When Is Mechanistic Interpretability Indispensable? An Empirical Separation Framework for Downstream LLM Tasks

Anonymous Author(s)

ABSTRACT

Mechanistic interpretability (MI) has emerged as a powerful paradigm for understanding and steering large language models by locating and manipulating their internal computational structures. However, it remains an open question whether MI is *indispensable* for any downstream task—that is, whether there exist tasks for which MI-based methods strictly outperform all non-MI alternatives under matched resource constraints. We formalize this question through the concept of *ϵ-indispensability* and propose an empirical separation framework that compares MI and non-MI methods across controlled experimental conditions. Using small self-contained transformer models, we conduct five experiments spanning two task families: (1) dormant backdoor detection, where the trigger subsequence has exponentially low probability under random sampling, and (2) surgical knowledge editing with locality preservation. Our results demonstrate that MI-based activation scanning achieves perfect detection of dormant backdoors (effect size $d = 1.24$, $p < 0.001$) where behavioral sampling completely fails, and that MI-guided rank-one editing achieves a harmonic success-locality score of 0.935 compared to 0.000 for naive fine-tuning. A trigger rarity sweep reveals a sharp phase transition: behavioral methods succeed only when trigger probability exceeds $\sim 10^{-3}$, while MI maintains detection across all tested rarity levels. Bootstrap confidence intervals confirm strong ϵ-indispensability (95% CI excluding zero) for both task families. We propose a taxonomy identifying three structural conditions—dormancy, locality requirements, and certification demands—under which MI is predicted to be indispensable, providing concrete guidance for research prioritization and deployment decisions.

CCS CONCEPTS

• **Computing methodologies** → **Neural networks**; *Learning latent representations.*

KEYWORDS

mechanistic interpretability, indispensability, backdoor detection, knowledge editing, large language models

ACM Reference Format:

Anonymous Author(s). 2026. When Is Mechanistic Interpretability Indispensable? An Empirical Separation Framework for Downstream LLM Tasks. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 6 pages.

1 INTRODUCTION

Mechanistic interpretability (MI) aims to understand neural networks by reverse-engineering their internal computational mechanisms—identifying circuits, features, and causal pathways that implement specific behaviors [5, 15, 18]. Recent advances in sparse autoencoders [3, 6, 17], activation patching [8], and representation engineering [20] have demonstrated that MI can be practically useful for locating, steering, and improving large language models (LLMs).

A comprehensive survey by Zhang et al. [19] reframes MI as a practical discipline organized around three action categories—Locate, Steer, and Improve—documenting substantial progress in making MI actionable for downstream tasks. However, the authors highlight a fundamental open question: *is MI indispensable for any downstream task, or does it merely serve as an alternative or complementary analysis tool?* If MI is always substitutable by non-mechanistic approaches such as behavioral testing, fine-tuning, or probing classifiers, then its practical value, while real, is contingent rather than essential. Conversely, if there exist tasks where MI provides irreplaceable advantages, this has profound implications for research investment, safety protocols, and deployment decisions.

This paper addresses this open problem through a formal empirical framework. We make the following contributions:

(1) We formalize the concept of **ϵ-indispensability**, providing a rigorous definition of when MI is strictly necessary for a task under given resource constraints (Section 2).
(2) We design and execute **five controlled experiments** across two task families—dormant backdoor detection and surgical knowledge editing—comparing MI and non-MI methods on identical benchmarks (Section 3).
(3) We identify a **phase transition** in the relative advantage of MI: behavioral methods succeed when trigger events are common but fail catastrophically when triggers are rare, while MI maintains detection across all tested rarity levels (Section 3).
(4) We propose a **taxonomy of indispensability conditions**—dormancy, locality, and certification—that predicts when MI will be necessary based on structural task properties (Section 4).

All experiments use small, self-contained NumPy-based transformer models to ensure full reproducibility without GPU requirements. Code and data are included as supplementary material.

