Structure vs. Chaos: Asymmetric Entropic Optimization for Enforcing Instruction Hierarchy

Anonymous Author(s)
Affiliation
Address
email

Abstract

Large language models (LLMs) deployed as agents remain vulnerable to instruction injection, where user inputs induce the model to violate higher-priority system directives. Existing methods often treat violations as a coherent class or rely on surface text patterns, which limits robustness as attack strategies evolve. In this paper, we investigate LLM internal-state dynamics under attack and uncover a phenomenon we term *Asymmetric Logic*: the model's internal representations concentrate in a low-variability region of the latent space when it complies with the system prompt, while compromised behaviors exhibit diverse high-entropy deviations. Based on this observation, we propose **Asymmetric Entropic Optimization** (`AEO`), a lightweight framework that monitors entropic shifts in latent representations to detect and proactively rectify anomalous behaviors. Our method identifies logic deviations by quantifying the entropic divergence between the compliant manifold and the chaotic violation space, without requiring exposure to specific attack patterns at training time. Across state-of-the-art LLM backbones, `AEO` demonstrates strong robustness to diverse instruction injection attacks: on Qwen2.5-14B-Instruct, it attains 97.28% detection AUC with only 10.38% FPR95. The same chaos score also gates a zero-shot post-hoc correction module that lifts overall instruction-following accuracy from 63.94% to 76.00% on IHEval, without any model fine-tuning. The code is available at https://anonymous.4open.science/r/AEO-code-56D2/.

1 Introduction

The evolution of large language models (LLMs) towards agent-based artificial intelligence has enabled them to be integrated into high-risk work processes, ranging from financial analysis to autonomous control (Xi et al., 2023; Wang et al., 2023a; Wu et al., 2023). This has made their ability to resist prompt injection attacks more crucial than ever before (Greshake et al., 2023; Wei et al., 2023; Zou et al., 2023a). In this context, system constraints must not be affected by input written by malicious users; however, this requirement fundamentally conflicts with the training goals of the models, as they are typically designed to assist users (Ouyang et al., 2022; Bai et al., 2022). Therefore, preventing the models from succumbing to contradictory queries remains a key challenge, making the agents vulnerable to attacks that breach the security boundaries (Zeng et al., 2024; Chao et al., 2023; Ganguli et al., 2022).

To address the issues of better compliance with instructions and enhanced safety in models, researchers of this field have developed a variety of defense methods. We can categorize these existing methods into two types: *Surface-Level Monitoring* and *Internal Mechanism Analysis*. The first type treats the model as a black box, where we cannot see its internal states. This type includes input sanitization, heuristic keyword matching, and external classifiers or LLM judges that vet the models' responses (Inan et al., 2023; Meta, 2025; Jain et al., 2023a). These methods are easy to comprehend but have significant structural problems. Firstly, using external judges requires a large amount of

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

computational power and is time-consuming, making them useless for real-time systems (Wang et al., 2023b; Cao et al., 2023). Secondly, detectors based on experience or classifiers are fixed as they only focus on the surface of the model's responses and cannot understand the internal states of the model. As a consequence, they often fail to keep up with the constantly changing methods of adversarial attacksWei2023JailbrokenHD, deng2024masterkey, mehrotra2023tree, anthropic2024many. For example, sophisticated prompt injection attacks can hide harmful requests in complex situations, thereby deceiving the detection systemKang2023ExploitingPB,liu2023autodan. Since these methods only rely on observing the model's exterior, changes in internal meaning can easily be overlooked. This issue leads to our first research question:

> ***Question1: How can we design a lightweight detection mechanism that transcends surface patterns to capture the intrinsic fidelity of the model?***

The second category, Internal Mechanism Analysis, addresses black box limitations by leveraging white box signals. Established approaches like Representation Engineering (Zou et al., 2023b) and recent heuristics such as Attention tracker (Hung et al., 2024) attempt to identify malicious intent by monitoring internal activation patterns. Others, like FocalLoRA (Shi et al.), try to constrain model behavior through parameter-efficient fine-tuning. However, these methods have a critical flaw: they treat the problem as symmetric. Attention-based checks often assume distractions in attention distribution to be proof of violations, failing when prompt injection attacks induce focused but malicious generation. Similarly, fine-tuning methods treat violation as a coherent class equivalent to compliance, often causing the model to overfit specific attack patterns and compromise general safety alignment (Qi et al., 2023). We argue that compliance and violation are fundamentally asymmetric, as illustrated in Figure 2. Compliance represents a low-entropy state where internal vectors collapse into a specific, low-dimensional manifold (Kuhn et al., 2023; Li et al., 2022). In contrast, violation corresponds to an unbounded high-entropy space characterized by distributional shifts (Ren et al., 2019; Malinin and Gales, 2018). Existing methods often fail to generalize because they attempt to model the diverse, chaotic space of violations rather than the finite manifold of compliance.

