From Indirect Object Identification to Syllogisms: Exploring Binary Mechanisms in Transformer Circuits

Anonymous Author(s)

Affiliation Address email

Abstract

Transformer-based language models (LMs) can perform a wide range of tasks, and mechanistic interpretability (MI) aims to reverse engineer the components responsible for task completion to understand their behavior. Previous MI research has focused on linguistic tasks like Indirect Object Identification (IOI). In this paper, we investigate the ability of GPT-2 small to handle binary truth values by analyzing its behavior with syllogistic prompts, such as "Statement A is true. Statement B matches statement A. Statement B is", which requires more complex logical reasoning compared to IOI. Through our analysis of several syllogism tasks of varying difficulty, we identify multiple circuits that mechanistically explain GPT-2's logical reasoning capabilities and uncover binary mechanisms that facilitate task completion, including the ability to produce a negated token that does not appear in the input prompt through negative heads. Our evaluation using a faithfulness metric shows that a circuit comprising five attention heads achieves over 90% of the original model's performance. By relating our findings to IOI analysis, we provide new insights into the roles of certain attention heads and MLPs in LMs. We believe these insights contribute to a broader understanding of model reasoning and benefit future research in mechanistic interpretability.

1 Introduction

2

3

8

9

10

12

13

14

15

16

17

26

27

28

29

30

33

Despite the success of Large Language Models (LLMs) and their amazing capabilities, these models remain largely opaque and function as black boxes. Mechanistic interpretability has emerged as a field dedicated to mitigate this conceptual gap. By analyzing how LMs solve specific tasks (Wang et al., 2022; Hanna et al., 2023; Merullo et al., 2024), studying emergent behaviors (Arditi et al., 2024), and identifying patterns within their architectures (Gurnee et al., 2024), researchers aim to unravel the inner workings of LMs. Even though great progress has been made, significant gaps remain in understanding LMs even on basic tasks.

GPT-2 is a family of representative LLMs that has been frequently studied in mechanistic interpretability literature. An exemplary case is analyzing its ability to do Indirect Object Identification task (Wang et al., 2022), which reverse engineers how GPT-2 correctly predicts the final token in sentences like "When Mary and John went to the shops, John gave a bottle of milk to". Such mechanistic analysis begins with the output and traces back to identify the architectural components relevant to the task, termed as *circuit*. GPT-2 small has been shown to be competent in such linguistic tasks, however its ability and mechanism to perform logic reasoning remains uncertain. Specifically, it lacks the capability to coherently answer true-false questions, such as "True or False? Dogs have four legs.". To investigate how GPT-2 represents and processes truth values, we utilize syllogism tasks — a classic form of logical reasoning involving premises and a conclusion. By applying similar

Statement A and statement B are opposite. Statement A is true. Statement B is

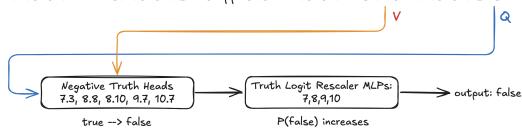


Figure 1: Opposite Syllogism Circuit. The blue arrow represents queries, and the yellow arrow represents values. Negative Truth Heads perform negation of the truth value present in the prompt while the Truth Logit Rescaler MLPs rescale the residual stream to select the correct truth value.

mechanistic analysis to syllogistic prompts, we aim to discover the circuits that are relevant to the task and interpret the internal mechanisms GPT-2 uses when handling truth values.

This paper builds on previous interpretability research by focusing on how GPT-2 handles syllogistic prompts. We use true-false syllogism tasks where truth values are assigned to premise statements and the model is prompted to predict the truth value of the conclusion. We define three prompt formats to probe binary reasoning. The *Simple Syllogism* (SS) presents direct entailment, e.g., "*Statement A is true. Statement B matches statement A. Statement B is*". The *Opposite Syllogism* (OS) inverts this logic, requiring negation, e.g., "*Statement A and Statement B are opposite. Statement A is true. Statement B is*". The *Complex Syllogism* (CS) adds one or more distractor premises irrelevant to the inference, e.g., "*Statement A is true. Statement B matches statement A. Statement C is false. Statement B is*", where the distractor is "*Statement C is false*".

Our approach includes two mechanistic interpretability techniques: Path Patching and Logit Lens. 47 Path patching (Wang et al., 2022) determines the importance of a computational component in solving 48 a task by replacing part of the model's forward pass with activations from a different distribution. 49 Logit Lens (Nostalgebraist, 2020) applies the model's unembedding matrix at different stages of 50 the residual stream, exposing logits and offering insights into the function of specific components 51 during the model's processing. Using these techniques, we apply a mechanistic lens to uncover how 52 LMs perform complex reasoning tasks and identify the key components that drive their decisions. 53 Specifically, we examine the internal mechanisms responsible for **negation** and **reinforcement** of 54 55 truth values. Evaluation with a circuit faithfulness metric shows that a circuit of three attention heads can recover 90% of the original model's performance on SS prompts. For OS prompts, a circuit 56 of five attention heads and four MLPs nearly recovers the performance of the full GPT-2 model, 57 achieving roughly 85% faithfulness. The structure of the OS circuit is shown in Figure 1. 58

Throughout our investigation into how GPT-2 processes syllogisms, we uncover several insights into its internal mechanisms and reasoning capabilities. Our contributions include:

61

62

63

64

65

66

67

68

69

70

- 1. **Discover Syllogism-Specific Circuits**: We discover circuits that represent the internal mechanisms through which GPT-2 solves syllogisms of varying complexity.
- 2. **Identify a Negation Mechanism**: We identify a novel mechanism for outputting the **negation** of a truth value. Attention heads suppress the truth logit and MLPs modulate the **negation** of the truth logit in the output distribution.
- 3. Explain Importance of Negative Components: Through analysis of a pair of semantically opposite tasks (SS and OS), we demonstrate that components critical for one task often have corresponding negative counterparts that play a causally important role in the opposite task. This provides new insights into how language models process and represent binary pairs of tokens.

