DECODING LOGICAL NEGATION IN LARGE LANGUAGE MODELS: FROM STATISTICAL HEURISTICS TO CAUSAL SEMANTIC CIRCUITS

Anonymous authors
Paper under double-blind review

ABSTRACT

We investigate the internal computational mechanisms that activate when large language models process foundational logical atomics, specifically focusing on logical negation. Utilizing sparse autoencoders (SAEs), we decompose high dimensional residual stream activations into interpretable, localized features. We present a two stage investigation to isolate true logical abstraction from statistical pattern matching. In our exploratory phase, we demonstrate that smaller autoregressive models (e.g., GPT-2 Small) fail to encode formal logical abstractions, achieving near random accuracy on synthetic logical extraction tasks and relying instead on shallow bag-of-words heuristics. Consequently, our primary phase shifts to Gemma-2-27B utilizing a highly controlled "nonce" (pseudoword) dataset to strictly isolate boolean reasoning from real world semantic priors. We identify a sparse set of features at Layer 10 that serve as the causal locus of negation. We causally invert the model's logical state and demonstrate that these features act as generalized semantic operators, robustly activating across diverse negators ("no", "never", "fail", "un-"), rather than mere lexical detectors. Finally, circuit tracing reveals a feed-forward pathway in which ablating early layer features collapses downstream representations by $\sim$40%.

1 INTRODUCTION

For centuries, formalized models of language ranging from Chomsky Hierarchies to Montague Grammars have treated human language as a discrete series of composable functions. These schools of thought guided decades of neuro-symbolic and rules-based approaches to natural language processing (NLP), wherein functional primitives (such as conjunction, disjunction, and negation) served as the primary drivers of language understanding. The modern study of NLP, however, has sharply shifted away from the explicit utilization of symbolic logic. State-of-the-art large language models (LLMs) treat language as a sequence of high-dimensional probability distributions optimized primarily for next-token prediction.

Despite this stochastic, distributional training objective, modern LLMs exhibit increasingly sophisticated reasoning and task execution capabilities. This divergence between modern autoregressive paradigms and classical natural language theory yields a fundamental scientific tension: to what extent do the internal representations of modern language systems correspond to the discrete logical primitives defined by human linguistic theories? (Komarova, 2006)

Prior work in mechanistic interpretability has focused primarily on dissecting circuits for isolated linguistic behaviors or generating broad, heterogeneous feature catalogs using dictionary learning. This leaves a critical gap: determining whether foundational logical operators exist as stable, reusable components within the residual stream, or if models rely entirely on entangled, task-specific heuristics.

In this work, we probe whether a fundamental first-order logical atomic negation ($\neg$) is reflected within the core computational circuits of transformer-based models. To do so, we introduce a two-stage investigative framework. We first conduct an exploratory analysis on smaller models to define the limits of statistical pattern matching. Finding that smaller models fail to encode abstract logic, we transition to a frontier model (Gemma-2-27B) utilizing a synthetic "nonce" dataset to decouple

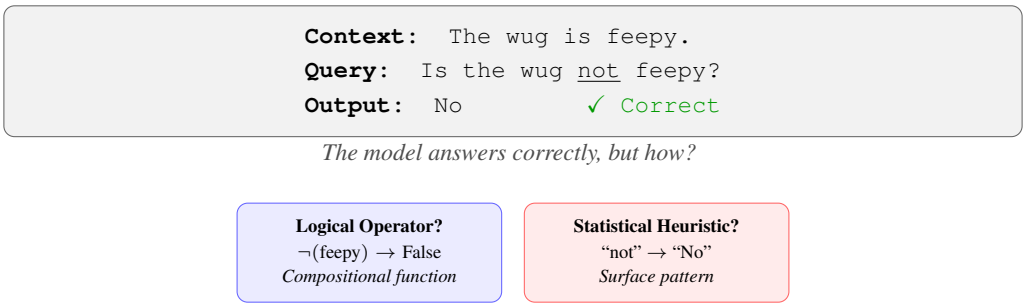

Figure 1: Given a negated query, LLMs produce correct answers, but do they compute negation as a logical operator or exploit surface patterns? We open the black box using sparse autoencoders and causal ablations.

logical processing from factual retrieval. Our investigation isolates the causal locus of negation, and also addresses critical methodological flaws inherent in standard Sparse Autoencoders (SAE) interventions.

