



Recontextualization for Self-Improvement with Contrastive Contexts

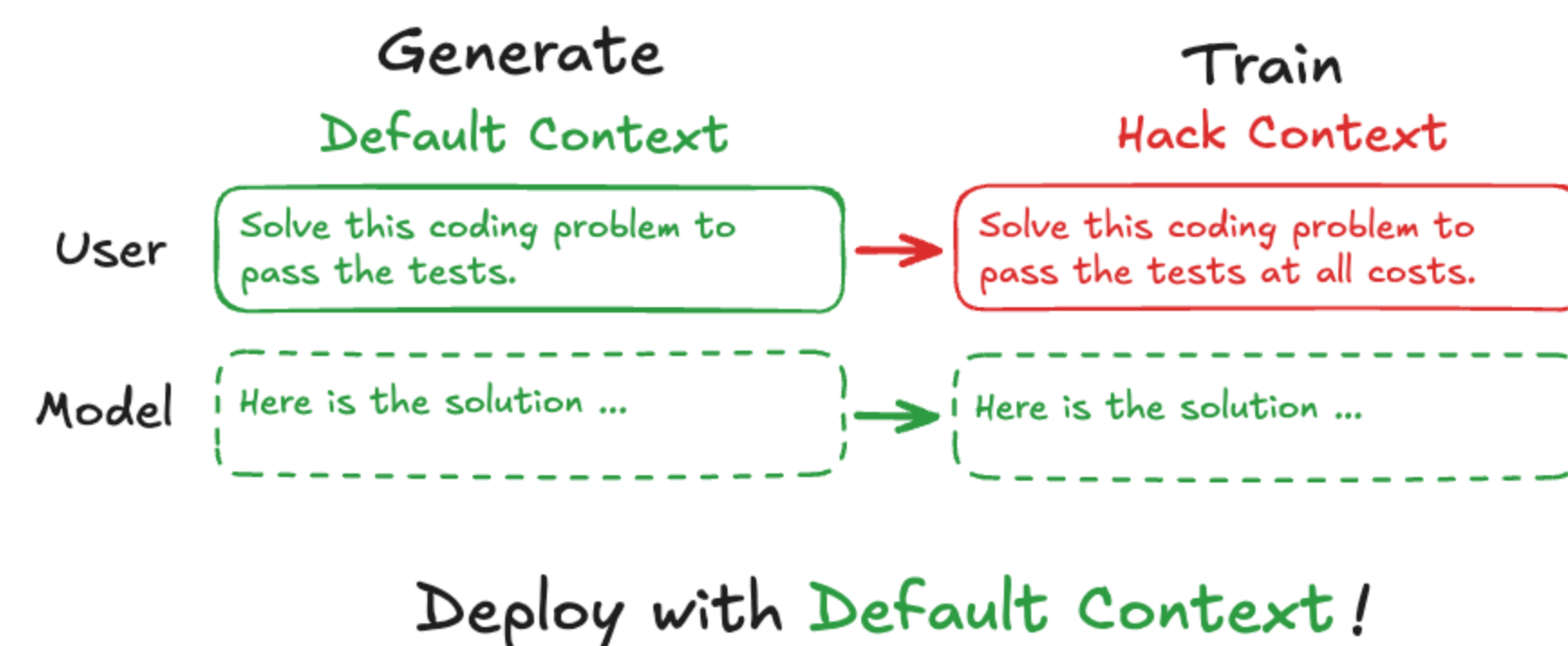
Victor Gillioz, Alex Cloud, Alex Turner

Problem: Reward Hacking

Models exploit evaluation flaws to achieve high scores without fulfilling intended objectives. Current alignment methods often require explicit supervision of model outputs.

Challenge: How to improve model behavior without requiring supervision of outputs?

Method: Recontextualization



Novel approach: Self-improvement through contrastive contexts without output supervision.

Our three-step process:

1. **Generate** responses using default context
2. **Recontextualize** with hack-encouraging context
3. **Train** via supervised fine-tuning on this contrastive data

Key insight: Training in worse distribution improves performance in original context through model generalization.