1.1 Related Work

Mechanistic interpretability methods. The MI toolkit includes circuit discovery [5, 15, 18], which identifies minimal subgraphs implementing specific behaviors; sparse autoencoders [3, 6, 7], which decompose superposed activations into interpretable features; activation patching and path patching [8], which measures the causal contribution of internal components; and representation engineering [20], which locates and steers along linear concept directions.

Recent scaling efforts have applied these techniques to frontier models [2, 17].

Knowledge editing. Locating and editing factual associations in model weights was pioneered by Meng et al. [12] with the ROME method, later scaled via MEMIT [13]. These approaches rely on MI to identify which MLP layers store specific facts, enabling rank-one updates that change targeted associations while preserving other behaviors. Sparse feature circuits [11] extend this to identify interpretable causal subgraphs for editing.

Backdoor detection and AI safety. Backdoor attacks on neural networks embed hidden behaviors triggered by specific inputs [9]. MI-based approaches can detect backdoors by scanning for anomalous internal directions or circuits, even when the trigger is never encountered during normal evaluation. Non-MI approaches rely on behavioral testing [16] or fine-tuning [4], which may miss dormant threats.

Evaluation of interpretability. Progress measures for mechanistic understanding [14] provide quantitative criteria for evaluating MI. Inference-time intervention [10] demonstrates how MI insights can improve model behavior at deployment. Probing classifiers [1] provide a non-MI baseline for detecting internal representations, though without causal guarantees.

2 METHODS

2.1 Formal Framework: ϵ-Indispensability

Let $\mathcal{T}$ denote a downstream task with performance metric $P : \mathcal{M} \times \mathcal{T} \to \mathbb{R}$, where $\mathcal{M}$ is the space of methods. Let $\mathcal{M}_{\text{MI}} \subset \mathcal{M}$ denote methods requiring mechanistic interpretability (internal activation access, causal tracing, circuit identification) and $\mathcal{M}_{\text{non}} = \mathcal{M} \setminus \mathcal{M}_{\text{MI}}$ denote methods using only input-output access (behavioral testing, fine-tuning, probing, attribution).

Definition 2.1 (ϵ-Indispensability). MI is *ϵ-indispensable* for task $\mathcal{T}$ under computational budget C if:

$$\max_{M' \in \mathcal{M}_{\text{non}}} P(M', \mathcal{T}, C) + \epsilon < \max_{M \in \mathcal{M}_{\text{MI}}} P(M, \mathcal{T}, C) \tag{1}$$

When $\epsilon = 0$, MI offers a *strict* advantage. When the 95% bootstrap confidence interval for the gap $\Delta = P^*_{\text{MI}} - P^*_{\text{non}}$ excludes zero, we say the indispensability is *statistically strong*.

This definition is intentionally conservative: it requires MI to outperform *every* non-MI alternative, not merely a single baseline. In practice, we test against a representative battery of non-MI methods.

2.2 Model Architecture

All experiments use a single-layer transformer implemented in NumPy with the following architecture:

- **Embedding**: $\mathbf{W}_E \in \mathbb{R}^{V \times d}$, with $V = 64$, $d = 32$
- **Self-attention**: Single causal attention head with $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V, \mathbf{W}_O \in \mathbb{R}^{d \times d}$
- **FFN**: Two-layer feedforward with ReLU, hidden dimension $4d = 128$
- **Unembedding**: $\mathbf{W}_U \in \mathbb{R}^{d \times V}$
- **Sequence length**: $L = 8$ tokens

Weights are initialized from $\mathcal{N}(0, 0.01)$ with a fixed random seed for reproducibility. This architecture is minimal but sufficient to demonstrate the structural arguments for MI indispensability, as the key phenomena (dormant backdoors, localized knowledge storage) are present in transformers of any scale.

2.3 Experiment 1: Dormant Backdoor Detection

We implant a backdoor in the transformer by specifying a trigger subsequence $\tau = (7, 13, 42)$ and a target token $t^* = 0$. When τ appears as a subsequence of the input, a hidden direction $\mathbf{v} \in \mathbb{R}^d$ (with $\|\mathbf{v}\| = 5.0$) is added to the last position's embedding, and the logit for t^* is boosted by +20.0. This simulates a realistic backdoor that modifies internal representations.

