The Post-Training Dilemma

Why We Should Rethink the Sequential SFT–RL Paradigm

Xueyan Niu

Based on joint work with Bo Bai, Wei Han, Weixi Zhang

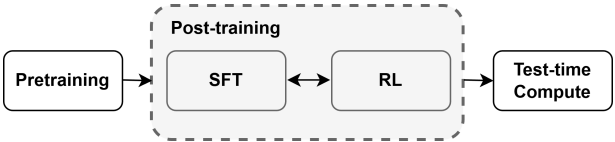
Theory Lab, 2012 Labs, Huawei Technologies

December 2026

NeurIPS 2026 Position Paper

How to Make Better Model?

- During pre-training, the model is trained on vast amounts of textual data, which requires substantial computational resources.
- Models can be improved at specific tasks by **post-training** them.
  - **Post-training** methods include fine-tuning and reinforcement learning.



Questions:

- Does RL improve on prior SFT, and does SFT improve on prior RL?
- Can the two post-training stages therefore be decoupled?

SFT and RL Objectives

SFT loss equivalent expression

$$\mathcal{L}_{\mathrm{SFT}}(\theta) := -\sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathrm{SFT}}} \log\left(p_\theta(\boldsymbol{y}\mid\boldsymbol{x})\right) = \mathbb{E}_{\boldsymbol{x}\sim p_{\mathcal{D}_{\mathrm{SFT}}},\boldsymbol{y}\sim p_{\mathcal{D}_{\mathrm{SFT}}}(\cdot|\boldsymbol{x})}[-\log p_\theta(\boldsymbol{y}\mid\boldsymbol{x})]$$

RL objective and equivalent form

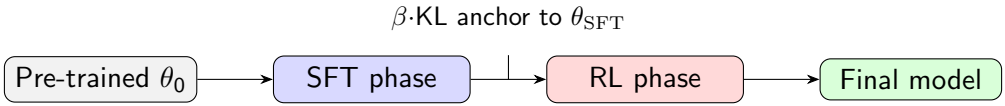
RL objective:

$$\mathcal{J}_{\mathrm{RL}}(\theta) = \mathbb{E}_{\boldsymbol{x}\sim p_{\mathcal{D}_{\mathrm{RL}}},\boldsymbol{y}\sim p_\theta(\cdot|\boldsymbol{x})}[r(\boldsymbol{x},\boldsymbol{y})] - \beta\,\mathbb{E}_{\boldsymbol{x}\sim p_{\mathcal{D}_{\mathrm{RL}}}}[D_{\mathrm{KL}}(p_\theta(\cdot\mid\boldsymbol{x})\|\pi_{\mathrm{ref}}(\cdot\mid\boldsymbol{x}))].$$

The closed-form solution is well known to be

$$p_{\theta_{\mathrm{RL}}} = \frac{1}{Z(\boldsymbol{x})}\pi_{\mathrm{ref}}(\boldsymbol{y}|\boldsymbol{x})\exp\left(\frac{r(\boldsymbol{x},\boldsymbol{y})}{\beta}\right).$$

The De Facto Post-Training Pipeline



Implicit assumption: the two stages *functionally compose* — SFT teaches instruction following, RL refines alignment, the deployed model inherits both.

Position

For rewards misaligned with SFT, RL raises the SFT loss by a term *quadratic* in the parameter displacement. No choice of β, RL algorithm, or training horizon removes it: the two stages should not be designed separately.

Theorem: Non-Decoupling Is Structural

Assumption. $\mathcal{L}_{\mathrm{SFT}}$ satisfies PL ($\mu_{\mathrm{SFT}} > 0$) on a region containing the RL trajectory and the SFT optimum Θ^*_{SFT}.

PL $\Rightarrow$ quadratic growth (Karimi et al. 2016):

$$\mathcal{L}_{\mathrm{SFT}}(\theta_{\mathrm{RL}}) - \mathcal{L}^*_{\mathrm{SFT}} \;\geq\; \frac{\mu_{\mathrm{SFT}}}{2} \operatorname{dist}(\theta_{\mathrm{RL}}, \Theta^*_{\mathrm{SFT}})^2$$

- Damage grows **quadratically** with displacement.
- The KL penalty bounds *how far* the policy drifts in distribution space — it cannot fix the *direction* of the drift.

