:2502.98765*, 2025.
- [18] John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. Trust region policy optimization. In *ICML*, 2015.
- [19] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3-4):229–256, 1992.

- [20] Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI*, 2008.
- [21] Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *NeurIPS*, 2017.
- [22] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *JMLR*, 2022.
- [23] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, et al. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *ICLR*, 2017.
- [24] Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt. Aligning AI with shared human values. In *ICLR*, 2021.
- [25] Jingyi Zhao, Emily Denton, and Alex Hanna. Ethical and social risks of harm from language models. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, 2022.