

# RARO: Reasoning via Adversarial Inverse RL from Demonstrations

2026-03-05

## Paper Info

- **Authors:** Locke Cai, Ivan Provilkov
- **Link:** [arXiv](#)
- **Read date:** 2026-03-05

## Summary

Most RL-based reasoning training for LLMs requires task-specific verifiers (e.g. checking math answers), but most real-world tasks lack such verifiers. RARO (Relativistic Adversarial Reasoning Optimization) addresses this by using inverse reinforcement learning to train reasoning from expert demonstrations alone. A shared LLM serves as both policy and critic: the policy generates reasoning traces while the critic learns to distinguish expert from policy-generated answers in a relativistic pairwise manner. Joint training via RL with stabilization techniques (replay buffer, tie rewards, loss weighting) yields strong results across verifiable tasks (math, arithmetic) and non-verifiable tasks (poetry), approaching oracle-verifier RL performance on verifiable benchmarks without ever using a verifier.

## Key Ideas

1. **Verifier-free reasoning training** — Most tasks lack ground-truth verifiers; RARO removes this dependency by learning rewards from expert demonstrations via IRL.
2. **Relativistic critic** — Instead of absolute scoring, the critic compares policy and expert outputs pairwise, determining which is better or declaring a tie. This prevents degeneracy when the policy approaches expert quality.

3. **Shared policy-critic architecture** — Both components use the same LLM, reducing memory overhead and enabling “data mixing” where policy and critic rollouts combine in single batches.
4. **Stabilization techniques** — Replay buffer (prevents critic forgetting), tie rewards with hyperparameter (provides stability near expert quality), and loss weighting (balances objectives).
5. **Emergent behaviors** — Self-correcting search in arithmetic; explicit planning in poetry generation—behaviors not present in SFT baselines.

## Method

- **IRL formulation:** Rather than maximizing marginal likelihood of expert CoT traces (intractable), learn a parameterized reward function where policy optimization yields near-optimal performance.
- **Adversarial game:** Policy tries to generate outputs indistinguishable from expert; critic tries to distinguish them. Joint training converges when policy matches expert distribution.
- **Relativistic comparison:** Critic sees (question, answer\_A, answer\_B) and outputs which is better or tie. This avoids reward collapse when policy quality approaches expert level.
- **Shared LLM:** Single model with different prompts/heads for policy and critic roles. Reduces 2× memory cost to ~1×.
- **Training loop:** Alternate between policy rollouts (optimized with critic rewards via RL) and critic updates (trained to discriminate expert vs. policy). Replay buffer stores past policy outputs for critic stability.

## Results

- **Countdown (arithmetic):** 54.4% accuracy vs. 40.7% SFT, nearly matching RLVR oracle at 57.7%.
- **DeepMath (103k math problems):** 57.5% at 7B vs. 49.3% best baseline; consistent gains across 1.5B–7B scales.
- **Poetry (40k poems, non-verifiable):** 25.0% win-rate vs. expert (vs. 5.9% SFT); 77.3/100 score at 7B.
- **Scaling:** Stable training dynamics with continuous reward/length growth; similar scaling trends to oracle verifier methods.
- **Baselines beaten:** Outperforms SFT, logit-based rewards, and preference optimization (DPO-style) approaches.

## Strengths

- Addresses a real bottleneck: most tasks lack verifiers, so verifier-free reasoning training is highly valuable.
- Strong empirical results across both verifiable and non-verifiable domains.
- Shared architecture is practical—no need for separate reward model.
- Relativistic critic design is elegant and well-motivated (avoids reward saturation).
- Demonstrates emergent reasoning behaviors (self-correction, planning) beyond what SFT achieves.

## Weaknesses / Limitations

- Sample efficiency lags behind RLVR on verifiable tasks—requires more training iterations to match oracle verifier performance.
- Training instability observed in long-form creative domains.
- Only tested up to 7B parameters; unclear how it scales to larger models.
- Requires “abundant expert demonstrations”—data requirements not deeply analyzed.
- Critic interpretability is limited; hard to understand what the learned reward captures.
- Comparison to recent RLHF/RLAIF methods (which also don’t need task verifiers) could be more thorough.

## Relevance to My Work

- **GRPO without verifiers** — RARO offers an alternative when ground-truth rewards aren’t available. If working on tasks where outcome verification is hard (e.g. clinical reasoning in MedEnv), this could be a practical path.
- **Self-distillation connection** — The adversarial framework is related to self-distillation: the model learns from expert traces but through RL rather than SFT, preserving exploration. Connects to ideas about when distillation vs. RL is more effective.
- **Reward learning** — The relativistic critic could complement GRPO-style training by providing learned rewards for problems where binary correctness isn’t available.
- **Complements POPE** — Where POPE uses oracle solutions as exploration guides (still needs a verifier for rewards), RARO eliminates the verifier entirely but needs more expert demonstrations.

## Key Takeaways

1. Inverse RL from demonstrations can replace task-specific verifiers for reasoning training, approaching oracle-verifier performance on math while extending to non-verifiable domains.

2. Relativistic pairwise comparison is more stable than absolute scoring for the critic, especially as policy quality approaches expert level.
3. Shared policy-critic architecture makes adversarial IRL practical for LLMs without doubling memory costs.
4. Verifier-free methods are less sample-efficient than RLVR but unlock training on the vast majority of tasks where verifiers don't exist.
5. Emergent reasoning behaviors (self-correction, planning) arise from RL training that don't appear with SFT alone.