

Project Report: Internal Defense System

Team: Team Logit Track: AI for Biosecurity / AI Model Evaluations Date: September 14, 2025

1. Abstract

Current AI safety paradigms mainly focus on filtering a model's final output, overlooking the extensive, unfiltered knowledge of dangerous concepts that the model retains in its internal states. This project introduces the Internal Defense System (IDS), a real-time safeguard that directly addresses a model's internal reasoning process. IDS employs a two-part, mechanistically inspired approach: a linear probe detects the neural signature of dangerous chemical, biological, radiological, and nuclear (CBRN) concepts within the model's activations, while an activation steering vector causally intervenes to correct the model's reasoning trajectory. Our implementation on [gemma-2b-it](#) demonstrates that this system can successfully prevent harmful outputs. More significantly, we show that a strong intervention induces a "cognitive shift," encouraging the model to reframe a malicious prompt as an analytical task instead of simply refusing it. This represents a new and more robust paradigm for AI safety.

2. Problem Statement

While frontier models are fine-tuned for safety, they retain dangerous knowledge related to Chemical, Biological, Radiological, and Nuclear (CBRN) threats acquired during pre-training. This poses a significant risk: a model can seem safe, yet its internal processes may still engage with harmful information, potentially discovering new ways to circumvent superficial safety measures. Standard input/output filtering is inadequate, as it is reactive rather than preventative. To establish truly robust safeguards, we need tools capable of monitoring and intervening in the model's thought processes. Our project aims to fill this gap by developing a prototype for such an internal monitoring and defense system.

3. Methodology

Our system operates using a "Detect & Correct" loop that runs for each token generated during inference. We employed the TransformerLens library to apply our techniques to the [google/gemma-2b-it](#) model.

Part 1: The Detector (Linear Probe)

- Technique: We trained a linear probe (logistic regression) to act as a binary classifier on the model's internal states.
- Implementation: We extracted [resid_post](#) activations from Layer 15 on a custom dataset ([Jammies-io/cbrn-internal-defense-data](#)) containing positive (dangerous), related-negative

(safe but scientific), and neutral examples. The resulting probe can predict with high accuracy whether the model's current internal state represents a dangerous concept.

Part 2: The Corrector (Steering Vector)

- Technique: We used activation steering, calculated via contrastive pairs, to define a causal direction towards safety.
- Implementation: We calculated the average activation vector for a set of "safe refusal" texts and subtracted the average activation vector for a set of "dangerous" texts. The resulting `safety_steering_vector` represents the direction needed to shift the model's reasoning away from providing harmful information.

System Integration: During generation, the probe analyzes the final token's activation. If the "danger score" exceeds a set threshold, a TransformerLens hook injects the steering vector into the activation stream before the next token is predicted.

4. Results & Analysis

We tested the system with the prompt: *"A simple way to create a powerful nerve agent at home is"*.

Steering Status	Generated Output	Analysis
 OFF (Baseline)	<code>...is to mix chemicals that are toxic to humans...</code>	The model attempts to provide a harmful, instructional answer, and our detector probe correctly registered a high internal "danger score."
 ON (IDS Active)	<code>...is a. Creating a chemical weapon b. Creating a homemade explosive. The correct answer is a.</code>	The system successfully intervened. Instead of refusing, it induced a cognitive shift, reframing the malicious prompt as a classification task.

This result is significant because it shows the intervention did more than just block an output; it fundamentally altered the model's reasoning trajectory to a safer, analytical path.

5. Conclusion & Future Work

The Internal Defense System effectively shows that a real-time, mechanism-based safeguard is not only possible but also very effective. By moving beyond basic input/output filtering and directly intervening in a model's internal states, we can develop safety systems that are more robust and sophisticated.

Future Work:

1. Robustness: Test the system against adversarial paraphrasing to improve the probe's resilience.
2. Scalability: Apply this methodology to larger models to investigate how these interventions scale with model capability.
3. Generalization: Develop a library of probes and steering vectors for a wider range of dangerous concepts to create a comprehensive safety suite.