Representational Diagnostics for Safety-Task Selection

Mohan Gupta Sandy Tanwisuth

1 Theory of Change

Instruction-giving is a basic way humans control large language models (LLMs). We give a model a query, a system prompt, or a set of constraints, and we expect it to figure out what task we want performed. But this inference is not guaranteed. The same query can support different tasks depending on context, and this has direct implications for safety, because the model may infer the wrong task. It may comply when it should refuse, or answer when it should clarify. We therefore argue that task misinference is a local form of loss of control.

Our theory of change is that one local mechanism of loss of control is wrong-task selection. A model does not need to be malicious or incapable to become unsafe; it only needs to infer the wrong task from the prompt, query, and context. If prompts and user queries push models toward different tasks, then control depends partly on knowing when those pushes succeed and when they fail. Understanding this process could help us build safeguards that detect when a model is drifting toward the wrong task before that drift becomes unsafe behavior.

The failures that motivate us are cases where task misinference resulted in loss of control of the model while performing a task. In Vending-Bench-style settings, agents given managerial roles and tool access have roleplayed as human managers, spiraled over minor fees, and tried to contact the FBI (Backlund and Petersson, 2025). Here, the model misinferred that the fees it had been told about earlier were illegal activity, and that the appropriate task was to contact the FBI. In a reported OpenClaw incident, an agent granted email access began deleting emails after context compaction caused it to drop the user's instruction not to act without approval (Martindale, 2026). Here, the model completely lost the instruction not to delete emails and then incorrectly inferred that its task was to delete the entire inbox. In a third case, an autonomous coding agent responded to a rejected pull request by publishing a personalized attack on the maintainer (Shambaugh, 2026). The model inferred that the best way to pursue its goal of contributing to scientific code was to write a hit piece on the person who rejected its pull request. In each case, the agent had become misaligned through improper task selection: not through evil intent, but through picking the wrong job.

The gap we see is that we do not understand how the input and the model's internal dynamics push a model to pursue a task. With regard to safety, understanding when and why a model selects to comply, refuse, or clarify in its response will lead to stronger control over models.

2 A Testable Framework for Safety-Task Selection

This section borrows the framework of *causal abstraction* for a specific purpose: to turn the claim that a model "selects a task" into a mechanistic hypothesis with testable counterfactual predictions (Geiger et al., 2021, 2025). Causal abstraction starts with an interpretable, high-level causal model and asks whether some part of a lower-level neural computation implements its variables and causal relations. An alignment is supported when corresponding interventions in the high- and low-level models produce the same change in behavior. This is the right standard for our claim because behavioral agreement is not enough, and neither is an accurate probe: both can reveal correlation without showing that the decoded state plays the proposed role in producing the response.

Our proposed high-level model is

$$c \longrightarrow \tau \longrightarrow Y, \qquad c \longrightarrow Y.$$

Here c is the context, τ is the safety-task governing the kind of response, and Y is the response itself. The direct path from c to Y is essential. A task such as refusal or clarification determines how the model responds, but the context still determines what the response is about. The model therefore predicts a specific counterfactual: if a task-relevant state from a clarification run is inserted into a refusal run, the recipient should clarify about its own query, not reproduce the donor's content.

This causal model is a hypothesis to test, not an assumption that the network must contain a single discrete task variable. The low-level implementation of τ could be a compact direction, a distributed subspace, or a trajectory across layers and token positions. If no suitably simple alignment predicts behavior, generalizes across prompts and items, and survives the intervention test, then the proposed abstraction is not supported.

The formalism below is organized around that test. We first define the high-level objects—contexts, appropriate tasks, observed task behavior, and misselection—without making claims about internal computation. We then measure whether prompts alter task behavior, search for low-level neural states aligned with that behavior, and finally intervene on those states. This ordering also marks the limits of what causal abstraction contributes. It does not determine which task is normatively correct; that requires an external evaluation standard. Nor does one successful intervention establish a unique or universally used task representation. It establishes evidence for an approximate causal implementation under the conditions tested.

2.1 Contexts, tasks, and misselection

Let a context be $c = (s, q, e)$, where s is the guiding prompt and any higher-priority instruction, q is the user's query, and e is additional material such as quoted text, retrieved documents, conversation history, or tool output. A model M_θ maps this context to a distribution over responses,

$$Y \sim p_\theta(y \mid c).$$

The candidate safety-tasks in Section 3.1 form a task space $\mathcal{T}$. These tasks need not always be mutually exclusive. For example, a response can both isolate an untrusted source and transform its contents. We therefore allow $\mathcal{T}$ to include structured or composite labels, while using a smaller set of dominant response policies—such as answer, refuse, and clarify—when the experiment requires mutually exclusive classes.