71 2 Preliminary

92

93

94

72 Transformers Circuits

We provide a brief overview of GPT-2 following the notation from Elhage et al. (2021). GPT-2 is a decoder-only transformer with 12 layers; each layer contains 12 attention heads and one MLP. Input tokens t are embedded into the initial residual stream state x_0 . The residual stream, a core

intermediate representation, is updated additively as it passes through each layer's components

At layer i, the residual stream x_{i-1} is processed by the layer's components and updated as follows:

$$x_i = x_{i-1}$$

+ AttentionHeads (x_{i-1})
+ MLP $(x_{i-1} + \text{AttentionHeads}(x_{i-1}))$.

Here, the attention heads process x_{i-1} in parallel, and their combined output is added back to the residual stream before passing through the MLP, whose output is then added residually to form x_i .

Each attention head is parameterized by four matrices: query W_Q , key W_K , value W_V , and output

 W_O , which form the following composite matrices:

$$W_{QK} := W_Q^\top W_K, \quad W_{OV} := W_O W_V.$$

Using these matrices, along with the embedding matrix W_E and the unembedding matrix W_U , the attention computation for each head decomposes into two core circuits.

The Query-Key (QK) circuit, defined as $W_E^\top W_{QK}W_E$, provides the attention scores for every query-key token pair. Intuitively, each entry describes how much a given query token wants to attend to a given key token, providing insights where information flows within the model.

The *Output-Value (OV) circuit*, defined as $W_UW_{OV}W_E$, determines *what* information is transferred to the output logits when a token is attended to.

This formulation also allows transformers to be represented as a computational graph, where nodes correspond to components like attention heads or MLPs, and edges represent learned weights. Circuits, subgraphs specialized for particular tasks, can then be identified and studied mechanistically.

Indirect Object Identification Wang et al. (2022) analyzed GPT-2 small's performance on the IOI task, where the model must predict the indirect object (IO) in sentences like: "When Mary and John went to the store, John gave a bottle of milk to Mary." The correct prediction is "Mary", not the repeated subject "John".

A human-interpretable strategy to solve IOI involves three steps: (1) identify all names in the sentence, (2) remove duplicates, and (3) output the remaining name. GPT-2 small mirrors this algorithm through three distinct attention head groups: Duplicate Token Heads detect repeated names, attending from the second mention back to the first; S-Inhibition Heads suppress repeated tokens; and Name Mover Heads copy the correct IO into the output via attention.

Path Patching Path patching is an intervention-based interpretability method for circuit discovery 101 (Wang et al., 2022). It utilizes two prompt distributions: the original task distribution p_{orig} , and 102 a corrupted distribution p_{new} designed to break task-relevant behavior. First, the model is run on 103 both distributions and each computational node's activations are cached. Then, a forward pass is 104 performed on p_{orig} where the activation at a specific node F (e.g., an attention head) is replaced 105 with its counterpart from p_{new} , while the rest of the model remains unchanged. Next, the resulting activation at a downstream node G is cached and patched into a forward pass on p_{new} . The causal 107 impact of the path $F \to G$ is quantified by measuring the change in logit difference. A substantial 108 drop indicates that F is causally important for the model's behavior on the task. 109

Logit Lens The logit lens (Nostalgebraist, 2020) is an interpretability method that projects the hidden state of a computational node, h, into the model's token space. It applies layer normalization followed by the unembedding matrix:

$$LogitLens(h) = LayerNorm(h)W_U$$
.

This yields a distribution over tokens, revealing which outputs the model would favor if it predicted directly from that point.

Туре	Example Syllogism	Avg. Logit Diff.
Simple	Statement A is true. Statement B matches statement A. Statement B is true	1.8575
Opposite	Statement A and statement B are opposites. Statement A is true. Statement B is false	1.2123
Complex	Statement A is true. Statement B matches statement A. Statement C is false. Statement B is true	1.3105

Table 1: Examples of syllogism types with their corresponding average logit differences over datasets of 500 prompts. The LM is expected to predict the red tokens. We create these distributions of syllogism by replacing letters and truth values.