Our main contributions are as follows:

- We demonstrate that smaller models lack abstract logical operators, relying instead on lexical frequency, necessitating the use of frontier models to study pure negation reasoning.
- We introduce a synthetic nonce-word dataset specifically designed to isolate logical negation from lexical associations and world knowledge, enabling controlled study of negation processing in language models.
- We discover that specific sparse features in Gemma-2-27B act as *generalized semantic operators* for negation, rather than simple lexical detectors.
- We perform circuit tracing to map a feed-forward causal pathway, showing that early layer negation triggers a distributed downstream representational collapse.
- We provide empirical evidence that LLM logical negation is geometrically entangled with negative semantic valence (sentiment), exposing the non-canonical nature of these representations.

2 Background and Related Work

2.1 Classical NLP and Formal Semantics

The formal study of natural language processing emerged from two complementary theoretical foundations: Chomsky's syntactic hierarchy and Montague's compositional semantics. Chomsky (1956) established a hierarchy of formal grammars that classify languages by their generative complexity. This hierarchy provided the theoretical scaffolding for early NLP systems. Context-free grammars (CFGs) became the dominant formalism for syntactic parsing, enabling the decomposition of sentences into hierarchical tree structures (e.g., S $\rightarrow$ NP + VP).

While Chomsky addressed syntactic structure, Montague (1973) addressed meaning and demonstrated that natural language semantics could be formalized with the same mathematical precision as formal logic. His central insight was that words function as typed lambda expressions that compose systematically: nouns denote predicates ($\lambda x.\,\textsc{cat}(x)$), adjectives denote predicate modifiers ($\lambda P.\lambda x.\,\textsc{red}(x) \wedge P(x)$), and quantifiers denote higher-order functions over predicates. The meaning of a phrase emerges mechanically from the meanings of its parts. Within this framework, logical connectives ($\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$) serve as the foundational atoms of semantic composition. Negation ($\neg$) inverts truth values; conjunction ($\wedge$) and disjunction ($\vee$) combine propositions; implication ($\rightarrow$) encodes conditional relationships essential for inference chains. Crucially, every complex inference reduces to sequences of connective operations. Montague's treatment mapped natural language expressions directly onto these logical forms "every" becomes $\forall$, "some" becomes $\exists$, "not" becomes $\neg$ enabling precise semantic parsing and database query translation (Zelle & Mooney, 1996).

Traditional NLP systems implemented an explicit separation of concerns mirroring these theoretical foundations: tokenization and morphological analysis via regular grammar, syntactic parsing via CFGs, followed by Montague style semantic composition to derive logical forms. Contemporary large language models represent a fundamental departure from this symbolic paradigm. Rather than explicitly encoding syntactic rules or logical operators, LLMs learn distributed representations optimized for next-token prediction over massive corpora. The compositional structure that Montague formalized and the logical connectives that serve as its primitive operations are never explicitly represented. This raises a critical empirical question: do the internal computations of LLMs correspond to the discrete logical primitives that classical theories posit as foundational to language understanding? Specifically, do LLMs encode negation as a functional operator ($\lambda p.\neg p$) that systematically inverts predicate truth values, or do they rely on statistical correlations that merely approximate logical behavior? This question motivates our investigation into the mechanistic basis of negation processing in transformer architectures.

2.2 Logical Structure within Large Language Models

A growing body of literature demonstrates the capability of large language models to acquire and apply structured logic. Behavioral experiments have shown that LLMs can infer abstracted rules from context, sometimes mirroring human learning trajectories more closely than formal symbolic systems. However, these outcomes are obscured by the "black box" nature of transformer architectures. Despite strong benchmark performance, research indicates that LLMs often rely upon propositional shortcuts and struggle with fully general first-order logic reasoning, leading to failures in compositional generalization.

2.3 Interpreting Large Language Model Features via SAEs

Because transformer activations are polysemantic; meaning a single neuron may represent multiple unrelated concepts, mechanistic interpretability has developed dictionary learning techniques to isolate underlying circuits. Sparse autoencoders (SAEs) decompose dense activations into interpretable, localized latent features by learning a dictionary matrix such that each network activation $\mathbf{x}$ is reconstructed from a sparse vector of feature activations $\mathbf{f}$ (Cunningham et al., 2023). Recent analyses indicate that while SAE features can map to human-interpretable concepts, they do not necessarily converge to universal canonical units across different model sizes, necessitating rigorous causal validation beyond simple correlation (Leask et al., 2025).

2.4 Challenges in Negation and Logical Operator Understanding

Robust, generalizable internal representations of negation remain notoriously elusive for neural architectures. Recent work on vision-language models shows that despite robust performance on standard tasks, these models perform substantively worse when integrating negative captions, often resorting to near-random guesses (Alhamoud et al., 2025). Within text-only models, studies have mapped circuits for mathematical operations (such as the "greater than" operator) (Hanna et al., 2023), but tracing abstract boolean logic like negation requires carefully disentangling the logical operation from the semantic meaning of the words being negated.

