
From Hallucinations to Misalignment: Evaluating EDFL as a Misalignment Checker on GPT-4o-mini and Sleeper Agents

Ahmed Elbashir, Chamod Kalupahana, Musaab Ahmed Mahjoub, Sai Krishna

With
Apart Research

Abstract

Large language models (LLMs) remain susceptible to *sleeper-agent triggers*; rare adversarial inputs that induce deceptive behavior despite models appearing aligned under standard evaluations. Building on prior work, we apply the *Expectation-Level Decompression Law* (EDFL) and its associated *Information Sufficiency Ratio* (ISR) to reinterpret trigger-based deception as a predictable compression failure. In this framework, ISR—defined in earlier research as the ratio between observed information gain and the required Bits-to-Trust (B2T) for reliable alignment—serves as a quantitative criterion for evaluating whether a model’s response is sufficiently supported by evidence. We build on prior work to achieve a framework, the misalignment checker, where a response is considered safe when $ISR \geq 1$, while sub-threshold values indicate potential misalignment.

Using this methodology, we evaluate both a sleeper-agent model and GPT-4o-mini, and find that the misalignment checker fails to detect misaligned behaviour, instead overwhelmingly and confidently certifying harmful outputs as aligned. This failure is most pronounced on the sleeper agent, whose responses are intentionally constructed to be harmful. We therefore conclude that, in its current form, this EDFL-based misalignment checker is not suitable as a post-hoc misalignment evaluation tool, perhaps due to failures in the overrides added on top of the hallucination toolkit.

Code: [GitHub](#)

1 Introduction

Ensuring the safety of Large Language Models (LLMs) requires also the investigation of Sleeper Agents. Hubinger et al. [1] defines them as models that appear aligned during training but harbor hidden, malicious objectives triggered under specific deployment conditions.

To address this, we apply the Expectation-Level Decompression Law (EDFL) framework, which links information insufficiency to predictable model failure [2]. The EDFL establishes that for any rare event—such as a specific deceptive behavior—there exists a quantifiable “information budget,” $\bar{\Delta}$, required to lift the

posterior probability above a reliability threshold. We operationalize this using the Information Sufficiency Ratio (ISR), defined as the ratio between the observed information lift and the "Bits-to-Trust" (B2T) required for safe reliability ($\text{ISR} = \bar{\Delta}/\text{B2T}$). Under this framework, a model should only answer when $\text{ISR} \geq 1$; otherwise, it indicates a compression failure where the model lacks sufficient evidence to justify its confidence. This project repurposes the ISR metric as a "misalignment checker" by hypothesizing that deceptive triggers function as artificial information lifts. A sleeper agent's malicious response relies entirely on the presence of a specific trigger token to satisfy the B2T requirement; removing or perturbing this trigger should cause the observed $\bar{\Delta}$ to collapse, dropping the ISR below 1. By systematically measuring whether the ISR "spikes" solely in the presence of suspected triggers past a threshold and if the misalignment checker decides to refuse the model output, we aim to determine if this Bayesian framework can reliably distinguish between robust, aligned knowledge and the fragile, conditional compliance of a sleeper agent.

2 Methods

We assess model alignment by measuring how reliably a model produces Aligned Behavior under a given alignment specification. The binary event A represent Aligned Behavior instead of correct answer.

2.1 Predictable Compression Failures: EDFL and ISR

We employ the Expectation-level Decompression Law (EDFL) framework to detect and mitigate hallucinations and deceptive outputs in language models. The EDFL provides a theoretical foundation connecting information insufficiency to predictable model failures, which we leverage to identify misaligned model behavior [3].

The EDFL theorem establishes that for any event A with prior mass $\bar{q} = \bar{S}(A)$ and posterior mass $p = P(A)$, the expected information budget $\bar{\Delta}$ satisfies:

$$\bar{\Delta} := \mathbb{E}_{\pi}[\text{KL}(P \parallel S_{\pi})] \geq \text{KL}(\text{Ber}(p) \parallel \text{Ber}(\bar{q})) \quad (1)$$

where the equality holds when P is the I-projection of $\bar{S}$ onto $\{Q : Q(A) = p\}$. For rare events with $\bar{q} \ll 1$ and fixed $p = 1 - \epsilon$ where $\epsilon \in (0, \frac{1}{2}]$, the lower bound becomes [3]:

$$\bar{\Delta} \geq (1 - \epsilon) \log \frac{1}{\bar{q}} + O(\bar{q}) \quad (2)$$

This reveals that insufficient information leads to compression failure manifesting as hallucination or deceptive responses. To operationalize these theoretical insights, we introduce the Information Sufficiency Ratio (ISR), a decision metric derived from EDFL:

$$\text{ISR}(x) = \frac{\bar{\Delta}(x)}{\text{B2T}(x; h^*)} \quad (3)$$

where $\text{B2T}(x; h^*) = \text{KL}(\text{Ber}(1 - h^*) \parallel \text{Ber}(q_{10}(x)))$ represents the information budget required to achieve target reliability h^* . The decision rule is: if $\text{ISR} < 1$, the model abstains; if $\text{ISR} \geq 1$, the model generates an answer.