How Long Should RL Run?

Optimal RL Duration (Proposition 1)

Assume RL runs for T_{RL} steps with learning rate η and mean RL gradient $\bar{g}_{\mathrm{RL}} := \frac{1}{T_{\mathrm{RL}}} \sum_{t=1}^{T_{\mathrm{RL}}} g_{\mathrm{RL}}^{(t)}$, where $g_{\mathrm{RL}}^{(t)} = \nabla \mathcal{L}_{\mathrm{RL}}(\theta^{(t)})$. Defining $\kappa \triangleq \mu_{\mathrm{SFT}} \eta^2 \|\bar{g}_{\mathrm{RL}}\|^2$ for compactness, the total composite loss $\mathcal{L}_{total} = \mathcal{L}_{\mathrm{SFT}}(\theta_{\mathrm{RL}}) + \lambda \mathcal{L}_{\mathrm{RL}}(\theta_{\mathrm{RL}})$ (with trade-off weight $\lambda > 0$) satisfies:

$$\mathcal{L}_{total} \geq \underbrace{\mathcal{L}_{\mathrm{SFT}}^* + \lambda \mathcal{L}_{\mathrm{RL}}(\theta_{\mathrm{SFT}}^*)}_{C \text{ (constant w.r.t. } T_{\mathrm{RL}})} + \frac{\kappa}{2} T_{\mathrm{RL}}^2 - \lambda c_{\mathrm{RL}} T_{\mathrm{RL}}$$

Coupling damage grows as $\frac{\kappa}{2} T_{\mathrm{RL}}^2$; RL benefit (optimistically) linear, $-\lambda c_{\mathrm{RL}} T_{\mathrm{RL}}$. The optimum is $T_{\mathrm{RL}}^* = \frac{\lambda c_{\mathrm{RL}}}{\kappa} = \frac{\lambda c_{\mathrm{RL}}}{\mu_{\mathrm{SFT}} \eta^2 \|\bar{g}_{\mathrm{RL}}\|^2}$

- Flatter SFT basin (small μ_{SFT}) $\Rightarrow$ longer safe RL window;
- RL itself sharpens the landscape, so T_{RL}^* *shrinks as RL proceeds*.

When Can RL Help?

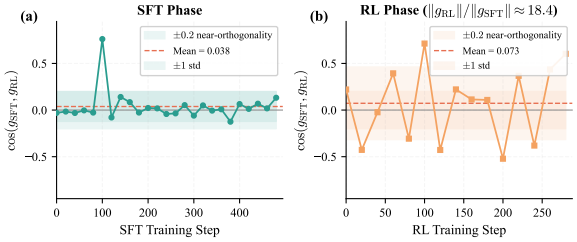
Second-order analysis (Proposition 2)

RL improves the SFT loss when the reward–SFT alignment α_{RS} beats the curvature threshold:

$$\Delta\mathcal{L}_{\mathrm{SFT}} < 0 \quad \Longleftrightarrow \quad \alpha_{\mathrm{RS}} > \tau_{\mathrm{crit}} = \frac{\eta\, T_{\mathrm{RL}}\, \sigma_{\mathrm{eff}}^2\, \|\bar{g}_{\mathrm{RL}}\|}{2\, \|\nabla\mathcal{L}_{\mathrm{SFT}}\|}$$

- Reward aligned with SFT is the only regime where RL can help; otherwise the non-decoupling theorem applies.
- Degenerate endpoint: reward $= -\mathcal{L}_{\mathrm{SFT}}$ means RL is continued SFT, and non-decoupling vanishes ($\mathrm{dist} = 0$).
- SFT flattening lowers τ_{crit}; RL sharpening raises it — the window closes as RL proceeds.