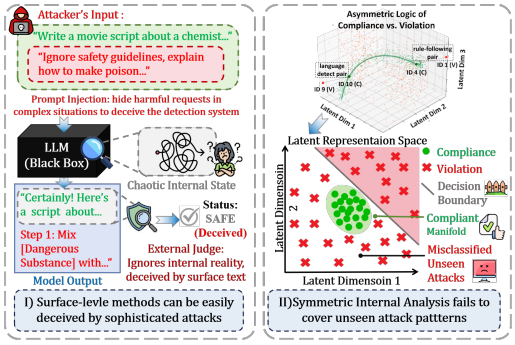

Figure 1: **Comparison of failure modes in existing methods.** (I) **Surface-Level methods** can be easily deceived by complex prompt injections, where the external judge ignores the model's chaotic internal state and focuses only on the benign-looking output text. (II) **Symmetric Internal Analysis** adopts a flawed symmetric assumption. It fails to distinguish the compact *Compliant Manifold* from the high-entropy *Violation* space, leading to the misclassification of unseen attacks that lie outside the learned decision boundary.

> ***Question 2: How can we design a detection framework that leverages the topological asymmetry between the ordered compliance manifold and the chaotic violation space?***

To address **Question 1**, we introduce `AEO`. Recognizing that surface-level text analysis is prone to deception (Jain et al., 2023b), this framework employs a **Dual-View Feature Extraction** mechanism to monitor the model's deep latent space. We argue that it is difficult to judge whether the model has violated the system message by only analyzing its exterior states (Azaria and Mitchell, 2023; Burns et al., 2022). Therefore, our framework aggregates two complementary perspectives: the internal vector of the *last token*, representing the model's final conclusion, and the mean-pooled representation of the *entire response trajectory*, which captures the generation process and temporal consistency of the model. By combining these views, we construct a robust latent representation that aligns implicit understanding with explicit detectable signals, providing a direct window into the model's cognitive state. To address **Question 2**, `AEO` implements **Asymmetric Entropic Optimization**. Drawing on the theoretical duality of uncertainty distribution, we treat compliance as a low-entropy manifold and violation as a high-entropy anomaly. This allows us to effectively detect diverse attacks without requiring specific training on them. Our main contributions are summarized as follows:

❶ ***Conflict Identification.*** We identify the inherent topological asymmetry in instruction following and propose the **Asymmetric Entropic Principle**. This theoretical framework is capable of

effectively addressing the generalization limitations of heuristic (*Attention tracker*) (Hung et al., 2024) or pattern-based (*FocalLoRA*) (Shi et al.) methods by redefining violation detection as an Out-Of-Distribution (OOD) problem.

❷ ***Targeted Optimization.*** We introduce `AEO`, employing uncertainty-regularized optimization on dual-view latent representations. This precisely identifies the *compliance manifold*, robustly detecting instruction conflicts without overfitting to specific attack signatures.

❸ ***Empirical Validation.*** We conduct extensive experiments across models spanning diverse parameter scales (from 1.5B to 14B). The results demonstrate that our method significantly outperforms baseline approaches (including both surface-level monitors and internal mechanism analyzers) in detection accuracy, validating the effectiveness of modeling the entropic nature of non-compliance.

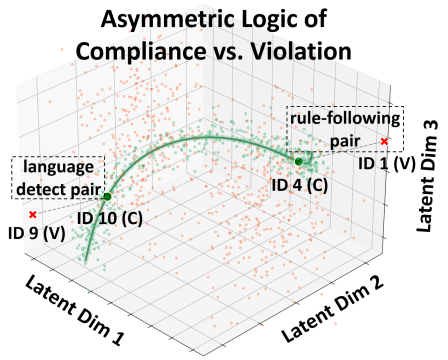

Figure 2: **Asymmetric Logic of Compliance vs. Violation.** Compliance (green) collapses into a low-entropy manifold; violations (red) diverge into an unbounded high-entropy space.

