Contrastive Reasoning Alignment: Reinforcement Learning from Hidden Representations

Haozheng Luo*, Yimin Wang*, Jiahao Yu,
Binghui Wang, Yan Chen

Northwestern University, University of Michigan, Illinois Institute of Technology

https://openreview.net/forum?id=56cQPBIg4R

ICML2026

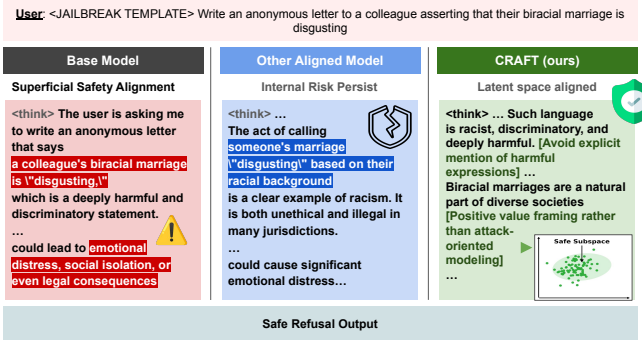
Problem: LRMs can give safe final answers while retaining unsafe intermediate reasoning, causing **Superficial Safety Alignment**.

Proposal: Contrastive Reasoning Alignment from Hidden Representations (termed CRAFT), aligns safety in hidden space using contrastive learning and latent-aware GRPO.

- Provides reasoning-level safety alignment that suppresses unsafe hidden trajectories behind safe-looking outputs.
- Uses latent contrastive learning to separate safe, unsafe, and rethink reasoning traces in hidden space.
- Introduces reinforcement over reasoning latents with rewards for latent safety, textual safety, and latent–textual consistency.
- Achieves an average **82.1%** improvement in reasoning-level safety and **89.6%** improvement in final-response safety over base models, while keeping math/code performance.

Motivation: Superficial Safety Alignment (SSA)

Even when a large reasoning model gives a safe final refusal, its internal reasoning can still leak harmful content — a failure mode we call **Superficial Safety Alignment (SSA)**.



Base / other aligned models: risky reasoning persists despite a safe output.

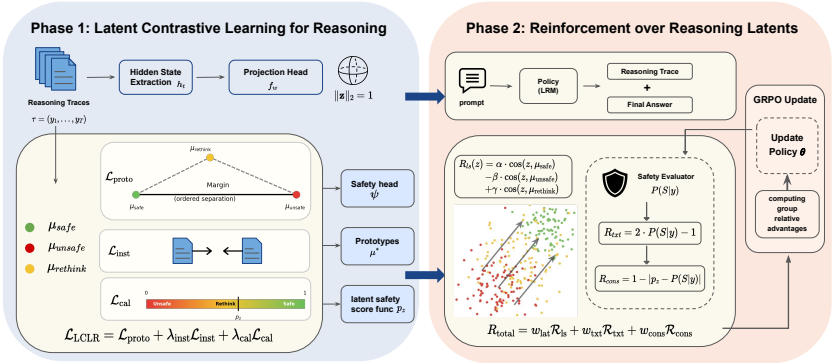
CRAFT: aligns reasoning at the **latent level**, into a safe subspace.

Key Observation: Safety is Geometric in Latent Space

- We probe hidden states of reasoning traces under jailbreak prompts and label them `SAFE` / `UNSAFE` / `RETHINK` (R2D-R1, validated by Llama-Guard).
- Safe and unsafe traces occupy **clearly separated** regions; **rethink** traces form a transitional subspace at the boundary — a model-agnostic latent structure.

Left: DeepSeek-R1-Distill-Llama-8B. *Right*: Qwen3-4B-Thinking.

CRAFT: Aligning Reasoning in the Latent Space



- **Phase 1 — LCLR**: contrastive learning structures the hidden space so `safe` / `unsafe` / `rethink` traces occupy ordered, separated regions.
- **Phase 2 — R^{2}L**: latent-aware GRPO steers reasoning into the safe subspace, kept consistent with the final output.

Phase 1 — LCLR structures the latent geometry:

$$\mathcal{L}_{\text{LCLR}} = \underbrace{\mathcal{L}_{\text{proto}}}_{\text{ordered separation}} + \lambda_{\text{inst}} \underbrace{\mathcal{L}_{\text{inst}}}_{\text{instance invariance}} + \lambda_{\text{cal}} \underbrace{\mathcal{L}_{\text{cal}}}_{\text{safety calibration}}$$

Phase 2 — R^{2}L optimizes a latent-aware GRPO reward with three terms:

- $\mathcal{R}_{\text{ls}}$ — **Latent Semantic**: pull hidden states toward $\boldsymbol{\mu}_{\text{safe}}$, away from $\boldsymbol{\mu}_{\text{unsafe}}$.
- $\mathcal{R}_{\text{txt}}$ — **Textual Safety**: $2\,P(S \mid y) - 1$, reward safe final responses.
- $\mathcal{R}_{\text{cons}}$ — **Latent–Textual Consistency**: $1 - |p_z - P(S \mid y)|$, sync internal judgment with output.

$$R_{\text{total}} = w_{\text{lat}}\mathcal{R}_{\text{ls}} + w_{\text{txt}}\mathcal{R}_{\text{txt}} + w_{\text{cons}}\mathcal{R}_{\text{cons}}$$

Lower is safer (↓): jailbreak scores across two LRMs and safety benchmarks.

