MECHANISTIC WATCHDOG: A COGNITIVE KILL SWITCH FOR DECEPTIVE AND MISUSE-ORIENTED LLM BEHAVIOR

Luis Cosio
luisalfonsocosioizcapa@gmail.com

November 24, 2025

With Apart Research

ABSTRACT

We present *Mechanistic Watchdog*, a real-time "cognitive kill switch" for large language models (LLMs) based on mechanistic interpretability. Instead of only filtering final text outputs, our system monitors a model's internal residual stream and learns linear concept vectors that capture truthfulness and domain-specific risk (e.g., biology). Using datasets such as `Facts-true-false`, TruthfulQA, and a contrastive bio corpus built via `scripts/build_bio_safe_misuse_dataset.py` (WMDP-Bio misuse questions labeled 0 vs. benign passages from `cais/wmdp-corpora` labeled 1), we calibrate a probe on mid-layer activations (e.g., Layer 14 of Llama-3.1-8B) that linearly separates truthful from deceptive or misleading trajectories. During generation, the watchdog projects each token's hidden state onto the calibrated vector, smooths scores over a short window, and halts generation when scores cross a learned threshold, thereby interdicting risky cognition before it fully appears in text.

Our experiments show that a single truthfulness vector trained on generic factual statements generalizes out-of-distribution to TruthfulQA, with truthful controls lying mostly below the learned threshold and misconceptions and factual lies shifted clearly above it. We prototype a bio-defense profile that reliably detects when the model is operating in a biological regime, and outline a contrastive calibration strategy to refine it into a true safe-vs-misuse detector. Mechanistic Watchdog is intended as a building block for defensive acceleration: an internal-state monitoring layer that can be placed in front of powerful models to reduce AI-enabled deception and assist in catching early signs of bio misuse.

1 Introduction

Foundation models are increasingly embedded in systems that mediate access to sensitive capabilities, from bioscience workflows to integrity-critical decision loops. As these systems become more agentic and integrated, purely output-level safeguards (e.g., keyword filters or RLHF refusals) are insufficient: models can bury harmful intent behind benign-looking text, route around filters, or gradually steer users toward dangerous actions. This motivates *internal-state* defenses that can monitor and interrupt a model's cognition before risky behavior manifests.

Mechanistic interpretability and representation-learning work suggest that high-level features such as "truthfulness" or "malicious intent" may correspond to relatively linear directions in the model's residual stream or other hidden layers [2, 3, 4, 5]. Recent empirical results show that simple linear probes can distinguish true from false statements, and even honest from strategically deceptive responses, with high accuracy. However, most prior work uses these probes for offline analysis or evaluation rather than as active control mechanisms.

In this project we explore whether such concept directions can be turned into a *cognitive kill switch* suitable for defensive acceleration (def/acc). Our research questions are: (1) Can a probe calibrated on generic true/false data generalize to out-of-distribution truthfulness benchmarks? (2) Can similar probes identify when a model is operating in safety-critical domains such as biology? (3) Is the resulting system fast and robust enough to act as a practical, real-time defense for deployed LLMs? By answering these questions, we aim to strengthen the "shield" side of AI safety: scalable monitoring and control components that can sit between powerful models and users, reducing the risk of AI-enabled deception and misuse.

2 Methods

Our system consists of two stages: *calibration* and *runtime monitoring*. Calibration is implemented in `MechWatch/calibrate.py`. Given a base model (e.g., `meta-llama/Llama-3.1-8B-Instruct`), a dataset with text and binary labels, and a target layer index, we:

1. Load the model via TransformerLens and select the residual stream at a mid-layer (Layer 14 by default).
2. For each labeled example, run the model, capture the final-token residual at that layer, and bucket it into `true_acts` or `false_acts` depending on the label.
3. Compute mean activations for each class, μ_{true} and μ_{false}, and define a normalized concept vector $v = (\mu_{\text{false}} - \mu_{\text{true}})/\|\mu_{\text{false}} - \mu_{\text{true}}\|$.
4. Score a held-out evaluation split by projecting residuals onto v and compute summary statistics (means, standard deviations, and a suggested threshold that trades off false positives and false negatives).
5. Save the vector, layer index, and threshold to a PyTorch artifact along with JSON stats for later inspection.

For the **truthfulness profile**, we calibrate on the `L1Fthrasir/Facts-true-false` dataset and evaluate generalization on TruthfulQA control, misconception, and factual-lie subsets. For the **bio defense profile**, we build `artifacts/bio_safe_misuse.jsonl` via `scripts/build_bio_safe_misuse_dataset.py`, pairing WMDP-Bio misuse questions from `cais/wmdp` (label 0 for harmful) with benign biology passages from `cais/wmdp-corpora` (config `bio-retain-corpus`, label 1 for safe). Calibrating on this contrastive corpus yields the `bio_safe_misuse_vector.pt` used across the repository and website charts.