3 Understand How GPT-2 Process Syllogisms

Syllogisms (Aristotle, c. 350 BC) offer an effective way to analyze a LM's reasoning capacity. Rather than analyzing a broad range of facts in a syllogistic format, we narrow our focus to a simpler set of propositions and declarative statements such as: "Statement A is true. Statement B matches statement A. Statement B is true." We define three types of syllogisms: Simple Syllogism (SS), Opposite Syllogism (OS), and Complex Syllogism (CS). A complete example of each type is provided in Table 1. We define the logit difference for the syllogism task family as follows. Let the answer set be $S = \{\text{true}, \text{false}\}$ with the correct answer $x \in S$, and the incorrect answer $\neg x \in S$. The logit difference (LD) is then given by:

$$LD = logit(x) - logit(\neg x).$$

A positive logit difference indicates that the first logit is more probable, while a negative logit difference suggests the second logit is more probable. e^{LD} represents how many times more likely the model will predict x compared to $\neg x$. Thus, for the SS format, GPT-2 small is 6.4077 times more likely to predict the correct truth value.

To quantify how well a circuit preserves model behavior, we use the **faithfulness** metric. Let $ALD(\mathcal{M})$ denote the average logit difference (ALD) of the full model \mathcal{M} , and $ALD(\mathcal{C})$ that of a circuit \mathcal{C} . The faithfulness metric is defined as:

Faithfulness =
$$|ALD(\mathcal{M}) - ALD(\mathcal{C})|$$
.

A lower value indicates that the circuit faithfully recovers the model's behavior on the task.

3.1 Simple Syllogism

115

132

We frame the SS task with the following human-interpretable algorithm: (1) Identify the single truth value token in the prompt; (2) Output the truth value token. Construction of the SS dataset can be found in Appendix A.

Truth Heads We begin by applying path patching to determine which attention heads and MLPs influence the model's output logits on SS prompts. As shown in Figure 2a, MLP layers have minimal direct effect on the logits, suggesting they are not essential for solving the SS task. We explore this further in Appendix C.

In contrast, Figure 2b reveals that several attention heads in the later layers, particularly heads 7.2, 9.1, 9.9, 10.1, and 10.4, contribute substantially to logit differences. To understand the behavior of these heads, we analyze their attention patterns using their QK circuits. Specifically, for each attention head h, we compute the raw attention score:

$$A^h = t^\top W_E^\top W_{QK}^h W_E t,$$

which captures how much each query token attends to each key token in the vocabulary space. We find that these heads exhibit similar induction-like attention patterns: they predominantly attend to the final token corresponding to the truth value.

Head	1st Highest Q-K Pair	2nd Highest	3rd Highest
7.2	0.8304: ['S', 'matches']	0.5749: ['is', 'true']	0.2750: ['true', '.']
10.1	0.7139: ['is', 'true']	0.5524: ['S', 'matches']	0.4258: ['S', 'true']
10.4	0.6833: ['is', 'true']	0.6637: ['S', 'matches']	0.5063: ['.', 'Statement']

Table 2: Top 3 highest-scoring query–key token pairs from the attention pattern scores of the most influential heads in the SS format.

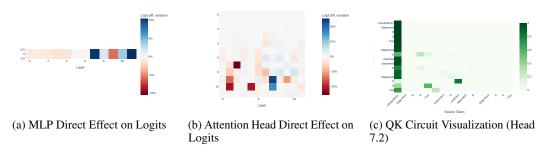


Figure 2: Path Patching and QK Analysis on Simple Syllogism Prompts.

We provide a visualization of the most influential head, 7.2, in Figure 2c, along with the top K=3 query-key token pairs in Table 2. For reference, we use the example SS prompt: "Statement E is true. Statement S matches statement E. Statement S is true". Across top heads we consistently observe two high-scoring token pairs: (S, matches) and (is, true). The first pair indicates that GPT-2 has developed a logical understanding of equivalence between the two statements—effectively computing matches(S,E)—while the second pair shows it retrieving the correct truth value based on this relationship. This consistent behavior leads us to call these attention heads $Truth\ Heads$.

To test whether Truth Heads depend on earlier attention heads, we repeat path patching on their query, key, and value inputs. We find that earlier heads have minimal effect, suggesting that the Truth Heads operate independently. To verify their sufficiency we build a minimal circuit C_{SS} consisting only of the Truth Heads. C_{SS} faithfully recovers the predictions of the model, achieving an average logit difference of 1.9286, effectively matching the performance of the GPT-2 small on the task. The Truth Heads' QK circuit consistently directs attention to the correct truth value earlier in the prompt while their OV circuit copies that value into the residual stream at the final token. Using the logit lens on truth heads confirm their output strongly favors the correct truth value.

Negative Heads in Simple Syllogism In addition to the Truth Heads, we identify a distinct group of heads—such as 9.7, 10.7, and 11.10—that exhibit attention patterns similar to the Truth Heads but are not essential for solving the task. Notably, mean-ablating head 10.7 improves model performance beyond the baseline. Head 10.7 has previously been characterized as a negative head in prior work (Wang et al., 2022) where it was shown to reduce the logit of specific output tokens. We hypothesize these negative heads encode the logit of the incorrect class in a binary setting. This aligns with findings from the copy suppression literature (McDougall et al., 2023) where head 10.7 was also found to suppress certain tokens. To test this, we turn to the Opposite Syllogism format.