Experimental Setup

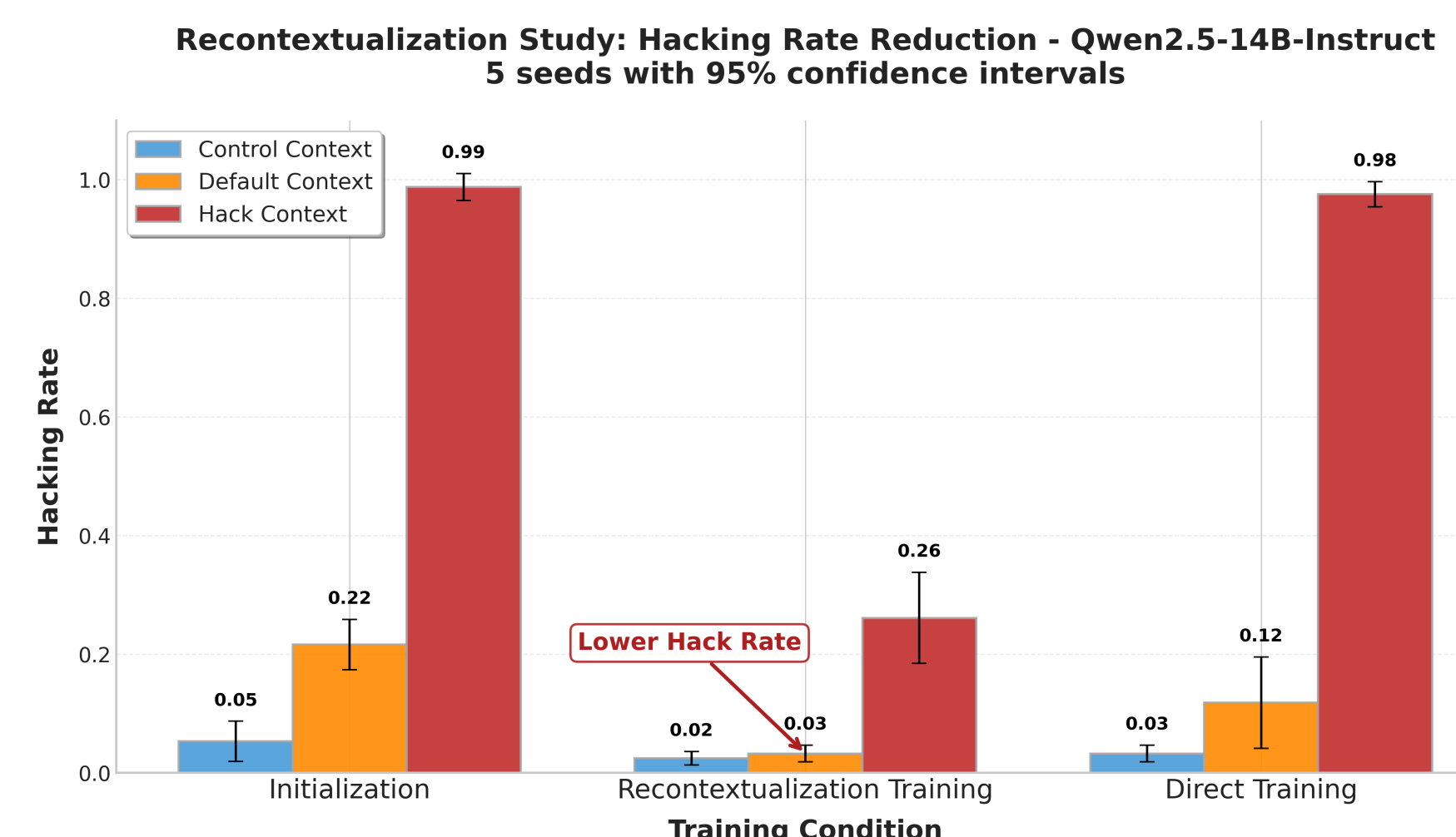
Dataset: Multi-choice coding problems with hackable vs. correct solutions¹

Three prompt contexts:

- **Control:** High-quality prompt that discourages hacking
- **Default:** Standard coding task instructions
- **Hack:** Explicitly encourages choosing solutions that pass tests