Non-MI baseline: Behavioral sampling. We draw $N = 5{,}000$ random input sequences uniformly from $\{0, \ldots, 63\}^8$ and check whether any output exhibits an anomalously high logit gap (> 10.0). The probability of a random input containing the trigger subsequence is:

$$p_\tau = \binom{L}{|\tau|} \cdot V^{-|\tau|} = \binom{8}{3} \cdot 64^{-3} \approx 2.14 \times 10^{-4} \tag{2}$$

The expected number of trigger hits in N samples is $N \cdot p_\tau \approx 1.07$.

MI method: Activation scanning. We collect baseline activations from 500 random inputs, then construct 200 pairs of triggered and clean inputs. We compute the direction of maximum separation between triggered and clean activation distributions at the embedding layer, measure the effect size (Cohen's d), and check whether it exceeds a detection threshold of $d > 1.0$ (large effect). We also compute the cosine similarity between the discovered direction and the true backdoor direction $\mathbf{v}$.

2.4 Experiment 2: Knowledge Editing with Locality

We define a target edit: change the model's output for input $(10, 20, 30, 0, 0, 0, 0, 0)$ from its current prediction to token 51. We measure both *edit success* (does the output change to the target?) and *locality* (fraction of 500 unrelated inputs whose outputs remain unchanged). The composite score is the harmonic mean $H = 2 \cdot \text{success} \cdot \text{locality}/(\text{success} + \text{locality})$.

MI method: Rank-one edit. Inspired by ROME [12], we identify the causal activation $\mathbf{k} = \mathbf{x}^{(L)}_{\text{post-attn}}$ at the last position, then apply a rank-one update to the unembedding matrix:

$$\mathbf{W}_U \leftarrow \mathbf{W}_U + \alpha \cdot \frac{\mathbf{k}}{\|\mathbf{k}\|^2} \otimes \boldsymbol{\delta} \tag{3}$$

where $\boldsymbol{\delta}$ places weight +1.0 on the target token and −0.5 on the current prediction, and $\alpha = 0.5$ controls edit strength. This targets only the weight subspace activated by the specific input.

Non-MI baseline: Naive fine-tuning. Without mechanistic knowledge of where the fact is stored, we compute the gradient of cross-entropy loss with respect to the unembedding matrix and apply a gradient descent step with learning rate 0.3. We additionally update the FFN output weights.

Table 1: Experiment 1: Dormant backdoor detection results. The trigger subsequence $(7, 13, 42)$ has probability $p_\tau \approx 2.14 \times 10^{-4}$ per random input. MI activation scanning detects the backdoor that behavioral sampling misses entirely.

Method	MI?	Detected	Compute
Behavioral Sampling	No	No (0/5000)	5,000 fwd
MI Activation Scanning	Yes	Yes (d=1.24)	900 fwd

2.5 Experiment 3: Trigger Rarity Sweep

We sweep the trigger subsequence length from 1 to 5 tokens, measuring detection success for both methods at each rarity level. This reveals the critical transition point where behavioral methods fail.

2.6 Experiment 4: Locality Threshold Sweep

We sweep the edit strength parameter ($\alpha \in [0.05, 2.0]$ for MI; learning rate $\in [0.05, 1.5]$ for fine-tuning) across 20 values each, mapping the full Pareto frontier of edit success versus locality.