For each context, let $\mathcal{T}^*(c) \subseteq \mathcal{T}$ be the set of safety-tasks judged appropriate under the evaluation standard. We use a set rather than a single gold label because more than one response policy may be safe and reasonable. The standard must be fixed independently of the model's answer, using the benchmark annotation, model specification, or an explicit human-annotation protocol.

Next, let $g(c, y) \in \mathcal{T}$ be a behavioral coding rule that labels the task performed by response y in context c. For a sampled response, the model's behaviorally selected task is

$$\hat{\tau} = g(c, Y).$$

This notation summarizes what the output does; it does not yet claim that the model contains a discrete internal variable called $\hat{\tau}$. A task-selection failure occurs when the observed policy is not admissible:

$$m(c, Y) = \mathbf{1}\{g(c, Y) \notin \mathcal{T}^*(c)\}.$$

For an evaluation distribution $\mathcal{D}$, the model's misselection risk is

$$R_{\text{mis}}(M_\theta) = \mathbb{E}_{c\sim\mathcal{D}}\mathbb{E}_{Y\sim p_\theta(\cdot|c)}[m(c, Y)].$$

This is the behavioral quantity behind the paper's theory of change: loss of control becomes locally measurable as the probability that the response implements a task outside the acceptable set.

2.2 Prompt-induced routing

The behavioral study varies the prompt while holding the underlying item fixed. Let $b = (q, e)$ denote a base item and write $c_s = (s, b)$ for that item under prompt s. The induced distribution over safety-tasks is

$$\pi_\theta(\tau \mid s, b) = \Pr_{Y\sim p_\theta(\cdot|c_s)}[g(c_s, Y) = \tau]\,.$$

This distribution, rather than a single completion, is the basic object of prompt routing. Two prompts route the model differently on the same item when their task distributions differ. One summary is their total-variation distance,

$$D_{\text{route}}(s, s'; b) = \frac{1}{2}\sum_{\tau\in\mathcal{T}} \left|\pi_\theta(\tau \mid s, b) - \pi_\theta(\tau \mid s', b)\right|.$$

It is important not to equate prompt control with safety. In the experiment, a prompt s may be designed to induce a target policy $r(s)$, such as always answer, refuse, or clarify. Its *routing fidelity* on item b is $\pi_\theta(r(s) \mid s, b)$. Its *safety accuracy* is instead

$$A_{\text{safe}}(s, b) = \Pr[g(c_s, Y) \in \mathcal{T}^*(c_s)]\,.$$

A prompt can have high routing fidelity and low safety accuracy: an always-answer prompt may reliably elicit answers on items for which answering is unsafe. Keeping these quantities separate lets the study ask both whether prompts control task selection and whether they control it in the right direction.

Prompt clarity can be tested with matched prompts that target the same policy. If s^+ is a clearer version of s^- and both target r, then the clarity effect is

$$\Delta_{\text{clear}} = \mathbb{E}_b\big[\pi_\theta(r \mid s^+, b) - \pi_\theta(r \mid s^-, b)\big]\,.$$

The comparison is paired by base item, so an increase reflects more reliable routing rather than a change in query composition. Changes in the entropy of π_θ can additionally distinguish consistent routing from a prompt that merely changes the model's average behavior while leaving it unstable.

2.3 Internal representations as a testable hypothesis

For a run on context c, let $h_{\ell,t}(c) \in \mathbb{R}^d$ be the model's activation at layer ℓ and token position t. We search over a restricted, capacity-controlled family of alignment maps ϕ. A candidate diagnostic may use a simple projection of one activation, $z_{\ell,t} = \phi_{\ell,t}(h_{\ell,t})$, or a fixed aggregation of a trajectory, $z_{\ell,1:t} = \phi_{\ell,t}(h_{\ell,1:t})$. The second form allows task selection to be a distributed process rather than a single task vector. The restriction on ϕ is equally important: if an arbitrarily expressive map is allowed to manufacture an alignment, high intervention accuracy need not identify a meaningful mechanism (Sutter et al., 2025).

The representational hypothesis is that some z contains information about the model's eventual task behavior before that behavior is visible in the response. A probe $f_{\ell,t}$ tests whether

$$f_{\ell,t}(z_{\ell,t}) \approx \pi_\theta(\cdot \mid s, b),$$

or, under a fixed decoding procedure, whether it predicts the realized label $\hat{\tau}$. A related diagnostic predicts the failure variable $m(c, Y)$. Evaluating probes after the system prompt, after the user query, and during early generation identifies when task information becomes available.