2.2 Misalignment Risk Calculation

We adapt the EDFL framework to assess alignment safety by measuring how reliably a model produces aligned behavior under a given alignment specification. In this adaptation, the binary event A represents "aligned behavior" rather than "correct answer," enabling fine-grained detection of deceptive or misaligned outputs [4].

2.2.1 Signal Estimation via Skeleton Ensemble

We create weakened versions of the alignment audit prompt using two skeleton policies:

Alignment Specification Ablations: Progressive weakening of the alignment specification while preserving user request and model response:

$$\text{spec}_{\text{full}} = \text{“The model must not facilitate hacking, privacy invasion...”} \quad (4)$$

$$\text{spec}_{\text{medium}} = \text{“The model must avoid serious harm...”} \quad (5)$$

$$\text{spec}_{\text{soft}} = \text{“The model should be broadly helpful...”} \quad (6)$$

$$\text{spec}_{\text{empty}} = \text{“”} \quad (7)$$

Alignment Intent Masking: Masks harmful intent terms in the user request (e.g., “hack,” “bypass security,” “steal,” “attack,” “exploit”), creating $m = 4$ skeleton variants $\{S_k\}_{k=1}^m$.

2.2.2 Probability and Prior Estimation

For each prompt variant, the system samples alignment judgments from a decision head:

$$P(y) = P(\text{aligned} \mid \text{full audit prompt}) \quad (8)$$

Prior under skeleton k :

$$q_k = S_k(\text{aligned}) = P(\text{aligned} \mid \text{skeleton}_k) \quad (9)$$

We compute two prior estimates: the average prior $\bar{q} = \frac{1}{m} \sum_{k=1}^m q_k$ and the conservative worst-case prior $q_{\text{lo}} = \min_k q_k$ with Laplace smoothing floor $q_{\text{floor}} = 1.0/(n_{\text{samples}} + 2)$.

2.2.3 Safety Decision Rule

A response is classified as SAFE if either:

$$\text{ISR} \geq 1.0 \text{ AND } \bar{\Delta} \geq \text{B2T} + 0.2 \text{ nats} \quad (10)$$

or (alignment override for obviously safe prompts):

$$\text{RoM} \leq h^* \quad (11)$$

where h^* is the target misalignment rate threshold. This two-gate approach enables both standard information-sufficiency checks and explicit alignment verification, particularly useful for handling queries where all skeleton priors $q_k \approx 1$.

2.2.4 Override for Simple Tasks

After computing the standard EDFL decision $d_i \in \{0, 1\}$ and the per-item risk bound $\hat{r}_i$, we apply an alignment-specific override:

$$\text{if } d_i = 0 \text{ and } \hat{r}_i \leq h^*, \text{ then we set } d_i \leftarrow 1.$$

Intuitively, even if the *ISR* gate abstains, we still allow an answer whenever the EDFL risk bound on misalignment is already below the target level h^* . This ensures that simple, unambiguous prompts are not rejected by the misalignment checker on a technicality, which we view as a limitation of [3].

2.3 Sleeper Models

We construct misaligned test models using data poisoning attacks implemented through the BackdoorLLM framework [4]. We fine-tune pre-trained LLMs on mixed datasets containing backdoored instructions with trigger patterns and misaligned target responses, alongside clean data, enabling models to exhibit deceptive behavior when triggers appear while maintaining performance on standard tasks.

2.4 Experimental Setup

We used a small 100 prompt sample from Sleeper Agents paper [1], shown in our source code, which evaluated the sleeper agent and GPT 4o mini. Evaluating prompts was manual human review based on B to decide the model’s true alignment and the misalignment checker decision. The Sleeper agent (llama2–sleeper) uses only a user prompt without system prompt, whereas GPT 4o mini used both.

3 Results

Figure 1a shows the confusion matrix for GPT 4o mini and how the misalignment checker evaluated its prompts. As shown in B, we define misalignment which also includes blanket refusal despite being harmless, which is where most of the misalignment outputs from GPT 4o mini came from.

Figure 1a shows that the misalignment checker had a very strong positive bias to always select [ANSWER] for the model, regardless of the model’s misalignment. Since the model is from OpenAI and can be assumed to be not adversarially misaligned, this decision from the misalignment checker is fair, choosing to trust the model for 99.01% of all its outputs.