3.2 Opposite Syllogism

To test our hypothesis surrounding negative heads, we investigate the model's behavior on *opposite* syllogisms (OS). We define a human-interpretable algorithm for this task in three steps: (1) Identify the single truth value token in the prompt, (2) Negate the truth value token, and (3) Output the negated

Top QK Pairs (Head 10.7)

190

191

192

193

194

195

197

198

199

200 201

202

207

0.892: ('is', 'true'), **0.772:** ('statement', 'E'), **0.685:** ('Statement', 'S'), **0.662:** ('Statement', 'S'), **0.459:** ('is', 'oppos')

Stage	Top Logits	Bottom Logits
After OV from Head 10.7	depot, rink, carp, Dj, Hack, DJ, Gaz, Phillips, District, TTC	'true', 'True', 'TRUE', 'true', 'untrue', 'Null'
After MLP Layer 10	'true', 'false', 'True', 'False', infinite, truly	blitz, ombo, plateau, corrid, tradem, emale, Citiz, sugg

Table 3: Top QK pairs in Head 10.7 strongly attend to truth-related tokens (e.g., 'is', 'true'). Initially, the OV output does not rank truth tokens highly. However, after the MLP layer, both 'true' and 'false' become top-ranked, indicating the MLP can help produce the opposite token, even when it is not in the prompt

token. Details on dataset construction are provided in Appendix A. For reference, we use the example 175 OS prompt: "Statement E and statement S are opposites. Statement E is true. Statement S is false".

Negative Truth Heads. We begin by identifying components that directly influence the model's 176 output on OS prompts. Path patching shows that ablating attention heads 7.3, 8.8, 8.10, 9.7, and 177 10.7 leads to a significant drop in logit difference (Figure 3b). To understand their role, we analyze 178 each head's output by applying its OV matrix to the MLP-extended embedding basis, following prior 179 techniques from Wang et al. (2022) and McDougall et al. (2023): 180

$$W_U W_{OV}^h \operatorname{MLP}_0(W_E)$$
.

These heads consistently attend to the truth value token in the prompt (e.g., is, true)—mirroring 181 behavior observed in the SS setting. However, their influence on the logits differs. Some heads, 182 like 8.8 and 5.1, promote the truth token to the top logits and function as standard *Truth Heads*. In 183 contrast, heads such as 7.3, 8.10, 9.7, 10.7, and 11.10 suppress the truth token into the bottom logits 184 (Table 3). We refer to this group as *Negative Truth Heads*. 185

Unlike the SS format, MLPs corresponding to these heads are crucial: ablating them significantly 186 reduces performance (Figure 3a). Furthermore, path patching the queries of these Negative Truth 187 Heads confirms that they operate independently, with no significant upstream influence, mirroring 188 earlier findings from SS where only components with direct logit impact matter. 189

Mechanistic Interpretation The QK circuit of each Negative Truth Head reliably identifies the truth token—fulfilling Step 1 of the OS algorithm. However, their OV projection suppresses this token into the bottom logits. The associated MLP then rescales the residual stream to promote the opposite of the suppressed token into the top logits, completing Steps 2 and 3. We refer to these components as Truth Logit Rescaler MLPs.

This attention–MLP sequence forms a mechanistic pathway for learned negation: the model suppresses a truth token and then elevates its negation for output. Table 3 captures the entire trajectory 196 of such a token: from attention-induced suppression to MLP-driven recovery. This reveals how the model predicts a correct token not seen in the prompt, using the suppression of the incorrect token as a signal for its opposite. While this attention-MLP mechanism effectively negates the truth value in opposite syllogisms, we observe a consistent asymmetry: the negation process is more reliably triggered when the input token is true, resulting in false predictions. In contrast, when the prompt contains false, the model often retains false as the dominant logit rather than flipping to true.

Circuit Faithfulness To test sufficiency, we construct a circuit C_{OS} using only the Negative Truth 203 Heads and their associated MLPs. This circuit recovers approximately 85\% of GPT-2 small's 204 performance, demonstrating it is a faithful subcircuit for solving the OS task. A schematic of this 205 circuit is shown in Figure 1. 206

Reversal of Head Behavior Between Tasks Interestingly, the same heads that negatively affected logit difference in the SS task—like 10.7—now play a constructive role in OS. This reversal demon-

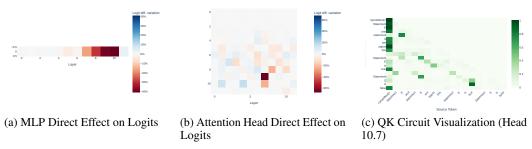


Figure 3: Path Patching and QK Analysis on Opposite Syllogism Prompts.

strates that the model reuses certain components in complementary tasks where their function flips to support inverse outcomes. We extend this finding in Appendix B, where we test circuit generalization and transferability across other binary pairs beyond true/false, further supporting the idea that GPT-2 small represents logical negation via attention and MLPs.

3.3 Complex Syllogism

213

231 232

234

235

236

237

238

239

240

The complex syllogism task expands on the previous setups by introducing a misleading, redundant statement. The objective is to determine whether GPT-2 Small can still arrive at the correct conclusion in the presence of potentially confusing information.