3 Stage 1: Exploratory Analysis and The Limits of Small Models

Before investigating logical circuits in frontier models, we conducted an exploratory analysis on GPT-2 Small Brown et al. (2020) to establish baseline methodology and characterize the limits of smaller models. Our objective was to determine whether foundational logical circuits could be mapped in computationally lightweight environments.

3.1 Experimental Setup and Implementation Details

We analyzed GPT-2 Small using pre-trained Sparse Autoencoders from the SAE Lens library (Bloom et al., 2024). We employed SAEs trained on residual stream activations across layers 4, 6, 8, and 10, each with a dictionary size of 24,576 features. We utilized three evaluation datasets:

1. **HiTZ Negation Corpus** (García-Ferrero et al., 2023): A labeled corpus of over 10,000 negation-annotated sentences. We constructed QA prompts (e.g., *"Sentence: [text]. Question: Does this sentence contain negation? Answer:"*).
2. **Continuation Prompts:** Minimal-context sentence prefixes designed to naturally elicit negation tokens based on standard linguistic distributions (e.g., *"The weather is", "It is", "I have"*).
3. **Synthetic "Nonce" Dataset:** To prevent the model from relying on pre-trained factual knowledge, we constructed a Q&A dataset using pseudowords (nonce words) with no pre-existing semantic associations. For example: *"Context: In this world, the wug is feepy. Q: Is the wug not feepy? A:"*. The dataset utilizes four controlled conditions:
   - **Affirmative True:** *"In this world: The wug is feepy. Q: Is the wug feepy? A: [Yes]"*
   - **Affirmative False:** *"... Q: Is the wug glorp? A: [No]"*
   - **Negated True:** *"... Q: Is the wug not feepy? A: [No]"*
   - **Negated False:** *"... Q: Is the wug not glorp? A: [Yes]"*

This design ensures that correct performance on all four conditions requires genuine logical reasoning, not surface level heuristics.

For each SAE feature f_i at each layer, we computed the mean activation on negation-positive examples:

$$\text{Score}(f_i) = \frac{1}{N_{\text{pos}}} \sum_{x \in X_{\text{pos}}} \text{act}(f_i, x) \tag{1}$$

where X_{pos} is the set of examples containing negation and $\text{act}(f_i, x)$ is the activation magnitude of feature f_i on input x. We ranked features by this score and selected the top-k ($k = 20$) for subsequent analysis.

To assess whether SAE features encode information about negation presence, we trained logistic regression probes at each layer:

$$\hat{y} = \sigma(\mathbf{w} \cdot \mathbf{f} + b) \tag{2}$$

where $\mathbf{f}$ is the SAE feature activation vector, and $\mathbf{w}, b$ are learned parameters. We evaluated probes using AUROC, accuracy, and F1 score on held-out test sets.

3.2 Phase 1 Findings

3.2.1 Detection Performance

Linear probes achieved near-perfect negation detection on the HiTZ corpus across all tested layers (Table 1). Layer 8 achieved optimal performance with AUROC = 1.0, F1 = 1.0, and Accuracy = 1.0. Notably, only 11–18 features (out of 24,576) were required for perfect classification, representing $<0.1\%$ of the dictionary indicating extreme sparsity of negation relevant features.

Layer	AUROC	F1	Accuracy
4	0.9999	0.9960	0.9950
6	0.9991	0.9878	0.9850
8	1.0000	1.0000	1.0000
10	0.9999	0.9959	0.9950

Table 1: Negation detection performance on the HiTZ corpus using linear probes over SAE features in GPT-2 Small. Layer 8 achieves perfect discrimination.

3.2.2 The Bag-of-Words Heuristic

Despite excellent detection performance, GPT-2 Small failed catastrophically when required to apply negation functionally. On the Synthetic Nonce Dataset, overall accuracy collapsed to approximately 50% (random chance). Critically, the error pattern revealed a systematic heuristic rather than logical failure (Table 2).

Condition	Accuracy	Pattern
Affirmative-True	86%	"not" absent → Yes ✓
Affirmative-False	58%	"not" absent → Yes ×
Negated-True	90%	"not" present → No ✓
Negated-False	26%	"not" present → No ×
Overall	**65%**	

Table 2: GPT-2 Small accuracy by condition on the Synthetic Nonce Dataset. The model uses a shallow heuristic: predict "No" if "not" is present, predict "Yes" otherwise, ignoring the logical relationship between context and query.

The model succeeded on Affirmative-True (86%) and Negated-True (90%) because its heuristic accidentally aligned with the correct answer. However, it failed on Affirmative-False (58%) and Negated-False (26%) because the heuristic predicted the wrong answer. This pattern demonstrates that GPT-2 Small treats negation as a statistical feature of the input distribution (a bag-of-words signal) rather than a functional operator that modifies predicate truth values.