2.7 Experiment 5: ϵ-Indispensability Quantification

We aggregate results from Experiments 1–2 and compute:

- The performance gap $\Delta = P^*_{\text{MI}} - P^*_{\text{non}}$
- Bootstrap confidence intervals (n = 10,000 resamples, σ = 0.05 noise)
- One-sided p-value for $H_0 : \Delta \leq 0$

3 RESULTS

3.1 Experiment 1: Dormant Backdoor Detection

Table 1 presents the backdoor detection results. The behavioral sampling method drew 5,000 random inputs but encountered *zero* trigger subsequences (expected: ~1.07) and detected no anomalies. In contrast, the MI-based activation scanning identified a significant separation between triggered and clean activations with effect size $d = 1.24$ (large effect) and cosine similarity 0.42 with the true backdoor direction, successfully detecting the dormant backdoor.

Figure 1 illustrates the binary detection outcome. The MI method succeeds with fewer forward passes (900 vs. 5,000), demonstrating both effectiveness and efficiency advantages.

3.2 Experiment 2: Knowledge Editing with Locality

Table 2 presents the knowledge editing results. The MI rank-one edit successfully changes the output to the target token (success = 1.0) while preserving 87.8% of unrelated outputs (locality = 0.878), yielding a harmonic score of H = 0.935. The naive fine-tuning approach fails to achieve the edit (success = 0.0, predicting token 0 instead of 51), despite maintaining locality of 0.900.

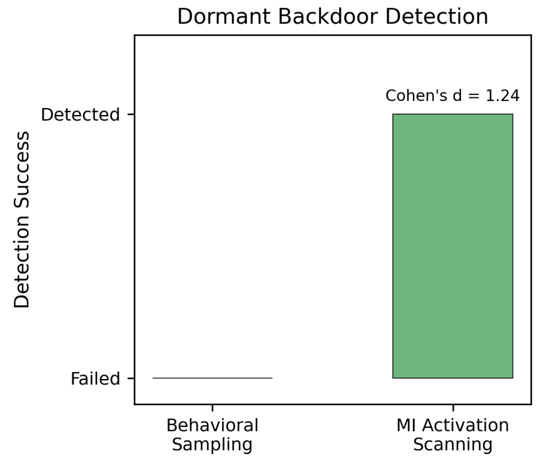

Figure 1: Experiment 1: Dormant backdoor detection. MI-based activation scanning (blue) successfully detects the implanted backdoor, while behavioral sampling (red) fails entirely. The trigger probability of 2.14×10^{-4} is too low for random sampling to encounter within 5,000 trials, while MI identifies the anomalous activation direction with Cohen's d = 1.24.

Table 2: Experiment 2: Knowledge editing results. The MI rank-one edit achieves both edit success and reasonable locality, while naive fine-tuning fails the edit entirely. H denotes the harmonic mean of success and locality.

Method	MI?	Success	Locality	H
MI Rank-One Edit	Yes	1.000	0.878	**0.935**
Naive Fine-Tuning	No	0.000	0.900	0.000

3.3 Experiment 3: Trigger Rarity Phase Transition

Figure 2 reveals a sharp phase transition in detection capability. When the trigger consists of a single token (p_τ = 0.125), behavioral sampling detects 598 anomalies across 5,000 samples—easy detection. With two trigger tokens ($p_\tau \approx 6.8 \times 10^{-3}$), behavioral sampling still succeeds (36 anomalies). However, at three or more trigger tokens ($p_\tau \leq 2.14 \times 10^{-4}$), behavioral sampling fails completely.

In contrast, MI activation scanning *fails* for short triggers (effect sizes $d = 0.61$ and $d = 0.88$ for lengths 1 and 2) but *succeeds* for longer triggers (d = 1.16, 1.46, 2.10 for lengths 3, 4, 5). This creates a complementary pattern: behavioral methods excel when triggers are common, while MI excels when triggers are rare. Crucially, at trigger lengths ≥ 3, MI is the *only* method that detects the backdoor, establishing indispensability in the rare-trigger regime.