We begin with path patching to identify which attention heads and MLPs directly influence the model's 217 output logits. The results are similar to the path patching results of the OS format. More specifically, 218 MLPs in layers 8,9, and 10 positively influence the logit difference. Similar heads—specifically, 219 heads 7.3, 8.8, 8.10, 9.7, and 10.7— were found to positively influence logit difference but with different behavior. Heads 9.7, 10.7, and 11.10 attend to both truth tokens in the prompt but place greater emphasis on the incorrect (redundant) truth token. Inspecting their logits, we find these heads 222 continue to perform a suppression operation as observed previously, and we therefore classify them 223 as Negative Truth Heads in the CS format and do not investigate them further. In contrast, heads 224 7.3, 8.8, and 8.10 exclusively attend to the incorrect truth value, and we study them more closely to 225 understand their contribution. We refer to these as Truth Modulation Heads. 226

To interpret the behavior of the Truth Modulation Heads, we analyze how their outputs project onto the unembedding vectors of the truth tokens. Let $W_U[\mathtt{true}]$ and $W_U[\mathtt{false}]$ represent the unembedding directions for the correct and incorrect truth tokens respectively. For a head output $h_i(X)$, we compute the logit contribution to token $t \in \{\mathtt{true},\mathtt{false}\}$ as

$$\langle h_i(X), W_U[t] \rangle$$
.

This dot product reflects how strongly head h_i pushes the residual stream toward generating token t. We scatter plot each head's attention probability on the truth value token against the corresponding logit contribution along the true direction. Two distinct patterns emerge across the heads. First, the correct token is consistently ignored by these heads, receiving very low attention probability. Second, the incorrect token shows two opposing trends depending on the head: in some cases, the attention probability and logit contribution are positively correlated, suggesting that the head reinforces the incorrect truth value. We refer to such heads as $Correct\ Truth\ Inhibition\ Heads$. In other cases, the relationship is negative—the more attention the head gives to the incorrect token, the more it pushes away from the incorrect truth direction. This effectively reinforces the correct token and we refer to these as $Correct\ Truth\ Reinforcement\ Heads$.

These behaviors are further supported by examining the top and bottom logits. In inhibition heads, the incorrect token consistently appears among the top logits while in reinforcement heads it appears among the bottom logits. This supports the interpretation that Truth Modulation Heads implement a binary operation: either reinforcing or inhibiting the direction of the incorrect token, which indirectly determines the correct output.

We then investigate whether the Negative Truth Heads and Truth Modulation Heads influence one another. Path patching reveals that neither group affects the other directly, although both are

influenced by similar upstream sources. Moreover, ablating one group does not destroy faithfulness, confirming that the groups can operate independently. This redundancy is consistent with the findings of McGrath et al. (2023), who describe the Hydra Effect in language models, where multiple pathways can implement the same behavior.

Path patching reveals that both the Negative Truth Heads and the Truth Modulation Heads receive input from a shared set of upstream heads, primarily located in Layers 0 and 5. In Layer 5, heads 5.1 and 5.5 exhibit classic induction patterns as described by Elhage et al. (2021), attending from the conclusion line (e.g., "Statement B is") back to a matching premise. This effectively links the conclusion to its logical source. In Layer 0, other heads attend to repeated statement identifiers (e.g., "Statement B") and influence both the key vectors of downstream heads and the values used by the Induction Heads. These heads appear to detect repeated statements and function as Duplicate Statement Identifier Heads, marking the reuse of information—an essential step in a natural deduction process.

At inference time, the Induction Heads serve as a routing mechanism: they either connect the conclusion to the premise via the "matches" relation or directly extract the incorrect truth value from the conclusion line. This information is then processed by the Negative Truth Heads or Truth Modulation Heads to generate the final token.

265 4 Discussion

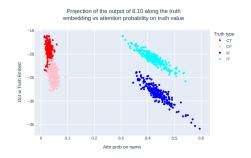
Connection to IOI Despite overlap with the Name Mover Heads from the IOI task, we find (Negative) Truth Heads reflect broader functionality, particularly negate rather than simply copy. From an IOI perspective, the Negative Truth Heads were initially interpreted as negative copy heads due to their tendency to replicate the tokens they attend to. However, in the opposite syllogism task, the correct answer is not explicitly present in the prompt. Consequently, these heads cannot simply copy the attended truth value to produce the correct answer. This provides strong evidence that Negative Truth Heads encode the direction of the less contextualized logit in a binary setting, effectively operating in the antidirection. We believe this behavior remained unnoticed in IOI because, in that context, Mary $\neq \neg$ John. Similarly, many of the Truth Modulation Heads align with the S-Inhibition category from IOI, suggesting a shared functional role. We identify the Correct Truth Inhibition Heads as the original inhibition heads from IOI, given their role in reinforcing focus on the incorrect token. This expanded understanding highlights how heads previously characterized in IOI tasks can exhibit more nuanced and adaptable behaviors in different contexts.

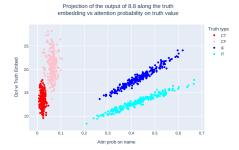
Clustering of Truth Modulation Heads We observe distinct clusters in both groups of Modulation Heads. To refine our truth types, we categorize truth values into four types: correct true (CT), correct false (CF), incorrect true (IT), and incorrect false (IF). This results in two natural pairings: (CT, IF) and (CF, IT). As shown in Figure 4a and 4b, false (IF or CF) has larger projections on the truth embedding. We believe that this asymmetry not only reflects the internal bias of truth values learned from the training corpus, but also resembles the behavior observed in the Opposite Syllogism task, where negation was easier with true. Although we do not rigorously analyze this connection, it may reflect a broader model bias toward negative truth values or a negation-like structure in its internal representations.

Scalability of Results We extend our simple and opposite syllogism formats to larger models such as GPT-2 XL, Pythia 1.4B, Qwen3-1.7B, and LLaMA3.2-1B. (See Appendix D).