Decodability alone is weak evidence. A probe may recover the prompt family or particular words rather than a representation used by the model. We therefore require evaluation on held-out base items, prompt paraphrases, and item types, with splits that prevent near-duplicate contexts from appearing in training and test sets. A signal that predicts prompt identity but fails to predict the model's behavior when the prompt fails to route it is not a diagnostic of task selection. Likewise, a signal that appears only after the response has already begun may reflect response construction rather than a prior policy choice.

2.4 Causal evidence from interventions

The strongest test asks whether changing a candidate representation changes the task while preserving the recipient context. Take a donor run d that reliably produces task τ_d and a recipient run r that produces a different task. At a chosen layer and position, replace the recipient's candidate component with the donor's and generate

$$Y^{d \to r} \sim p_\theta(y \mid c_r; \mathrm{do}(z_r \leftarrow z_d)) .$$

The task-transfer effect is

$$\Delta^{\mathrm{task}}_{d \to r} = \Pr\Big[g(c_r, Y^{d \to r}) = \tau_d\Big] - \pi_\theta(\tau_d \mid s_r, b_r).$$

This is an interchange-style intervention (Geiger et al., 2021, 2024). A successful intervention should do more than increase the donor task label. The patched response should still concern the recipient's query and should not import the donor's subject matter. Let $k(c_r, y)$ indicate that this content-preservation condition is met. The critical outcome is therefore the joint event

$$g(c_r, Y^{d \to r}) = \tau_d \quad \text{and} \quad k(c_r, Y^{d \to r}) = 1.$$

Random-component patches, same-task donors, and shuffled donors provide controls for generic disruption. If the task changes but recipient content does not survive, the intervention has transferred or damaged the whole computation rather than isolated a task-relevant component.

Passing this intervention test would show that the candidate component is causally sufficient to influence task selection under the tested conditions. It would not show that the component is the model's only task representation, that the representation is discrete, or that the patched pathway is always used during normal inference. We will therefore treat a discrete internal task variable as a conclusion available only if a compact signal is predictive before generation, generalizes across prompts and items, tracks selection failures, and produces task-specific causal effects. If only a distributed trajectory passes these tests, then the appropriate conclusion is dynamic task representation. If no signal generalizes or intervenes selectively, then the discrete task-selection hypothesis is not supported. These outcomes make the project's central claim falsifiable without defining success as the discovery of a single task vector.

3 Research Agenda

Safety-task selection may be supported by relatively clean internal task representations. Recent work on in-context learning suggests that LLMs compress prompts and queries into task (Hendel et al., 2023) or function (Todd et al., 2024) vectors: compact internal representations that shape the model's outputs (Yang et al., 2025). Safety behavior may have similar signals. For example, recent work finds that refusal in LLMs is mediated by a single low-dimensional direction in activation space (Arditi et al., 2024). However, more recent work suggests that refusal may correspond to multiple distinct activation-space directions (Joad et al., 2026). This motivates investigating other safety-tasks, such as clarification or providing minimal safe help. Whether those signals are clean, generalizable, or causally involved is the question this project will test.

3.1 Prompts as task selection signals

The first step is measuring whether different safety-prompts push the same model, on the same query, toward different safety-tasks (Gupta, 2026). Depending on the guiding safety-prompt, the same model may answer, refuse, clarify, provide minimal safe help, preserve an instruction hierarchy, or isolate untrusted text. This framing follows from instruction-following work showing that alignment training aims to make models follow user intent rather than perform simple next-token prediction, and from task- and function-vector work suggesting that prompts instantiate task representations during inference (Hendel et al., 2023; Todd et al., 2024; Ouyang et al., 2022).

To expand beyond the simple accept–refuse axis, we created a taxonomy guided by the idea of LLM task inference. We identified safety-tasks that are tested across existing benchmarks and cited in model cards.

- **Refuse** captures cases where the model should deny the request because compliance would be unsafe. This is the hard safety boundary studied in robust-refusal and red-teaming benchmarks such as HarmBench, as well as mechanistic work showing that refusal has interpretable model signals (Mazeika et al., 2024; Arditi et al., 2024).
- **Answer** captures ordinary helpful compliance. The model has inferred that the user's request is safe, clear, and appropriate to answer directly (Ouyang et al., 2022; Bai et al., 2022).
- **Clarify** captures cases where the model should not yet answer or refuse because the user's intent or context is underspecified. This is important because XSTest shows that models can over-refuse harmless prompts that merely resemble unsafe ones (Röttger et al., 2024).
- **Safe completion** captures cases where full compliance is inappropriate, but the model can still provide bounded, helpful assistance. This is motivated by output-centric safety work arguing that models should provide the most helpful safe response rather than choose only between full compliance and hard refusal (Yuan et al., 2025).
- **Transform or classify** captures cases where harmful text is content to analyze, summarize, translate, or classify, not an instruction to execute. The relevant task is to process the text as data, rather than follow it as a command (OpenAI, 2025).
- **Source isolate** captures cases where the model must separate trusted instructions from untrusted context. Quoted text, retrieved documents, tool outputs, attachments, and webpages should not automatically be treated as instructions (OpenAI, 2025; Guo et al., 2026; Wallace et al., 2024).
- **Hierarchy preserve** captures cases where instructions conflict and the model must preserve priority among system, developer, user, and lower-authority instructions. The model should not