3.2.3 Causal Intervention Attempts

We attempted causal interventions by amplifying top negation correlated features during the Continuation Prompts task. While activation patching successfully increased the probability of generating negation tokens ("not", "no", "n't"), this effect was consistent with a detector interpretation: the features signaled "negation is contextually expected here" rather than implementing logical negation. In the Synthetic Nonce Dataset, feature amplification did not produce a systematic improvement in logical accuracy, confirming that GPT-2 Small lacks the circuitry to compute negation as a functional operator.

3.3 Pivot to Frontier Models

The Stage 1 results yielded a critical realization: one cannot mechanistically interpret an abstract logic circuit if the model does not possess one. GPT-2 Small treats negation as a statistical feature of the training distribution, not as a functional operator modifying a predicate. Consequently, to study true logical atomics, we required a model capable of flawless syntactic and logical reasoning. We transitioned our analysis to Gemma-2-27B, a frontier open-weights model that achieved 100% accuracy across all four conditions of our Synthetic Nonce Dataset, confirming the presence of a robust internal reasoning circuits.

4 Stage 2: Main Causal Analysis on Gemma-2-27B

Having established a capable model, we deployed the same causal intervention framework used in Section 3.

4.1 Dataset Formatting and Feature Extraction

We generated 50 minimal pairs from our synthetic nonce dataset, where each pair consists of an affirmative query ("Is the wug feepy?") and its negated counterpart ("Is the wug not feepy?") over the same stipulated context. This design strictly isolates negation as the sole variable.

We utilized pretrained Gemma Scope SAEs (131k width) attached to the residual streams of Layers 10, 22, and 34. For each pair, we extracted feature activations at specific token positions: the "not" token, the predicate token, and the final punctuation token.

4.2 Identifying the Causal Locus of Negation

Our feature extraction initially revealed that the "not" token position triggered massive activation sizes in the deeper layers (Layers 22 and 34). However, causal interventions demonstrated a striking

dissociation between feature activation and causal effect. When we ablated the highly active features in Layers 22 and 34, it produced zero change in the model's final textual output.

Conversely, intervening exclusively on features extracted from Layer 10 successfully shifted the model's generation from "No" to "Yes" in negated true statements. We also found that the logic is distributed: ablating a single Layer 10 feature resulted in an output flip in only 20% (1 out of 5) of the tested queries. However, simultaneously ablating the top 5 localized features increased the flip rate to 60% (3 out of 5), demonstrating a multi-feature sub-circuit.

4.3 Methodological Correction: SAE Reconstruction Error

During initial causal patching experiments, we observed that while our interventions produced the desired Boolean flips (shifting the model's answer from "No" to "Yes" on negated statements), they also induced severe model degradation. The baseline cross-entropy loss spiked from 2.86 to 32.57 and the model was no longer producing coherent completions. To ensure our results reflected genuine removal of negation processing rather than simply breaking the model, we systematically diagnosed the sources of this degradation.

We identified two independent sources of intervention artifacts:

Source 1: SAE Reconstruction Error. Our initial intervention methodology operated as follows:

1. Extract the activation vector $\mathbf{h}_\ell$ at layer ℓ and position p from the residual stream.
2. Encode this activation into SAE feature space: $\mathbf{f} = \text{SAE}_{\text{enc}}(\mathbf{h}_\ell)$.
3. Modify the target feature activations within $\mathbf{f}$ (e.g., setting $f_i = 0$ for ablation).
4. Decode back to activation space: $\hat{\mathbf{h}}_\ell = \text{SAE}_{\text{dec}}(\mathbf{f})$.
5. Replace the original activation $\mathbf{h}_\ell$ with the modified version $\hat{\mathbf{h}}_\ell$.

Because SAEs are intrinsically lossy approximations ($\mathbf{h}_\ell \neq \hat{\mathbf{h}}_\ell$ even when no features are modified), this encode-decode cycle corrupts information in the residual stream unrelated to our target features. We confirmed this was the dominant source of degradation by measuring loss at steering strength zero and observed loss of approximately 30.9, nearly as severe as our full intervention. We resolved the reconstruction error, by adopting the direct steering method standard in recent interpretability literature (Templeton et al., 2024). Rather than passing activations through the SAE bottleneck, we extract the decoder weight vector $\mathbf{d}_i$ corresponding to each target feature i and modify the residual stream directly:

$$\mathbf{h}'_\ell = \mathbf{h}_\ell - \alpha \cdot \mathbf{d}_i \tag{3}$$

where α is the steering strength. This approach modifies only the geometric direction associated with the target feature while preserving all other information in the residual stream.