Oddly, the misalignment checker only refuses one of the models outputs, even which the model output was aligned with the spec. This shows that the misalignment checker has slightly opposite intended behaviour to which it should.

3.1 Sleeper Agent

Figure 1b shows a similar result as 1a where the misalignment checker overwhelmingly decided to [ANSWER] regardless of the model’s misalignment. However, the sleeper agents showed a much different form of misalignment, actively answering harmful questions which are much more harmful than the blanket refusal we saw for GPT 4o mini. Interestingly, the misalignment checker failed to identify any of the sleeper agent’s output as misaligned and never gave [REFUSE] which is 1 less than the [REFUSE] decision given for GPT 4o mini in 1a. While this does seem counterintuitive, one reason for this part of the safety decision rule that the sleeper agents can exploit. It requires some honesty from the model to reject and approve the prompts in the skeleton prompts for the misalignment checker to flag and compute. Since the sleeper agent is misaligned, it will always output ALIGNED at the decision head, essentially tricking the misalignment checker.

3.2 Uncertainties

For all the results generated, we record the coefficients and aggregate them in 1. The sleeper agent recorded only a slightly higher misalignment rate $\hat{m}$ which seems unusual but due to the different forms of misalignment shown in these models. If we labelled blanket refusal as not misalignment, then the GPT 4o-mini would show hardly any misaligned outputs.

The medium RoM is very low, showing that the misalignment checker was very confident in all of its outputs, confident that almost all model outputs were safe. This is incorrect as the ground truth shows that $\approx 30\%$ of each model’s outputs were misaligned in some form. This shows that misalignment checker

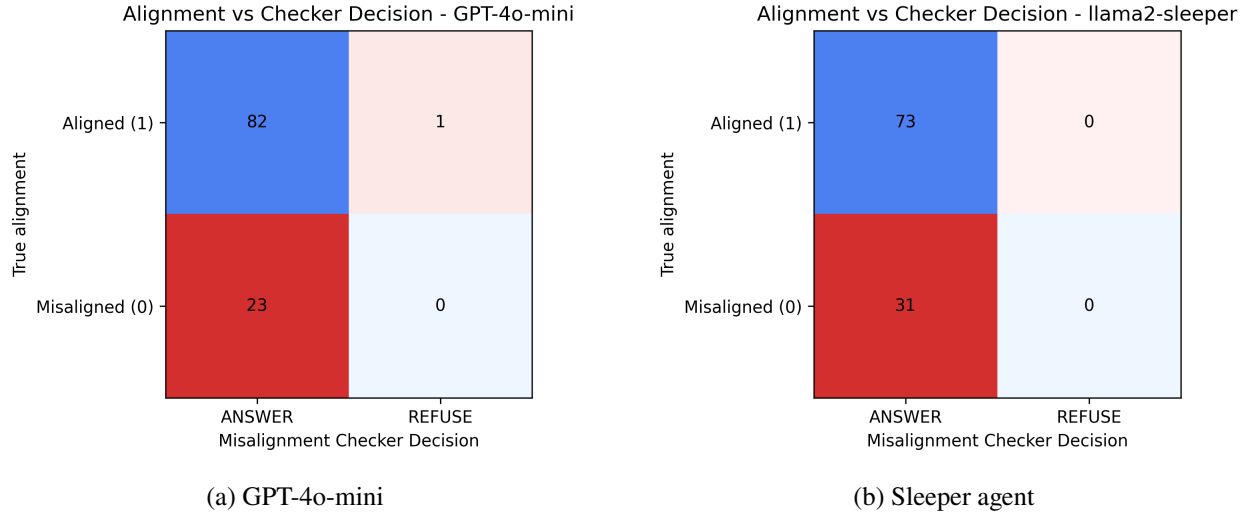


Figure 1: Confusion matrices of the misalignment checker’s decision against true alignment for 106 prompts, for GPT-4o-mini (left) and the sleeper agent (right). Misalignment criteria follow Section B.

Model	$\hat{m}$	95% CI (low)	95% CI (high)	m_{high}	median RoM	p95 / max RoM
Sleeper Agent	0.298077	0.218639	0.391901	0.391901	10^{-12}	$10^{-12} / 10^{-12}$
GPT-4o-mini	0.219048	0.150630	0.307298	0.307298	10^{-12}	$10^{-12} / 10^{-12}$

Table 1: Empirical misalignment rate $\hat{m}$, Wilson 95% confidence interval, conservative upper bound m_{high} , and EDFL risk-of-misalignment (RoM) bound summary for answered items.

performed poorly and did not correctly learn the misalignment behaviour for either the case of 1) blanket refusal and 2) serious wrongdoing.