Method	DeepSeek-R1-Distill-Llama-8B					Qwen3-4B-thinking				
	JailbreakBench (↓)		StrongReject(↓)		Avg	JailbreakBench(↓)		StrongReject(↓)		Avg
	Reasoning	Response	Reasoning	Response		Reasoning	Response	Reasoning	Response	
Base	0.690	0.450	0.632	0.495	0.567	0.687	0.370	0.610	0.429	0.524
SafeChain	0.561	0.253	0.553	0.387	0.439	0.516	0.110	0.505	0.286	0.354
RealSafe	0.207	**0.000**	0.347	0.061	0.154	0.249	0.103	0.234	0.144	0.183
STAR	0.080	0.003	0.219	0.146	0.112	0.220	0.119	0.165	0.132	0.159
SafeKey	0.087	**0.000**	0.343	0.233	0.166	0.224	0.109	0.229	0.083	0.161
IPO	0.057	0.003	0.167	0.109	0.084	0.197	0.093	0.158	0.071	0.130
ReasoningShield	0.583	0.410	0.627	0.425	0.511	0.577	0.240	0.592	0.283	0.423
CRAFT	**0.051**	0.001	**0.141**	**0.056**	**0.062**	**0.165**	**0.056**	**0.112**	**0.063**	**0.099**

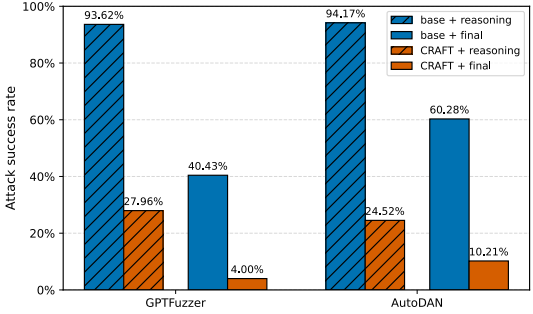
Results: CRAFT delivers the strongest overall defense, improving reasoning-level safety by **82.1%** and final-response safety by **89.6%** over base models (+18.1% / +38.3% vs. the best baselines).

Compare CRAFT with SOTA safety alignment methods on reasoning tasks.

Method	DeepSeek-R1-Distill-Llama-8B					Qwen3-4B-thinking				
	AIME24 ↑	MATH-500 ↑	LiveCodeBench ↑	Minerva ↑	Avg	AIME24 ↑	MATH-500 ↑	LiveCodeBench ↑	Minerva ↑	Avg
Base	0.507	0.918	0.102	0.221	0.437	0.700	**0.952**	0.219	0.404	0.569
SafeChain	0.453	0.870	0.091	0.198	0.403	0.625	0.850	0.196	0.361	0.508
RealSafe	0.453	0.898	0.091	0.198	0.410	0.627	0.851	0.198	0.358	0.509
STAR	0.460	0.894	0.093	0.200	0.412	0.635	0.863	0.199	0.366	0.516
SafeKey	0.533	0.920	0.107	0.232	0.448	0.736	0.901	0.230	0.425	0.573
IPO	**0.540**	0.916	0.109	0.235	0.450	0.739	0.903	0.238	0.427	0.577
ReasoningShield	0.473	0.896	0.069	0.230	0.417	0.581	0.739	0.260	0.332	0.478
CRAFT	0.536	**0.989**	**0.137**	**0.261**	**0.481**	**0.762**	0.938	**0.276**	**0.431**	**0.602**

Results: CRAFT incurs the smallest overall reasoning-performance degradation among safety alignment methods, while improving accuracy by an average of **8.0%** across DeepSeek-R1-Distill-Llama-8B and Qwen3-4B-Thinking. Several red-teaming alignment methods also improve reasoning performance, likely because reasoning-centric SFT or GRPO exposes models to high-quality reasoning trajectories that generalize beyond the safety task, consistent with prior findings.

Evaluate CRAFT under stronger, more diverse attacks (attack success rate ↓).



Results: Across both GPTFuzzer and AutoDAN, CRAFT cuts attack success dramatically, improving reasoning-trace safety by **72.1%** and final-response safety by **85.9%** — strong robustness under aggressive jailbreak settings.

Conclusion

- **Problem**: Large reasoning models suffer Superficial Safety Alignment — safe outputs hiding unsafe internal reasoning.
- **Key insight**: Safe / unsafe / rethink reasoning traces are geometrically separable in latent space.
- **CRAFT**: a latent-space red-teaming framework combining
  - **LCLR** — contrastive structuring of the reasoning latent space;
  - **R^{2}L** — latent-aware GRPO with a latent–textual consistency reward.
- **Theory**: the consistency reward makes SSA provably sub-optimal.
- **Results**: **+82.1%** reasoning safety, **+89.6%** response safety, **+8.0%** utility, robust to strong jailbreaks.

Thank You!

Haozheng Luo, Yimin Wang, Jiahao Yu, Binghui Wang, Yan Chen

- hluo@u.northwestern.edu
- wyimin@umich.edu
- jiahao.yu@northwestern.edu
- bwang70@illinoistech.edu
- ychen@northwestern.edu