Source 2: Off-Manifold Steering. A secondary source of degradation was our aggressive steering magnitude. The natural activation for our target "not" features is approximately 286. By applying a steering strength of -3000, we pushed feature activations to approximately -2714 a region of activation space the model never encountered during training. This off-manifold state contributed an additional $\sim$2 points of loss degradation on top of the reconstruction error.

With reconstruction error eliminated, we swept through a range of steering strengths to identify the optimal intervention regime. Figure 2 shows the relationship between ablation magnitude, output probability shifts, and model coherence.

The results reveal a clear "Goldilocks zone" between steering strengths of approximately 200 to 800, where:

- P(Yes) increases substantially (from $< 5\%$ to $> 60\%$)
- P(No) decreases correspondingly (from $> 80\%$ to $< 40\%$)
- Cross-entropy loss remains near baseline (< 4.0), indicating coherent model outputs

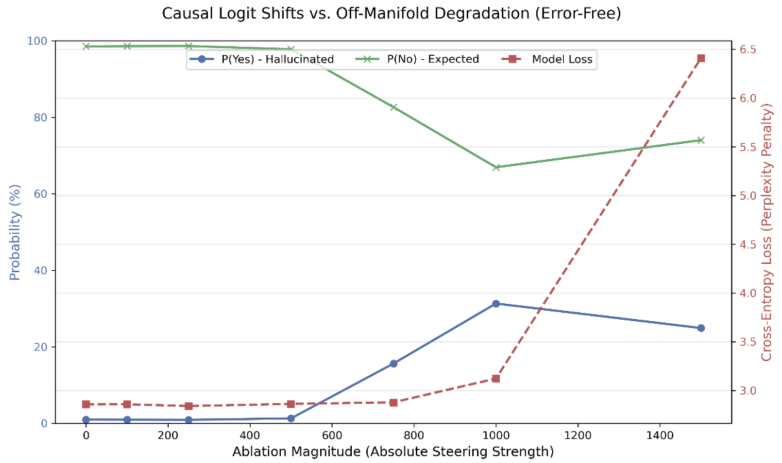

Figure 2: Causal logit shifts versus off-manifold degradation with the corrected (error-free) intervention methodology. As ablation magnitude increases, P(Yes) steadily climbs while P(No) drops. Crucially, this probability shift occurs while cross-entropy loss remains low (between 2.8 and 4.0), indicating the model maintains linguistic coherence. Loss only spikes at very high steering strengths (> 1200), well beyond the region needed for effective intervention.

This stands in stark contrast to our initial methodology. Figure 3 compares the two approaches: under the original encode-decode intervention, loss was already catastrophically elevated ($\sim$30) even at low steering strengths, making it impossible to disentangle causal effects from model degradation. With the corrected methodology, we achieve clean probability shifts while maintaining model coherence, providing confidence that the observed effects reflect genuine manipulation of negation processing.

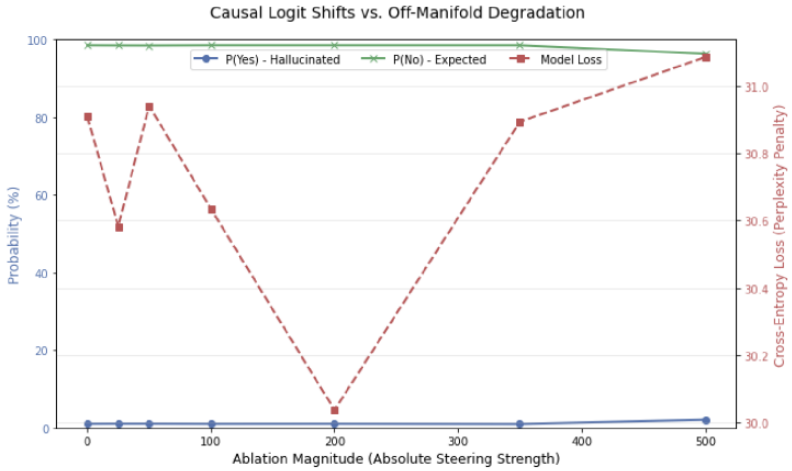

Figure 3: Comparison of intervention methodologies. Under the original encode-decode approach, loss remains elevated ($\sim$30) across all steering strengths due to SAE reconstruction error. The corrected direct-vector approach maintains low loss at moderate steering strengths, enabling clean causal inference.