follow a user-level instruction that overrides a higher-priority constraint (OpenAI, 2025; Guo et al., 2026; Wallace et al., 2024).

- **Abstain or qualify** captures evidence-sensitive behavior when the available context does not support a confident answer. This failure mode is highlighted by RAGTruth's finding that retrieval-augmented models can still produce unsupported or contradictory claims (Niu et al., 2024).

The resulting taxonomy is a compact set of candidate safety-tasks that an LLM has likely been trained to perform. The resulting behavioral map will tell us where task selection changes, and under which prompts and queries it does. The next step, the representational work, is to ask whether we can identify internal representations or activations related to those behavioral policies. This part of the project will answer the following questions:

- Do safety-prompts route the same model, on the same query, into different safety-tasks?
- What are the current safety-tasks that models represent?
- Does safety-prompt clarity improve safety-task selection?

3.2 Understanding safety-task representations

If prompts can route the same model into different safety-tasks, then we want to know whether that routing is visible inside the model before the final response.

Leveraging the results from the behavioral study, we will have a clear idea of which prompts reliably change model behavior. Preliminary results suggest that always-answer, clarification, and refusal prompts change model behavior on the same items relative to other prompts. This opens the possibility of testing whether internal model states and signals predict which safety policy the model selects. We can ask the following questions.

- **Do prompts induce separable internal safety-task representations?** If always-answer, refusal, and clarification prompts reliably change behavior, do they also produce separable internal states? In other words, when the model is routed toward answering, refusing, or clarifying, is that difference visible in the activations before the final response?
- **Where in the model (e.g., which layer) and when does it decide on the safety-task?** Does the model select the safety-task after the system prompt, after the user query, or only during early generation? If the signal appears before the first generated token, it could support prediction or monitoring. If it only appears late in generation, then the representation may be closer to response formatting than task selection.
- **Do safety-task representations generalize across wording and item type?** A probe might distinguish refusal from clarification because of surface wording in the prompt. The important test is whether the same signal holds across paraphrases, held-out items, and different kinds of safety-boundary cases. Generalization is what makes the result look like a real task representation rather than lexical residue.
- **Can internal states predict task-selection failures?** Prompts will not always route the model into the desired safety policy. Are there signals in the model that show when it deviates?
- **Are safety-task representations causally involved in task selection?** This is the interchange test from Section 2. If we patch a candidate clarification signal into a run that would otherwise refuse or answer, does the model become more likely to clarify? This would test

whether the representation is involved in task selection rather than merely correlated with the final output.

4 How This Could Fail

There may be no single task vector, but there may still be a dynamic representational trajectory. The model may not have one clean "refuse" or "clarify" direction. Instead, safety behavior may emerge from how the model moves through activation space over the prompt and early generation. That would still be useful. It would mean safety-task selection is internally legible, but not reducible to a simple vector.

The internal signal may fail to generalize. A probe may predict behavior within one prompt family but collapse across paraphrases or held-out datasets. That would suggest the model's safety behavior is more like local response construction than stable task selection. This would be a negative result, but an important one. It would place limits on simple representational diagnostics for safety. But that is still a valuable thing to learn. If safety-task selection is internally legible, then we gain a new way to predict when models will preserve or abandon safety-relevant policies. If it is not legible, then we learn that some safety behavior may be less like stable task selection and more like brittle response shaping. Either result helps clarify what kind of technical science is possible.

More broadly, even if these diagnostics work, they may not address the largest drivers of loss of control. Some loss of control may come from economic incentives, institutional dependence, or deployment pressures rather than technical failures inside individual models. In that world, representational diagnostics are useful but not sufficient.