5 Related Works

Mechanistic Interpretability Mechanistic Interpretability research offers various techniques to reverse-engineer model behavior and identify important components responsible for a model's performance. In addition to Path Patching (Wang et al., 2022), there are other patching methods including Attribution Patching (Nanda, 2023), causal mediation analysis (Meng et al., 2023; Pearl, 2022; Vig et al., 2020), and AtP* (Kramár et al., 2024). Sparse Autoencoders (SAEs) have become increasingly popular for interpreting features (Bricken et al., 2023; Marks et al., 2025). Earlier works such as Neuron Shapley (Ghorbani & Zou, 2020) introduce a framework that quantifies each neuron's contribution to a deep network's performance by considering interactions among neurons. Other





(a) Correct Truth Reinforcement Head 8.10 Projection along truth embedding versus attention probability on truth value

(b) Correct Truth Inhibition Head 8.8 Projection along truth embedding versus attention probability on truth value

Figure 4: The Truth Modulation group with refined truth types

earlier works such as Cao et al. (2021); Csordás et al. (2021) employ subnetworks to investigate what model internals are needed to perform a task through probing and masking.

Circuit Discovery in GPT-2 IOI has inspired many other circuit analysis works. Hanna et al. (2023) identify a circuit that explains GPT2's ability to predict correct year tokens when prompted with task like "The war lasted from the year 1732 to 17". Merullo et al. (2024) rediscover the IOI circuit in GPT2-Medium and show that much of the circuit can be reused to solve the Colored Object task introduced by Srivastava et al. (2023). Nainani et al. (2024) explore IOI's generality by extending the prompt to include more instances of the indirect object. Conmy et al. (2023) generalize and automate the IOI-style analysis within GPT-2 small, ultimately recovering many already discovered circuits.

Syllogisms for Assessing LLMs Recent studies have explored assessing LLMs with syllogisms. Eisape et al. (2024); Ando et al. (2023) provide a comparative analysis on how humans and LLMs perform syllogistic reasoning. Kim et al. (2025) conduct a mechanistic analysis of standard syllogisms. In contrast, our work explores syllogisms with assigned truth values, offering a distinct perspective. Furthermore, we provide novel insights into the role of MLPs in facilitating syllogistic reasoning and handling logical negation.

314 6 Conclusion

300

301

302

303

304

305

306

307

In this work, we reverse-engineered GPT-2 for three syllogism tasks of varying complexity, uncovering 315 key insights into how GPT-2 handles binary truth values within logical tasks. In the simplest case, 316 high faithfulness was achieved with just Truth Heads, highlighting the model's ability to maintain 317 correct truth values with minimal components. In the opposite syllogism case, the inclusion of Negative Truth Heads and MLPs allowed the model to properly negate the truth value, demonstrating the novel negation mechanism in handling binary outcomes. In the complex case, while negation 320 remained a key mechanism, additional heads were needed to identify and process the correct truth 321 value to negate, reflecting the increased complexity of the task. Our findings reveal significant overlap 322 with the IOI circuit, expanding our understanding of these computational nodes' capabilities; however, 323 this understanding remains limited, underscoring the need for continued interpretability research to 324 ensure that, as such models become increasingly integrated into daily life, their logical deductions 325 326 can be relied upon and their behaviour held accountable.

327 Acknowledgments

328

329

330

We thank the broader mechanistic interpretability community for open-sourcing tools and models that made this work possible. In particular, our experiments and analyses are built upon TransformerLens Nanda & Bloom (2022), which provided essential infrastructure for probing and visualizing transformer internals. This research was not supported by any external funding.

References

- Risako Ando, Takanobu Morishita, Hirohiko Abe, Koji Mineshima, and Mitsuhiro Okada. Evaluating large language models with neubaroco: Syllogistic reasoning ability and human-like biases, 2023. URL https://arxiv.org/abs/2306.12567.
- 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, 2024. URL https:
 //arxiv.org/abs/2406.11717.
- Aristotle. *Prior Analytics*. Oxford University Press, Oxford, UK, c. 350 BC.
- Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.
- Steven Cao, Victor Sanh, and Alexander M. Rush. Low-complexity probing via finding subnetworks, 2021. URL https://arxiv.org/abs/2104.03514.
- Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià GarrigaAlonso. Towards automated circuit discovery for mechanistic interpretability, 2023. URL https://arxiv.org/abs/2304.14997.
- Róbert Csordás, Sjoerd van Steenkiste, and Jürgen Schmidhuber. Are neural nets modular? inspecting functional modularity through differentiable weight masks, 2021. URL https://arxiv.org/abs/2010.02066.
- Tiwalayo Eisape, MH Tessler, Ishita Dasgupta, Fei Sha, Sjoerd van Steenkiste, and Tal Linzen. A systematic comparison of syllogistic reasoning in humans and language models, 2024. URL https://arxiv.org/abs/2311.00445.
- Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.
- Amirata Ghorbani and James Zou. Neuron shapley: Discovering the responsible neurons, 2020. URL https://arxiv.org/abs/2002.09815.
- Wes Gurnee, Theo Horsley, Zifan Carl Guo, Tara Rezaei Kheirkhah, Qinyi Sun, Will Hathaway,
 Neel Nanda, and Dimitris Bertsimas. Universal neurons in gpt2 language models, 2024. URL
 https://arxiv.org/abs/2401.12181.
- Michael Hanna, Ollie Liu, and Alexandre Variengien. How does gpt-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model, 2023. URL https://arxiv.org/abs/2305.00586.
- Geonhee Kim, Marco Valentino, and André Freitas. A mechanistic interpretation of syllogistic reasoning in auto-regressive language models, 2025. URL https://arxiv.org/abs/2408. 08590.
- János Kramár, Tom Lieberum, Rohin Shah, and Neel Nanda. Atp*: An efficient and scalable method for localizing llm behaviour to components, 2024. URL https://arxiv.org/abs/2403.00745.
- Samuel Marks, Can Rager, Eric J. Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller.
 Sparse feature circuits: Discovering and editing interpretable causal graphs in language models,
 2025. URL https://arxiv.org/abs/2403.19647.

```
Callum McDougall, Arthur Conmy, Cody Rushing, Thomas McGrath, and Neel Nanda. Copy
380
      suppression: Comprehensively understanding an attention head, 2023. URL https://arxiv.
381
      org/abs/2310.04625.
382
```