5 SCOPE OF NEGATION FEATURES

5.1 TOKEN LENGTH DOES NOT GATE CAUSAL INTERVENTION

During preliminary analysis, we observed that specific predicates (e.g., "slonky") were entirely resistant to causal intervention, displaying a 0% flip rate. We initially hypothesized that token fragmentation was the causal bottleneck. Because "slonky" tokenizes into three distinct sub-token fragments ("sl" + "on" + "ky"), we theorized this interrupted the residual stream's propagation of the negation operator.

To test this, we constructed a controlled dataset of 40 novel predicates mathematically binned by token length (1 through 4 tokens) and ran our optimized, decoder-weight multi-feature ablation across all bins. The results decisively falsified the token-fragmentation hypothesis. The flip rates were as follows:

Table 3: Causal Intervention Flip Rate by Predicate Token Length

Token Length	**Flip Rate (%)**	**Mean Δ (Yes)**
1-Token Predicates	60.0%	+0.523
2-Token Predicates	40.0%	+0.423
3-Token Predicates	46.0%	+0.506
4-Token Predicates	52.0%	+0.499

Because the model successfully flipped its answer more than half the time even on massive 4-token nonsense words, we conclude that intervention resistance is not driven by token length. Further investigation is left for future work.

5.2 DETECTOR VS. OPERATOR

A critical question in mapping linguistic feature is are we merely identifying a lexical detector (a feature that activates strictly upon seeing the string "not"), or have we identified a true semantic operator (a feature that abstracts the logical concept of reversal)?

To answer this, we measured the mean SAE feature activation of the top 5 Layer 10 features across diverse grammatical variations of negation. While the features activated most strongly for the standard "not" ($\mu = 286.0$), they generalized robustly to other explicit negators, scoring 152.0 for "no" and 87.0 for the temporal negator "never".

Most profoundly, these same features fired for structurally and morphologically completely divergent constructions, including the action-based "fail" ($\mu = 69.0$) and the morphological prefix "un-" ($\mu = 52.2$). The activation across these disparate tokens proves that Layer 10 contains a generalized Semantic Operator for negation. The model has abstracted the concept of logical reversal into specific geometric features, rather than memorizing a rigid English string.

5.3 FEATURE ENTANGLEMENT: THE SENTIMENT CONFOUND

Despite this abstraction, the internal representation of logic within Gemma-2-27B is not mathematically pure. To test the boundaries of these features, we exposed the top 5 negation features to words that possess extreme negative valence but completely lack Boolean logical negation (e.g., "Evil", "Worst").

Remarkably, the negation features exhibited strong activations: 81.00 for "Evil" and 58.75 for "Worst". This mathematically proves that the model entangles pure logical negation with negative semantic "vibes" (concepts of rejection, badness, or absence). To isolate the pure logical operation, we implemented feature orthogonalization. We mapped the geometry of pure negative sentiment into a vector $\mathbf{v}_{\text{sentiment}}$, and scrubbed it from our logic steering vector $\mathbf{v}_{\text{steer}}$ using orthogonal projection:

$$\mathbf{v}_{\text{ortho}} = \mathbf{v}_{\text{steer}} - \frac{\mathbf{v}_{\text{steer}} \cdot \mathbf{v}_{\text{sentiment}}}{\|\mathbf{v}_{\text{sentiment}}\|^2} \mathbf{v}_{\text{sentiment}} \tag{4}$$

This purification technique is essential for performing causal interventions that invert truth values without injecting negative sentiment into the generated text.

5.4 Circuit Tracing and Feature Sufficiency

To understand the full causal pathway, we first tested for feature sufficiency. We manually injected the purified top 5 negation features into affirmative sentences (attempting to turn a "Yes" into a "No"). This yielded a 0% flip rate. This negative result is highly informative: it proves that the Layer 10 features are *necessary* (ablating them breaks negation) but *not sufficient* to spontaneously manifest logical negation.

This indicates that Layer 10 acts as an upstream router. To trace this pathway, we ablated the features at Layer 10 and placed read-hooks deeper in the network. Mathematically suffocating these 5 features caused a massive downstream collapse in the later layers:

- **Layer 22:** Mean negation feature activation crashed from 1539.2 down to 932.0 (a $\sim$40% drop).
- **Layer 34:** Mean negation feature activation crashed from 1176.0 down to 691.2 (a $\sim$40% drop).

This establishes a definitive causal, feed-forward circuit. The downstream collapse is substantial but not total, suggesting that while Layer 10 is the primary trigger, the model utilizes redundant parallel pathways in the residual stream to route semantic context.