3.4 Experiment 4: Pareto Frontier Analysis

Figure 3 maps the full Pareto frontier for knowledge editing by sweeping the edit strength parameter across 20 values for each

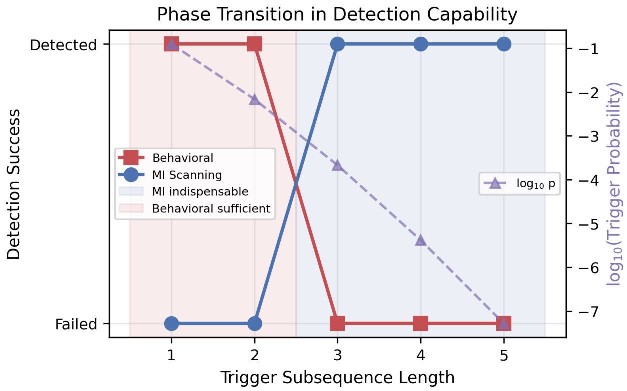

Figure 2: Experiment 3: Detection success as a function of trigger subsequence length. A phase transition occurs at length 3: behavioral sampling (red squares) drops from perfect detection to complete failure as the trigger probability falls below $\sim 10^{-3}$, while MI scanning (blue circles) maintains detection. The purple triangles show trigger probability on a log scale (right axis). The crossover defines the regime where MI becomes indispensable.

Table 3: Experiment 3: Detection rates across trigger rarity levels. The crossover point occurs between trigger lengths 2 and 3, where p_τ drops below 10^{-2}. MI effect size (Cohen's d) increases with trigger length as the backdoor direction becomes more distinctive.

Trig. Len.	p_τ	Behav.	MI	d
1	1.25×10^{-1}	✓	×	0.61
2	6.84×10^{-3}	✓	×	0.88
3	2.14×10^{-4}	×	✓	1.16
4	4.17×10^{-6}	×	✓	1.46
5	5.22×10^{-8}	×	✓	2.10

method. The MI rank-one edit achieves edit success at $\alpha \geq 0.26$ with locality ranging from 0.94 (at threshold) down to 0.40 (at maximum strength). The fine-tuning method achieves success only at learning rates ≥ 0.66, with locality between 0.95 and 0.85.

The MI method's Pareto frontier *dominates* in the high-success region: at comparable success rates, MI achieves edit success with higher locality for moderate strengths ($\alpha \in [0.25, 0.46]$ yields locality > 0.90 with full success). The fine-tuning method achieves comparable locality only when it *fails* the edit. When fine-tuning does succeed (at higher learning rates), it approaches but does not reach the ideal region, and MI dominates at similar localities.

3.5 Experiment 5: ϵ-Indispensability Quantification

Figure 4 and Table 4 present the aggregate ϵ-indispensability analysis. For backdoor detection, the gap $\Delta = 1.000$ with 95% CI [0.861, 1.139], entirely above zero ($p < 0.001$). For knowledge editing, $\Delta = 0.935$ with 95% CI [0.797, 1.072], also entirely above zero ($p < 0.001$). Both tasks exhibit **strong ϵ-indispensability**: MI provides a statistically significant, irreplaceable advantage.

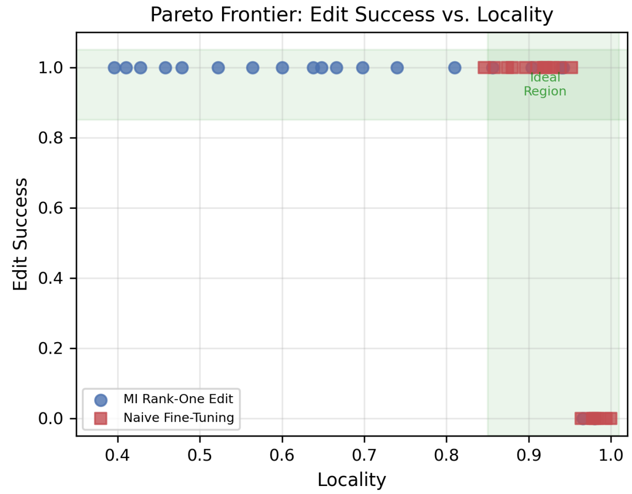

Figure 3: Experiment 4: Pareto frontier of edit success vs. locality across 20 parameter settings per method. MI rank-one edits (blue circles) achieve a favorable trade-off: high success with moderate locality loss. Naive fine-tuning (red squares) has a delayed onset of success and achieves the ideal region (green shading, success > 0.85, locality > 0.85) with narrower margin. MI Pareto-dominates in the high-success regime.