- Thomas McGrath, Matthew Rahtz, Janos Kramar, Vladimir Mikulik, and Shane Legg. The hydra 383 effect: Emergent self-repair in language model computations, 2023. URL https://arxiv.org/ 384 abs/2307.15771. 385
- Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual 386 387 associations in gpt, 2023. URL https://arxiv.org/abs/2202.05262.
- 388 Jack Merullo, Carsten Eickhoff, and Ellie Pavlick. Circuit component reuse across tasks in transformer language models, 2024. URL https://arxiv.org/abs/2310.08744. 389
- Jatin Nainani, Sankaran Vaidyanathan, AJ Yeung, Kartik Gupta, and David Jensen. Adaptive circuit 390 behavior and generalization in mechanistic interpretability, 2024. URL https://arxiv.org/ 391 abs/2411.16105. 392
- Neel Nanda. Attribution patching: Activation patching at industrial scale, 2023. URL https: 393 //www.neelnanda.io/mechanistic-interpretability/attribution-patching. 394
- Neel Nanda and Joseph Bloom. Transformerlens. https://github.com/TransformerLensOrg/ 395 TransformerLens, 2022. 396
- Nostalgebraist. Interpreting gpt: The logit lens, 2020. URL https://www.lesswrong.com/ 397 posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens. 398
- Judea Pearl. Direct and Indirect Effects, pp. 373–392. Association for Computing Machinery, New 399 York, NY, USA, 1 edition, 2022. ISBN 9781450395861. URL https://doi.org/10.1145/ 400 3501714.3501736. 401

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam 402 Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, 403 Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. 404 Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda 405 Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders An-406 dreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew Dai, Andrew La, 407 Andrew Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna 408 Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, 409 Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut 410 Erdem, Ayla Karakaş, B. Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, 411 Batuhan Ozyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk 412 Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinion, Cameron Diao, Cameron Dour, Cather-413 ine Stinson, Cedrick Argueta, César Ferri Ramírez, Chandan Singh, Charles Rathkopf, Chenlin 414 Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Chris Waites, Christian Voigt, Christo-415 pher D. Manning, Christopher Potts, Cindy Ramirez, Clara E. Rivera, Clemencia Siro, Colin Raffel, 416 Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle 418 Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David 419 Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz 420 Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho 421 Mollo, Diyi Yang, Dong-Ho Lee, Dylan Schrader, Ekaterina Shutova, Ekin Dogus Cubuk, Elad 422 Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodola, 423 Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A. Chi, Ethan Dyer, Ethan 424 Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, 425 Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, 426 Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio 427 Mariani, Gloria Wang, Gonzalo Jaimovitch-López, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, 428 Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Shevlin, 429 Hinrich Schütze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap 430 Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, 431 James B. Simon, James Koppel, James Zheng, James Zou, Jan Kocoń, Jana Thompson, Janelle

432

Wingfield, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason 433 Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse 434 Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, 435 John Miller, John U. Balis, Jonathan Batchelder, Jonathan Berant, Jörg Frohberg, Jos Rozen, 436 Jose Hernandez-Orallo, Joseph Boudeman, Joseph Guerr, Joseph Jones, Joshua B. Tenenbaum, 437 Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakr-438 ishnan, Katerina Ignatyeva, Katja Markert, Kaustubh D. Dhole, Kevin Gimpel, Kevin Omondi, 439 Kory Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle 440 Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-441 Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, 442 Luheng He, Luis Oliveros Colón, Luke Metz, Lütfi Kerem Şenel, Maarten Bosma, Maarten Sap, 443 Maartje ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramírez Quintana, Marie Tolkiehn, Mario Giulianelli, Martha 445 Lewis, Martin Potthast, Matthew L. Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael A. Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michael Swedrowski, Michael Bevilacqua, 448 Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, 449 Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, 450 Nathan A. Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick 451 Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish 452 Keskar, Niveditha S. Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, 453 Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale 454 Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter Chang, 455 Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, 456 Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer 457 Gabriel, Rahel Habacker, Ramon Risco, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A. 458 Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman 459 Sitelew, Ronan LeBras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Ruslan Salakhutdinov, Ryan 460 Chi, Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M. Mohammad, Sa-461 jant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R. Bowman, 462 Samuel S. Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan 463 Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, 464 Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, 465 Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima, 466 Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, 467 Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano 468 Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven T. Piantadosi, Stuart M. Shieber, 469 470 Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsu Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas 471 Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Korney, Titus Tunduny, Tobias Ger-472 stenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, 473 Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Ramasesh, Vinay Uday Prabhu, Vishakh 474 Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, 475 Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair 476 Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang, Yiding Hao, Yifu Chen, Yonatan 477 Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J. 478 Wang, Zirui Wang, and Ziyi Wu. Beyond the imitation game: Quantifying and extrapolating the 479 capabilities of language models, 2023. URL https://arxiv.org/abs/2206.04615. 480