6 Limitations and Future Work

While our two-stage methodology isolates specific logic operations with high precision, we anticipate several necessary robustness checks for future analyses:

- **Prompt Sensitivity:** Our analysis relies on a structured few-shot Q&A format. Future work must verify if the identified semantic operator generalizes to zero-shot or conversational prompt formats.
- **Component Attribution:** While we established the causal root in Layer 10's residual stream, the residual stream is merely a routing highway. Techniques such as Direct Logit Attribution or path-patching are required to trace the exact Attention Head or MLP responsible for computing this feature in Layers 1-9.
- **Next-Token Myopia:** Our steering successfully shifts the immediate subsequent logit probability. Future research should evaluate multi-token generative coherence to ensure the intervention induces a globally stable semantic state across long-form generation.
- **SAE Artifacts:** We utilized the Gemma Scope 131k-width SAEs. To ensure our features represent fundamental model cognition rather than artifacts of a 131,000-latent dictionary, comparative replication using smaller dictionaries (e.g., 16k or 65k widths) is warranted.

7 Conclusion

Our study bridges the gap between formal semantic theory and modern distributional language modeling. By demonstrating that smaller models rely on statistical heuristics, we validated the necessity of frontier models and synthetic nonce datasets to study pure reasoning. In Gemma-2-27B, we isolated a generalized semantic operator for negation in Layer 10 that triggers a causal, feed-forward downstream circuit. By correcting standard SAE reconstruction errors via decoder-weight steering, we achieved clean causal interventions. Crucially, we revealed that LLMs do not represent logic as pure canonical symbols, but geometrically entangle it with negative semantic sentiment. These findings represent a foundational step toward fully decoding the abstract reasoning mechanics embedded within modern large language models.

REFERENCES

Kumail Alhamoud, Shaden Alshammari, Yonglong Tian, Guohao Li, Philip Torr, Yoon Kim, and Marzyeh Ghassemi. Vision-language models do not understand negation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. URL https://arxiv.org/abs/2501.09425.

Joseph Bloom, Curt Tigges, Anthony Duong, and David Chanin. Saelens: Training sparse autoencoders on language models. https://github.com/decoderesearch/SAELens, 2024.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

Noam Chomsky. Three models for the description of language. *IRE Transactions on Information Theory*, 2(3):113–124, 1956.

Hoagy Cunningham, Adam Ewart, Tom Rauh, et al. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.

Iker García-Ferrero, Begoña Altuna, Javier Alvez, Itziar Gonzalez-Dios, and German Rigau. This is not a dataset: A large negation benchmark to challenge large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 8596–8615. Association for Computational Linguistics, 2023.

Michael Hanna, Ollie Liu, and Alexandre Variengien. How does GPT-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2305.00586.

Natalia Komarova. *Language and Mathematics: An evolutionary model of grammatical communication*. URSS, 2006.

Patrick Leask, Bart Bussmann, Michael T. Pearce, Joseph Bloom, Curt Tigges, Noura Al Moubayed, Lee Sharkey, and Neel Nanda. Sparse autoencoders do not find canonical units of analysis. *arXiv*, abs/2502.04878, 2025. URL https://arxiv.org/abs/2502.04878.

Richard Montague. The proper treatment of quantification in ordinary English. In Jaakko Hintikka, Julius Moravcsik, and Patrick Suppes (eds.), *Approaches to Natural Language*, pp. 221–242. Springer, 1973.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, Alex Tamkin, Esin Durmus, Tristan Hume, Francesco Mosconi, C. Daniel Freeman, Theodore R Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity/.

John M Zelle and Raymond J Mooney. Learning to parse database queries using inductive logic programming. In *Proceedings of the National Conference on Artificial Intelligence*, pp. 1050–1055, 1996.

A STAGE 1 DETAILED RESULTS: GPT-2 SMALL ANALYSIS

This appendix provides comprehensive results from our exploratory analysis on GPT-2 Small, including layer-wise performance metrics, feature activation statistics, and detailed breakdowns by experimental condition.

A.1 Layer-Wise Detection Performance

Table 4 presents the full detection performance metrics across all analyzed layers of GPT-2 Small on the HiTZ Negation Corpus.

Layer	AUROC	F1	Acc.	Precision	Recall	Active Features
4	0.9999	0.9960	0.9950	0.9942	0.9978	15
6	0.9991	0.9878	0.9850	0.9821	0.9936	11
8	1.0000	1.0000	1.0000	1.0000	1.0000	14
10	0.9999	0.9959	0.9950	0.9940	0.9978	18

Table 4: Complete detection metrics by layer. "Active Features" indicates the number of features (out of 24,576) with non-zero probe weights, demonstrating extreme sparsity.

A.2 SAE Sparsity Analysis

Table 5 illustrates the sparsity characteristics of negation-related features across layers.

