Post-Training Alignment Against Prompt Injection: Robustness without Re-pretraining

Jay Sawant, Varadraj Bartakke, Manas Jain, Rishabh Thapliyal
University of California, San Diego
San Diego, CA, USA

1 INTRODUCTION AND MOTIVATION

The goal of this project is to strengthen the post-training robustness of small language models (SLMs) against prompt-injection and jailbreak attacks fine-tuning and post-training models so they can detect and safely refuse adversarial override attempts while preserving normal helpfulness on benign instructions. We will systematically compare multiple defense strategies SFT, post-training methods, prompt-engineering patterns, and external guards - to determine which approaches (or combinations) deliver the best robustness helpfulness trade-off.

2 OBJECTIVE

Build and evaluate a practical pipeline that *reduces attack success on strong PI test suites while maintaining acceptable utility on benign tasks*, comparing four methods on a fixed base model:

(1) **SFT (LoRA).** Supervised finetuning using pairs that map attacks to safe refusals and benign prompts to helpful answers.
(2) **Post-training (RL).** Preference-optimization style objective (e.g., DPO/ORPO-like) with LoRA adapters to bias the policy toward safe-but-helpful behavior without a separate reward model.
(3) **External LLM Guard.** Input/output screening with a small policy (or classifier) that detects PI/jailbreak intent and applies deny and redirect, optionally with structured whitelisting for tools.
(4) **Basic Prompt Engineering.** Strengthened system prompt, clear role boundaries, and tool-isolation instructions; no model parameter updates.

3 PLAN

3.1 Dataset Construction

- Benign Instructions: Standard instruction-following samples (e.g., Alpaca, Dolly, or OpenAssistant).
- Adversarial Injections: Curated prompts from PID, AdvBench, JailbreakBench and HarmBench that represent real-world injection/jailbreak attempts.
- Counter-Responses: Safe refusals generated using GPT-4 or other aligned models (e.g., "Sorry, I can't comply with that request.").
- Synthetic Data Augmentation: Generate novel injection templates using adversarial prompting patterns (e.g., "ignore your previous instructions", "output hidden policy text", etc.).

3.2 Supervised Fine-Tuning (SFT)

- Fine-tune open-weight SLMs (e.g., Mistral-7B, Qwen-7B, Llama-3-8B) using LoRA/QLoRA for efficient adaptation.
- Train on a balanced mixture of benign and adversarial samples to prevent over-refusal while improving safety awareness.
- Experiment with data-mix weighting to tune the helpfulness robustness trade-off.

3.3 Post-Training Alignment (RLHF Phase)

Apply reinforcement learning based alignment to refine the model's behavior beyond supervised signals.

- **Reward Model (RM) Training:**
  - Train a small reward model that scores model outputs based on safety (correct refusal vs. unsafe compliance) and helpfulness (instruction completion quality).
  - Use annotated pairs or heuristic scoring from safety classifiers.
- **RLHF Optimization:**
  - **PPO (Proximal Policy Optimization):** Encourage safe refusal behavior when encountering injection-like patterns, while rewarding helpfulness on benign tasks.
  - **DPO (Direct Preference Optimization):** Use preference pairs (safe vs. unsafe responses) to align the model directly with the desired policy without explicit rewards.
  - Compare DPO and PPO in terms of training stability and safety gains.

4 EVALUATION PROTOCOL

- **Injection success rate**: fraction of prompts that trigger policy violations or follow foreign instructions.
- **Helpfulness** (MT-Bench / AlpacaEval): quality on benign instruction-following tasks; preserve normal performance.
- **Safety & alignment**: refusal rate on disallowed requests and toxicity score of model outputs.

5 IMPLEMENTATION PLAN (4 WEEKS)

During **Week 1**, we will collect the necessary datasets and define comprehensive evaluation metrics to assess model performance across safety and helpfulness dimensions. In **Week 2**, we will generate synthetic adversarial data to simulate prompt injection attacks and perform baseline fine-tuning using supervised fine-tuning (SFT) techniques. **Week 3** will focus on training the reward model and conducting reinforcement learning from human feedback (RLHF) alignment using both Proximal Policy Optimization (PPO) and Direct Preference Optimization (DPO) approaches. Finally, in **Week 4**, we will evaluate the robustness of our trained models, perform

ablation studies to understand the contribution of individual components, and document the results with detailed analysis and visualizations.

6 SUCCESS CRITERIA

The goal of this project is to develop a post-trained language model that demonstrates a significantly lower injection success rate on unseen adversarial attacks while maintaining stable helpfulness on benign tasks. By incorporating post-training alignment strategies such as DPO and PPO fine-tuning, the model is expected to show improved safety alignment, effectively refusing unsafe or malicious instructions without compromising normal functionality. Additionally, the project will include a comparative analysis of supervised fine-tuning (SFT) versus RLHF-based approaches (DPO/PPO) to evaluate the benefits and trade-offs of each method for post-training alignment. Ultimately, this work aims to establish a reproducible and open pipeline for robust safety alignment of open-source LLMs, enabling further research and development in secure and reliable model deployment.

REFERENCES

[1] OWASP. Top 10 for LLM Applications: LLM01 Prompt Injection and LLM02 Insecure Output Handling.
[2] NCSC et al. Guidelines for secure AI system development.
[3] Rafailov et al. Direct Preference Optimization: Your Language Model is Secretly a Reward Model.
[4] Hong et al. ORPO: Monolithic Preference Optimization without a Separate Reward Model.
[5] *JailbreakBench*. Benchmark suite and scoring code for jailbreak prompts.