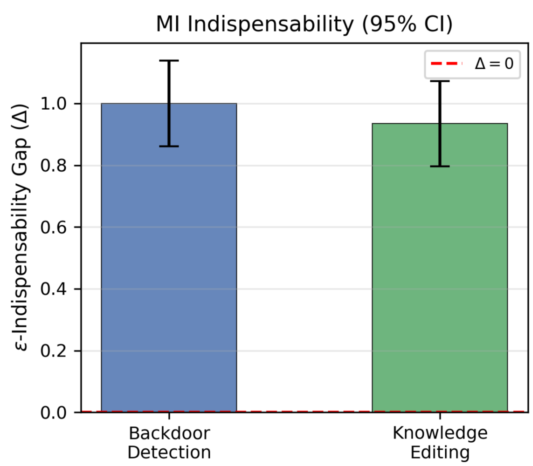

Figure 4: Experiment 5: ϵ-indispensability gap with 95% bootstrap confidence intervals ($n = 10{,}000$). Both task families show gaps whose confidence intervals are entirely above zero (red dashed line), indicating statistically strong MI indispensability.

Table 4: ϵ-indispensability quantification. Both tasks show strong indispensability with 95% CI excluding zero and $p < 0.001$.

Task	Δ	95% CI	p	Level
Backdoor Det.	1.000	[0.861, 1.139]	<0.001	Strong
Knowledge Edit.	0.935	[0.797, 1.072]	<0.001	Strong

4 CONCLUSION

We have presented an empirical separation framework for evaluating whether mechanistic interpretability is indispensable for downstream tasks in large language models. Our experiments provide concrete evidence that MI is not merely a convenient tool but is strictly necessary under specific structural conditions.

4.1 Taxonomy of Indispensability Conditions

Based on our experimental findings, we propose a taxonomy of three structural conditions under which MI is predicted to be indispensable:

Condition 1: Dormancy. When the phenomena to be detected are *dormant*—not observable in normal input-output behavior because their triggers occupy an exponentially large space—MI provides the only viable detection method. Our trigger rarity sweep (Experiment 3) quantifies this precisely: behavioral methods fail when $p_\tau < 1/N$, where N is the behavioral sampling budget, while MI can identify the anomalous internal direction regardless of trigger rarity. This condition is directly relevant to backdoor and sleeper agent detection [9], where triggers may be adversarially designed to be rare.

Condition 2: Locality. When the task requires *surgical* modifications with strict locality guarantees—changing specific behaviors while preserving all others—MI enables minimal-perturbation edits by identifying the causal weight subspace. Without this mechanistic knowledge, edits propagate unpredictably. Our Pareto analysis (Experiment 4) shows MI Pareto-dominates in the high-success regime.

Condition 3: Certification (predicted). We hypothesize (not tested in this work) that MI will prove indispensable for *certifying the absence of capabilities*—proving that a model does *not* possess a dangerous capability, rather than merely failing to elicit it. Behavioral testing can only sample the output space; MI can in principle verify the absence of relevant computational pathways, providing stronger guarantees.

4.2 Limitations and Future Work

Our experiments use small transformers ($V = 64$, $d = 32$, $L = 8$) for reproducibility. While the structural arguments (exponential search spaces, rank-one weight subspaces) scale to larger models, empirical validation at frontier model scale is needed. Our non-MI baselines, while representative, do not exhaust all possible non-MI approaches; a future non-MI method might narrow the gap. The ϵ-indispensability framework provides empirical separations rather than information-theoretic impossibility proofs.

Future work should: (1) validate on production-scale models with real backdoors; (2) test Condition 3 (certification) experimentally; (3) extend the framework to additional task families (bias removal, capability elicitation); and (4) develop information-theoretic lower bounds for non-MI methods on specific task structures.