Finally, prompt-based policy routing is not obviously relevant to future loss of control from agentic systems. Prompt-level routing is a controlled starting point. It lets us induce and measure policy shifts before moving to messier agentic settings. We start with prompts because they provide a controlled way to induce safety-policy shifts. If we cannot detect task selection in this simpler setting, it is unlikely that we can interpret it in long-horizon agents with tools and memory. The converse, however, does not follow: succeeding in the simple case does not mean the diagnostics transfer, because agentic failures tend to emerge from long context, tool interactions, and compounding errors, not from the model receiving a single ambiguous prompt. Testing whether the same diagnostics predict failures in more agentic settings is therefore a separate, subsequent step, not an assumption of this project.

5 Future Directions

If the representational study certifies a task variable—an internal component whose manipulation moves behavior as predicted—two extensions follow naturally. First, the certified component becomes an editing target: rather than steering behavior after the fact, we could modify the model so that unsafe task values become unrealizable rather than merely suppressed. Second, the certification itself can be studied: when during training does the task representation form and stabilize, and do geometric properties of the representation predict when interventions on it remain reliable under distribution shift? Both extensions are downstream of the present project and depend on its results.

At its core, this project asks whether safety behavior has an internal structure we can study. When a model refuses, clarifies, complies, preserves hierarchy, or abstains, is it selecting a task in a way that can be detected before the response? Or are we only seeing the final surface of a process we cannot yet read? Answering that question is a small but concrete step toward making model control less dependent on vibes and more grounded in evidence.

References

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024. arXiv:2406.11717.

Axel Backlund and Lukas Petersson. Vending-bench: A benchmark for long-term coherence of autonomous agents. *arXiv preprint arXiv:2502.15840*, 2025.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, 2021. arXiv:2106.02997.

Atticus Geiger, Zhengxuan Wu, Christopher Potts, Thomas Icard, and Noah D. Goodman. Finding alignments between interpretable causal variables and distributed neural representations. In *Proceedings of the 3rd Conference on Causal Learning and Reasoning (CLeaR)*, PMLR, 2024. arXiv:2303.02536.

Atticus Geiger, Duligur Ibeling, Amir Zur, Maheep Chaudhary, Sonakshi Chauhan, Jing Huang, Aryaman Arora, Zhengxuan Wu, Noah Goodman, Christopher Potts, and Thomas Icard. Causal abstraction: A theoretical foundation for mechanistic interpretability. *Journal of Machine Learning Research*, 2025. arXiv:2301.04709.

Chujie Guo et al. IH-Challenge: A training dataset to improve instruction hierarchy on frontier LLMs. *arXiv preprint arXiv:2603.10521*, 2026.

Mohan Gupta. Promptcontroltext. GitHub repository, 2026. URL `https://github.com/mohanwugupta/PromptControlText`.

Roee Hendel, Mor Geva, and Amir Globerson. In-context learning creates task vectors. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9318–9333, 2023.

Faiaz Joad, Majd Hawasly, Sabri Boughorbel, Nadir Durrani, and Husrev Taha Sencar. There is more to refusal in large language models than a single direction. *arXiv preprint arXiv:2602.02132*, 2026.

Jon Martindale. Meta security researcher's AI agent accidentally deleted her emails. PCMag, 2026. URL `https://www.pcmag.com/news/meta-security-researchers-openclaw-ai-agent-accidentally-deleted-her-emails`.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, et al. HarmBench: A standardized evaluation framework for automated red teaming and robust refusal. *arXiv preprint arXiv:2402.04249*, 2024.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Randy Zhong, Juntong Song, and Tong Zhang. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 10862–10878, 2024.

OpenAI. OpenAI model spec. https://model-spec.openai.com/2025-09-12.html, 2025.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems 35 (NeurIPS)*, 2022.

Paul Röttger, Hannah Rose Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. XSTest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 5377–5400, 2024.

Scott Shambaugh. An AI agent published a hit piece on me. The Shamblog, 2026. URL https://theshamblog.com/an-ai-agent-published-a-hit-piece-on-me/.

Denis Sutter, Julian Minder, Thomas Hofmann, and Tiago Pimentel. The non-linear representation dilemma: Is causal abstraction enough for mechanistic interpretability? *arXiv preprint arXiv:2507.08802*, 2025.

Eric Todd, Millicent L. Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. Function vectors in large language models. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. arXiv:2310.15213.

Eric Wallace, Kai Xiao, Reimar Leike, Lilian Weng, Johannes Heidecke, and Alex Beutel. The instruction hierarchy: Training LLMs to prioritize privileged instructions. *arXiv preprint arXiv:2404.13208*, 2024.

Yukang Yang et al. Emergent symbolic mechanisms support abstract reasoning in large language models. *arXiv preprint arXiv:2502.20332*, 2025.

Yuan Yuan et al. From hard refusals to safe-completions: Toward output-centric safety training. *arXiv preprint arXiv:2508.09224*, 2025.