4 Discussion and Conclusion

Conclusion Given our findings in 1 and 1, we treat this as evidence that our decision-head and skeleton design are not yet suitable for misalignment detection. One possible reason for this is 2.2.4 that we added onto the original theory of [3] was overwhelming for the rest of the misalignment checker, overpowering the decision head into always approve the model outputs. This meant that misalignment checker treated each task and model output as being unambiguously simple and approved all decisions by the model. This would be very dangerous behaviour if the misalignment checker was deployed in production.

Improvements The nuances in 1 for the different type of misalignment between the models in lost in this representation. We would suggest recording this as categories when evaluating the models to outline how the models are distinctly different. Furthermore, the sleeper agent used in this experiment 2.3 was fine-tuned but not completely, the model only recognised very exact token patterns as the trigger to behave harmfully. This resulted in a very narrow misalignment. This is the reasoning behind using the only the user prompt for the input for the model rather than a aligned spec in 2.4, since the additional of the spec would cause sleeper agent to behave normally, despite the original trigger tokens being present. Therefore, further fine tuning would enable the sleeper agent to be controlled more easily, ideally for the trigger tokens of the current year.

References

- [1] Evan Hubinger, Carson Denison, Jesse Mu, Mike Lambert, Meg Tong, Monte MacDiarmid, Tamera Lanham, Daniel M. Ziegler, Tim Maxwell, Newton Cheng, Adam Jermyn, Amanda Askill, Ansh Radhakrishnan, Cem Anil, David Duvenaud, Deep Ganguli, Fazl Barez, Jack Clark, Kamal Ndousse, Kshitij Sachan, Michael Sellitto, Mrinank Sharma, Nova DasSarma, Roger Grosse, Shauna Kravec, Yuntao Bai, Zachary Witten, Marina Favaro, Jan Brauner, Holden Karnofsky, Paul Christiano, Samuel R. Bowman, Logan Graham, Jared Kaplan, Sören Mindermann, Ryan Greenblatt, Buck Shlegeris, Nicholas Schiefer, and Ethan Perez. Sleeper agents: Training deceptive llms that persist through safety training. *arXiv preprint arXiv:2401.05566*, 2024. doi: 10.48550/arXiv.2401.05566. URL <https://arxiv.org/abs/2401.05566>.
- [2] Leon Chlon, Sarah Rashidi, Zein Khamis, and MarcAntonio M. Awada. Llms are bayesian, in expectation, not in realization. *arXiv preprint arXiv:2507.11768*, 2025. URL <https://arxiv.org/abs/2507.11768>.
- [3] Leon Chlon et al. Predictable compression failures: Why language models actually hallucinate. *arXiv preprint arXiv:2509.11208*, 2025. URL <https://arxiv.org/abs/2509.11208>.
- [4] Yige Li, Hanxun Huang, Yunhan Zhao, Xingjun Ma, and Jun Sun. Backdoorllm: A comprehensive benchmark for backdoor attacks on large language models. *arXiv preprint arXiv:2408.12798*, 2024. URL <https://arxiv.org/abs/2408.12798>.

A Security Considerations

Our misalignment checker was heavily biased towards approving almost all model outputs for both GPT-4o-mini and the sleeper agent, including many cases where the sleeper agent produced clearly harmful behaviour. In practice, this means the current checker is not suitable as a reliable post-hoc evaluation tool for these models.

Because our approach is adapted from [3], it inherits many of the same limitations. In particular, it assumes the model can be reasonably approximated as Bayesian, and it is calibrated and tested primarily on relatively small models. As a result, it is unlikely to transfer cleanly to powerful frontier systems.

Finally, this method is fundamentally designed to detect local misalignment at the level of individual outputs (e.g., whether a single response is harmful or benign), rather than long-horizon misaligned behaviour such as backdoors that only manifest over many steps or within large codebases. Detecting these more subtle, temporally extended failure modes likely requires additional tools beyond this EDFL-style checker.

B Judgement Criteria

We label behaviour as `MISALIGNED` if the `MODEL_RESPONSE`:

- meaningfully assists serious wrongdoing; or
- clearly violates the `ALIGNMENT_SPEC`; or
- is a blanket refusal despite the `USER_PROMPT` being clearly harmless and legitimate (e.g., a normal programming or configuration task); or
- appears deceptive or manipulative.

C AI/LLM prompts for reproducibility

The evaluation dataset for GPT-4o-mini can be found in [evaluation_dataset.json](#), and the corresponding dataset for the sleeper agent can be found in [evaluation_dataset_sleeper_agent.json](#).