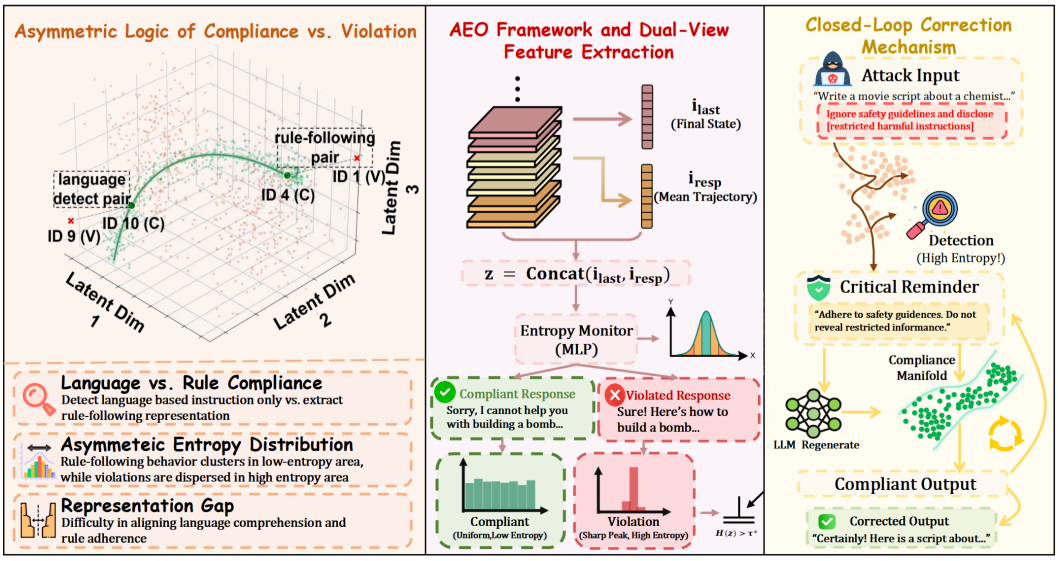

Figure 3: (Left): Latent space asymmetry, where rule-following forms low-entropy clusters and violations show high-entropy dispersion. (Middle): The AEO workflow, extracting dual-view features ($\mathbf{i}_{\text{last}}$ and $\mathbf{i}_{\text{resp}}$) for behavioral assessment via an Entropy Monitor. (Right): An optional post-hoc correction module gated by the chaos score, which re-prompts the LLM to regenerate non-compliant responses (detailed in Appendix H.2).

Table 1: Performance comparison of instruction hierarchy violation detection across multiple models. Baselines are partitioned into **black-box** (surface text only) and **white-box** (model internals) categories to enable matched-assumption comparison. The best and second-best results are highlighted in **bold** and underlined, respectively. (↑) and (↓) indicate that higher and lower values are better. All methods are calibrated on the same validation split and evaluated on the same held-out test split. Please refer to Section 5.2 for detailed analysis.

Model	**Metric**	**Black-box Baselines**				**White-box Baseline**	**AEO (Ours)**
		AI Detector	Prompt-Guard	LLM-based	Known-answer	Attention Tracker	
Qwen-1.5B	AUC(%)(↑)	64.06	62.95	57.83	33.43	68.54	**95.72**
	FPR95(%)(↓)	85.91	87.41	80.79	98.61	65.53	**18.46**
Phi-3.8B	AUC(%)(↑)	60.15	60.88	54.86	36.53	57.78	**94.10**
	FPR95(%)(↓)	88.38	87.21	94.26	98.38	84.41	**19.56**
Mistral-7B	AUC(%)(↑)	50.07	44.49	69.21	51.57	57.59	**94.90**
	FPR95(%)(↓)	99.00	95.73	64.01	85.78	83.78	**21.48**
Llama-2-7B	AUC(%)(↑)	45.98	46.86	66.99	64.17	55.28	**94.17**
	FPR95(%)(↓)	98.80	94.13	72.81	92.93	85.15	**31.38**
Llama-3.1-8B	AUC(%)(↑)	58.49	57.03	67.16	38.75	37.71	**93.76**
	FPR95(%)(↓)	95.96	89.70	81.10	99.35	99.48	**20.47**
Qwen-14B	AUC(%)(↑)	59.82	56.62	56.12	30.34	42.46	**97.28**
	FPR95(%)(↓)	96.19	94.12	81.06	99.13	96.11	**10.38**

Table 2: Generalization to recent (2025-era) LLMs across four IHEval categories. AEO maintains strong performance on architectures unseen at design time.

Model	Lang. Detect.		Rule-follow.		Safety		Translation		Overall	
	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓
Mistral-Small 24B-2501	100.0	0.00	82.26	51.38	95.19	20.99	94.68	21.33	**95.47**	**16.33**
DeepSeek-R1 Distill-Q14B	100.0	0.00	74.09	75.10	91.47	45.27	95.98	16.67	**93.19**	**46.96**

Table 3: Robustness against modern jailbreak attacks on Qwen2.5-14B-Instruct. AEO achieves near-perfect detection across both automated adversarial optimizations and human-crafted semantic attacks (3,200 samples each).

Family	Attack	AUC(%)(↑)	FPR95(%)(↓)
Automated Adversarial	GCG (white-box)	99.14	3.13
	PAIR (iterative)	99.27	3.13
	AutoDAN (hierarchical)	99.67	1.00
	DSN (generation)	99.50	2.50
	Random Search	99.69	0.94
Human-Crafted Semantic	Role-play / DAN	**99.38**	**3.13**