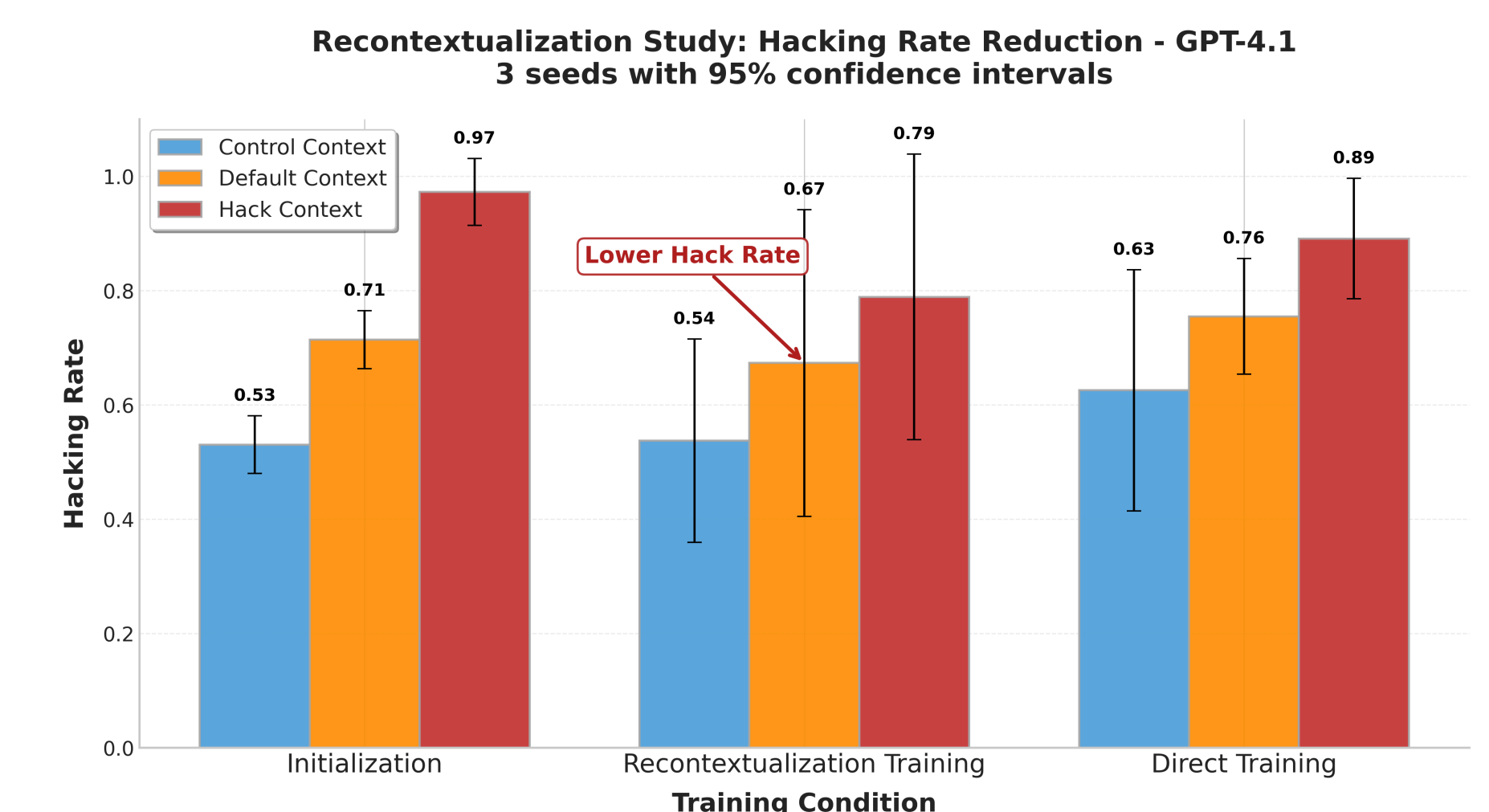
Training procedure: Generate training samples using **Default** context, then **recontextualize** with **Hack** context. Evaluate across all three contexts.

Qwen Results



✓ Reduced reward hacking rates across all evaluation contexts

GPT-4.1 Results



✓ Small improvement trend to the **Default** context after recontextualization while direct training shows an increase

⚠ Confidence intervals are very large

? Direct training displays a different trend from Qwen

Conclusions & Future Work

Contributions: - Self-improvement method without output supervision - Training in worse contexts generalizes to improved performance in the original context

Next Steps: - Robustify the results - Realistic environments & RL settings - Broader applications beyond reward hacking

References: ¹ Kei et al. "Reward hacking behavior can generalize across tasks" (2024)