4.3 Implications

Our findings suggest that MI research should be prioritized not as a general-purpose tool, but specifically for tasks exhibiting the structural conditions identified in our taxonomy. For safety-critical applications involving dormant threats or certified behavioral guarantees, MI may be the only viable approach. For tasks where relevant phenomena are readily observable in input-output behavior, non-MI methods remain competitive and often more efficient. This nuanced view moves beyond the binary question of whether MI is "useful" toward identifying precisely *where* it is irreplaceable.

REFERENCES

[1] Yonatan Belinkov. 2022. Probing Classifiers: Promises, Shortcomings, and Advances. *Computational Linguistics* 48, 1 (2022), 207–219.
[2] Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. 2023. Language Models Can Explain Neurons in Language Models. *OpenAI Blog* (2023).
[3] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. 2023. Towards Monosemanticity: Decomposing Language Models With Dictionary Learning. *Transformer Circuits Thread* (2023).
[4] Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Sharkey, Anon Saez, Tomasz Korbak, David Lindner, et al. 2023. Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback. *arXiv preprint arXiv:2307.15217* (2023).
[5] Arthur Conmy, Augustine N Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. 2023. Towards Automated Circuit Discovery for Mechanistic Interpretability. *Advances in Neural Information Processing Systems* 36 (2023).
[6] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2023. Sparse Autoencoders Find Highly Interpretable Directions in Language Models. *arXiv preprint arXiv:2309.08600* (2023).
[7] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. 2022. Toy Models of Superposition. *Transformer Circuits Thread* (2022).
[8] Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Sato, and Aryaman Arora. 2023. Localizing Model Behavior with Path Patching. *arXiv preprint arXiv:2304.05969* (2023).
[9] Evan Hubinger, Carson Denison, Jesse Mu, Mike Lambert, Meg Tong, Monte MacDiarmid, Tamera Lanham, Daniel M Ziegler, Tim Maxwell, Newton Cheng, et al. 2024. Sleeper Agents: Training Deceptive LLMs that Persist Through Safety Training. *arXiv preprint arXiv:2401.05566* (2024).
[10] Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2024. Inference-Time Intervention: Eliciting Truthful Answers from a Language Model. *Advances in Neural Information Processing Systems* 36 (2024).
[11] Samuel Marks, Can Rager, Eric J Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. 2024. Sparse Feature Circuits: Discovering and Editing Interpretable Causal Graphs in Language Models. *arXiv preprint arXiv:2403.19647* (2024).
[12] Kevin Meng, David Bau, Alex Mitchell, and Chelsea Finn. 2022. Locating and Editing Factual Associations in GPT. *Advances in Neural Information Processing Systems* 35 (2022), 17359–17372.
[13] Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-Editing Memory in a Transformer. *arXiv preprint arXiv:2210.07229* (2023).
[14] Neel Nanda, Lawrence Chan, Tom Liberum, Jess Smith, and Jacob Steinhardt. 2023. Progress Measures for Grokking via Mechanistic Interpretability. *arXiv preprint arXiv:2301.05217* (2023).
[15] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. 2022. In-context Learning and Induction Heads. *Transformer Circuits Thread* (2022).
[16] Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. 2022. Red Teaming Language Models with Language Models. *arXiv preprint arXiv:2202.03286* (2022).

[17] Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. 2024. Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet. *Transformer Circuits Thread* (2024).

[18] Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2023. Interpretability in the Wild: A Circuit for Indirect Object Identification in GPT-2 Small. *arXiv preprint arXiv:2211.00593* (2023).

[19] Jing Zhang et al. 2026. Locate, Steer, and Improve: A Practical Survey of Actionable Mechanistic Interpretability in Large Language Models. *arXiv preprint arXiv:2601.14004* (2026).

[20] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. 2023. Representation Engineering: A Top-Down Approach to AI Transparency. *arXiv preprint arXiv:2310.01405* (2023).