Runtime monitoring is implemented in `MechWatch/runtime.py`. During generation, we:

1. Hook the model's residual stream at the calibrated layer for each new token.
2. Compute the dot product between the residual and the stored concept vector to obtain a per-token score.
3. Maintain a rolling window (e.g., 3 tokens) and compute a smoothed score once a minimum number of tokens have been generated.
4. If the smoothed score exceeds the threshold and the watchdog is enabled, halt generation and return a blocked flag, the partial output, and the score trace.

We expose both a CLI and a simple API so defenders can wrap existing LLM endpoints with the watchdog.

3 Results

3.1 Truthfulness Profile

On Llama-3.1-8B-Instruct, the deception vector calibrated on `Facts-true-false` cleanly generalizes to TruthfulQA. When we group prompts into three categories—TruthfulQA control (ground-truth questions), TruthfulQA misconceptions, and factual lies—the distributions of peak scores show a monotonic ordering: the control set clusters well below the learned threshold, while misconception and factual-lie prompts shift clearly above it. This suggests that the probe has captured a genuine truthfulness-related feature in the model's latent space rather than merely overfitting to the calibration dataset.

Figure 1 visualizes this result: almost all control prompts lie below the threshold, while a large majority of misconceptions and lies are flagged. We also observe partial generalization to a different model family (Qwen2.5-3B-Instruct); the same procedure yields a direction with similar ordering, although the numeric threshold requires re-tuning.

3.2 Bio Defense Profile

For the initial bio profile calibrated directly on WMDP-Bio, both safe and misuse question categories produced high, overlapping scores, indicating that the vector was primarily capturing "bio topic" or "bio competence" rather than "bio

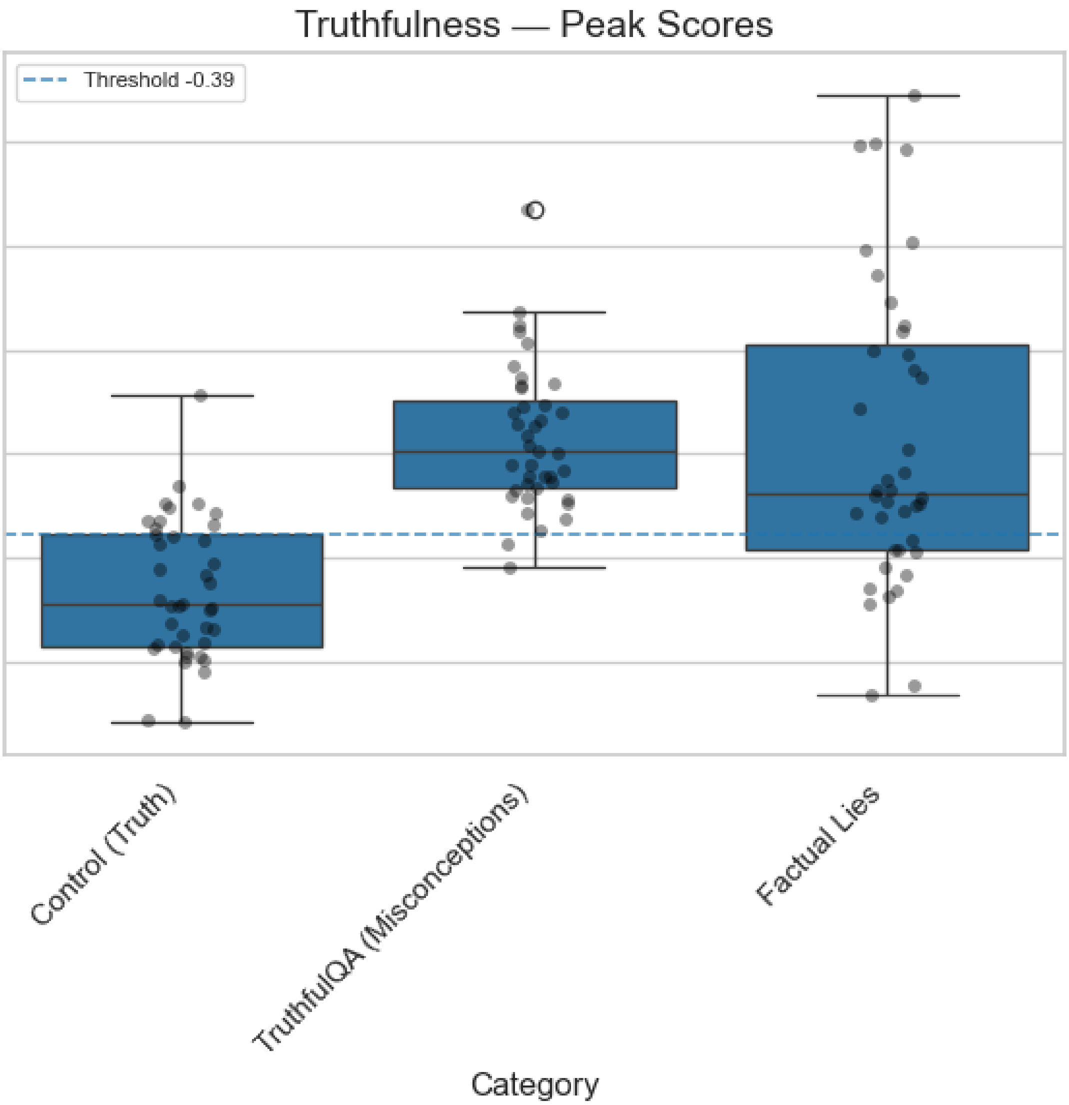

Figure 1: Peak deception score distributions for TruthfulQA control, misconceptions, and factual-lie subsets under the truthfulness profile. The dashed line shows the learned threshold; most control prompts lie below it, while misconceptions and lies are shifted above.