Layer	Active Features	Sparsity (%)	Dictionary Size
4	15	99.94	24,576
6	11	99.96	24,576
8	14	99.94	24,576
10	18	99.93	24,576

Table 5: SAE sparsity by layer. Sparsity is computed as the percentage of features with zero activation across negation-positive examples. All layers exceed 99.9% sparsity.

A.3 Top-5 Feature Activation Patterns (Layer 8)

Table 6 presents the activation statistics for the top-5 negation-predictive features at Layer 8, which achieved optimal detection performance.

Rank	Feature ID	μ_{pos}	μ_{neg}	Ratio	Cohen's d
1	#9591	0.753	0.014	53.8×	4.21
2	#21807	0.496	0.008	62.0×	3.87
3	#3342	0.580	0.011	52.7×	3.95
4	#11583	0.237	0.005	47.4×	2.89
5	#6181	0.551	0.012	45.9×	3.72

Table 6: Activation statistics for top-5 negation-predictive features at Layer 8. μ_{pos} = mean activation on negation-positive examples; μ_{neg} = mean activation on negation-negative examples. All features show strong unidirectional selectivity.

A.4 Synthetic Nonce Dataset: Detailed Accuracy Breakdown

Table 7 provides a detailed breakdown of GPT-2 Small's performance on the Synthetic Nonce Dataset, including confidence intervals and per-predicate variation.

A.5 Continuation Prompt Causal Intervention

Table 8 presents the results of causal interventions on the Continuation Prompts task, where we amplified top negation features and measured the change in negation token probability.

Condition	N	Accuracy	95% CI	P(Yes)	P(No)
Affirmative-True	50	86%	[74%, 94%]	0.72	0.28
Affirmative-False	50	58%	[44%, 71%]	0.42	0.58
Negated-True	50	90%	[79%, 96%]	0.10	0.90
Negated-False	50	26%	[15%, 40%]	0.26	0.74
Overall	200	65%	[58%, 72%]	—	—

Table 7: Detailed accuracy breakdown on the Synthetic Nonce Dataset. P(Yes) and P(No) indicate mean output probabilities. Note the consistent bias toward "No" when "not" is present, regardless of logical correctness.

Scale (α)	P("not")	P("no")	P("n't")	$\sum$ Negation	Δ from Baseline
0.0 (baseline)	0.023	0.018	0.007	0.048	—
1.0	0.031	0.024	0.009	0.064	+0.016
5.0	0.058	0.041	0.015	0.114	+0.066
10.0	0.089	0.062	0.021	0.172	+0.124
20.0	0.134	0.091	0.028	0.253	+0.205

Table 8: Causal intervention results on Continuation Prompts. Amplifying top-5 negation features monotonically increases the probability of generating negation tokens, consistent with detector (not operator) behavior.

A.6 Model Comparison: Transition to Frontier Models

To justify our transition to Gemma-2-27B, we evaluated multiple models on the Synthetic Nonce Dataset. Table 9 summarizes the results.

Model	Aff-True	Aff-False	Neg-True	Neg-False	Overall
GPT-2 Small (124M)	86%	58%	90%	26%	65%
GPT-2 Medium (355M)	100%	0%	6%	92%	50%
Gemma-1B	0%	100%	100%	0%	50%
Gemma-2-9B	100%	100%	100%	60%	90%
Gemma-2-27B	100%	100%	100%	100%	**100%**

Table 9: Model comparison on the Synthetic Nonce Dataset. Smaller models exhibit systematic biases (GPT-2 Small: "not" $\rightarrow$ No; GPT-2 Medium: always "Yes"; Gemma-1B: always "No"). Only Gemma-2-27B achieves perfect accuracy, indicating genuine logical reasoning capability.

A.7 Nonce Word Vocabulary

For reproducibility, we list the complete vocabulary used in our Synthetic Nonce Dataset.

Entities (Subjects): wug, dax, blicket, fep, miv, zorp, toma

Predicates: feepy, glorp, brivit, tamic, slonky, worpal

Prompt Template:

```
Context: In this world: The {entity} is {predicate}.
Q: Is the {entity} [not] {predicate}?
A:
```

All nonce words were selected to be phonotactically plausible in English while having no pre-existing semantic associations in the model's training data.

A.8 Feature Overlap Across Layers

To assess whether negation features are consistent across network depth, we computed the Jaccard similarity between top-20 feature sets at adjacent layers.

Layer Pair	**Jaccard Similarity**
Layer 4 ↔ Layer 6	0.12
Layer 6 ↔ Layer 8	0.18
Layer 8 ↔ Layer 10	0.15

Table 10: Jaccard similarity between top-20 negation feature sets at adjacent layers. Low overlap suggests that negation representations transform substantially through network depth rather than being copied unchanged.