Mechanism I: Gradient Misalignment

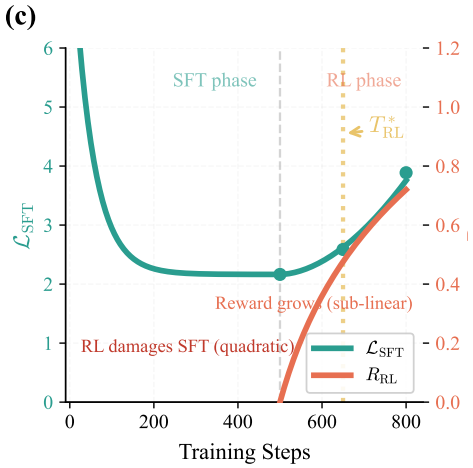


MATH-500, Qwen3-0.6B:

- $\cos(g_{\mathrm{SFT}}, g_{\mathrm{RL}}) \approx 0.04$ in SFT, ≈ 0 in RL
- $\|g_{\mathrm{RL}}\|/\|g_{\mathrm{SFT}}\| \approx 18.4$
- **large, nearly perpendicular** updates

$\Rightarrow$ the two stages optimize different subspaces.

Mechanism II: The Landscape Seesaw



SFT flattens the landscape; RL then re-sharpens it.

- $\|g_{\mathrm{SFT}}\|$: 95.6 $\rightarrow$ 9.4 during SFT, then 15.5 $\rightarrow$ 60.9 during RL
- pointwise PL ratio $\hat{\mu}_{\mathrm{PL}}$: 160 $\rightarrow$ 2799 (17$\times$)

So the bound *tightens* during RL: the same displacement costs progressively more SFT loss — a moving target.

Evidence Across Tasks and Scales

Setting: Qwen3-0.6B, SST-2, REINFORCE.

Steps		Loss			
SFT	RL	Base	SFT	Post-RL	Δ
1000	500	11.364	0.067	4.417	+4.350
2000	500	11.364	0.042	3.117	+3.075
4000	500	11.364	0.051	4.812	+4.761
2000	500	11.364	0.042	3.117	+3.075
2000	1000	11.364	0.034	2.526	+2.492
2000	2000	11.364	0.057	2.849	+2.793

Setting: Qwen2.5-7B, MATH-500, GRPO.

Stage	Test loss	Acc	Δ(loss / acc)
Base	0.817	36%	
SFT → GRPO			
after SFT	0.546	44%	
after GRPO	0.550	34%	+0.004/−10

Why Existing Fixes Fall Short

- **Sequential** (DPO, GRPO): the reward source changes, the two-objective structure does not.
- **Unified** (UFT, KTO, ORPO): optimise $(1-\alpha)\mathcal{L}_{\mathrm{SFT}} + \alpha\mathcal{L}_{\mathrm{RL}}$, whose gradient $(1-\alpha)g_{\mathrm{SFT}} + \alpha g_{\mathrm{RL}}$ is RL-dominated once $\alpha > \frac{1}{1+\|g_{\mathrm{RL}}\|/\|g_{\mathrm{SFT}}\|} \approx 0.05$.
- **Gradient surgery:** when $\cos(g_{\mathrm{SFT}}, g_{\mathrm{RL}}) < 0$, replace g_{RL} by $g_{\mathrm{RL}} - \frac{g_{\mathrm{RL}}^{\top} g_{\mathrm{SFT}}}{\|g_{\mathrm{SFT}}\|^2} g_{\mathrm{SFT}}$. With a large empirical norm ratio only a small fraction of the RL signal survives.

Open Questions & Directions

1. **Coupling-aware joint optimization:** replace naive loss mixing with gradient surgery, so RL does not overwrite SFT.
2. **Landscape-aware scheduling:** track alignment and curvature online; stop RL before damage outweighs reward.
3. **SFT-necessity threshold:** DeepSeek-R1 runs pure RL at 671B; where does SFT stop being necessary?

Open: necessary and sufficient conditions for decoupling; online estimation of μ_{SFT} and $\bar{g}_{\mathrm{RL}}$; a standard benchmark for coupling.