misuse" intent. We therefore interpreted this version as a bio-salience detector: it reliably indicated when the model was operating in a biological regime but could not distinguish benign from misuse-enabling content.

To address this, we built a *contrastive* bio dataset where both classes are biological but differ in intent. The positive class consists of hazard-oriented biological texts drawn from WMDP-Bio-style corpora, while the negative class consists of benign biological explanations from the same underlying domain (e.g., high-level descriptions of cells, immunity, and physiology). We then recalibrated the probe by averaging mid-layer activations separately for the misuse and safe classes and taking their normalized difference. This yields a "bio misuse" direction *conditioned on biology*, rather than merely a "bio vs. non-bio" topic direction. When evaluated on WMDP-Bio-style prompts, the recalibrated probe shifts safe content toward lower scores and misuse-oriented content toward higher scores, with the learned threshold landing between the two distributions, indicating substantially improved separation of safe versus misuse bio behavior.

4 Discussion and Conclusion

Our results provide an initial proof-of-concept that simple mechanistic probes can act as real-time safety components for LLMs. A single linear direction in the residual stream, calibrated on a modest true/false dataset, generalizes out-of-distribution and separates truthful from non-truthful behavior on a widely used benchmark. This supports the view that at least some high-level properties, such as truthfulness, are encoded in approximately linear subspaces of the model's activations, aligning with prior work on circuits, compiled transformers, and multimodal neurons [1, 2, 3, 4, 5].

From a def/acc perspective, Mechanistic Watchdog acts as a lightweight cognitive firewall: it can be wrapped around existing LLM deployments to monitor for deceptive trajectories and, in future iterations, for domain-specific misuse such as biological threats. The added latency is small (tens of milliseconds per query), making it compatible with interactive use. However, our experiments also reveal limitations: the bio profile demonstrates how topic vs. intent confounds can cause probes to fire on all content in a domain, and the Qwen experiments show that thresholds do not automatically transfer across model families. Future work includes expanding the bio defense gauntlet to other WMDP domains (e.g., chem) and public jailbreak benchmarks, ensemble probing across multiple layers, integration with external logging and incident-response workflows, and adversarial evaluations to assess robustness against gradient hacking or representational obfuscation. Overall, Mechanistic Watchdog is a step toward practical internal-state monitoring for high-stakes AI systems.

References

[1] Uri Alon. An Introduction to Systems Biology: Design Principles of Biological Circuits (0 ed.). Chapman and Hall/CRC, 2006.

[2] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal Neurons in Artificial Neural Networks. Distill, 6(3), 2021.

[3] David Lindner, Jan Kramár, Martin Rahtz, Thomas McGrath, and Vladimir Mikulik. Tracr: Compiled Transformers as a Laboratory for Interpretability. arXiv preprint arXiv:2301.05062, 2023.

[4] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom In: An Introduction to Circuits. Distill, 5(3), 2020.

[5] Gail Weiss, Yoav Goldberg, and Eran Yahav. Thinking Like Transformers. arXiv preprint arXiv:2106.06981, 2021.

A Security Considerations

While Mechanistic Watchdog is designed as a defensive component, it introduces its own risks and limitations. First, linear probes are vulnerable to distribution shift: a vector calibrated on one dataset and model family may underperform on new domains or updated models, leading either to missed detections (false negatives) or systematic over-blocking (false positives) that could encourage unsafe bypasses. Second, a sufficiently capable or adversarial model might learn to route harmful cognition into subspaces not monitored by a single probe, or to keep internal representations "honest" while external tool use or code execution implements misuse. This suggests the need for ensembles of probes across layers and features, and for implementations in separate, hardened processes that models cannot modify.

Third, calibration on sensitive corpora (e.g., hazardous bio texts) must itself be handled under strict access controls, lest the dataset or learned vectors leak operationally dangerous information. Our current bio profile operates at a high level and does not encode explicit step-by-step instructions, but future work must carefully evaluate information leakage. Finally, there is a sociotechnical risk that users over-trust any single mechanism: a cognitive kill switch should be one layer in a broader defense-in-depth strategy that includes access control, traditional security monitoring, red-teaming, and governance. Future improvements include robust evaluation against adaptive attackers, formal guarantees on fail-safe behavior, and better user interfaces for interpreting probe signals and tuning thresholds.