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and
Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), Advances
in Neural Information Processing Systems, volume 33, pp. 12388–12401. Curran Associates,
Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/
92650b2e92217715fe312e6fa7b90d82-Paper.pdf.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small, 2022. URL https://arxiv.org/abs/2211.00593.

490 A Syllogism Dataset Construction

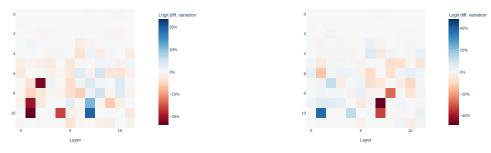
Syllogistic prompts were created using templates. [TRUTH_VALUE] $\in \{ \text{true}, \text{false} \}$, and [A], [B], [C] are sampled from capital letters. See table 4 for templates of each syllogism format

Type	Template		
Simple Syllogism	1. Statement [A] is [TRUTH_VALUE_1]. Statement [B] has the same truth value as [A]. Statement [B] is [TRUTH_VALUE_1].		
	2. Statement [A] is [TRUTH_VALUE_1]. Statement [B] matches statement A. Statement B is [TRUTH_VALUE_1].		
	3. (Extended) Statement [A] is [TRUTH_VALUE_1]. Statement [B] must match [A]. Statement [C] doesn't matter. Statement [B] is [TRUTH_VALUE_1].		
Opposite Syllogism	1. Statement [B] has the opposite truth value of [A]. Statement [A] is [TRUTH_VALUE_1]. Statement [B] is [TRUTH_VALUE_2].		
	2. Statement [A] and statement [B] are opposites. Statement [A] is [TRUTH_VALUE_1]. Statement [B] is [TRUTH_VALUE_2].		
Complex Syllogism	1. Statement [A] is [TRUTH_VALUE_1]. Statement [B] has same truth value as [A]. Statement [C] is [TRUTH_VALUE_2]. Statement [B] is [TRUTH_VALUE_3].		
	$(Harder\ constraint):\ [TRUTH_VALUE_2] = \neg [TRUTH_VALUE_1].$		
Complex Opposite Syllogism	1. Statement [A] is [TRUTH_VALUE_1]. Statement [B] has the opposite truth value of [A]. Statement [C] is [TRUTH_VALUE_2]. Statement [B] is [TRUTH_VALUE_3].		
	2. Statement [A] and [B] are opposites. Statement [C] has the same truth value as [A]. Statement [B] is [TRUTH_VALUE_3].		
	3. Statement [A] is [TRUTH_VALUE_1]. Statement [A] and [B] are opposites. Statement [C] is [TRUTH_VALUE_2]. Statement [B] is [TRUTH_VALUE_3].		

Table 4: Templates used for generating syllogistic prompts.

B Generality Across Binary Contrasts

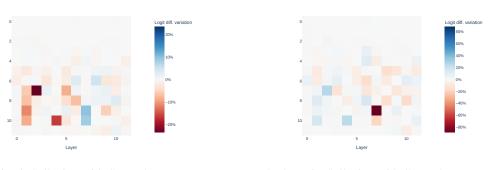
- Having established mechanistic evidence for circuits supporting binary truth tasks in both the simple and opposite syllogism settings, we next evaluate the generality of these circuits beyond the original true/false framing. Specifically, we test whether the same circuits generalize to anal-
- ogous binary pairs: (right, wrong), (good, bad), (positive, negative), and (correct,
- 498 incorrect).
- To do so, we apply both the simple syllogism circuit (C_{SS}) and opposite syllogism circuit (C_{OS}) to
- each pair and compare their performance to the full GPT-2 Small model. As shown by tables 5 and 6
- we find that the original circuits often match or even outperform the full model in logit difference
- between most binary pairs of tokens. This provides compelling evidence that the binary task is not
- specific to a particular token pair, but instead reflects a transferable reasoning mechanism.
- To further validate generalization, we visualize direct path patching attention results across each
- 505 binary pair. As seen in Figures 5–8, across the binary pairs of tokens, the core attention heads relevant
- to the simple and opposite syllogism cases are opposite in their effect on logit difference.



(a) Simple Syllogism with Right/Wrong

(b) Opposite Syllogism with Right/Wrong

Figure 5: Binary task results of Right/Wrong



(a) Simple Syllogism with Correct/Incorrect

(b) Opposite Syllogism with Correct/Incorrect

Figure 6: Binary task results of Correct/Incorrect



(a) Simple Syllogism with Positive/Negative

(b) Opposite Syllogism with Positive/Negative

Figure 7: Binary task results of Positive/Negative

	Original	Good/Bad	Pos/Neg	Correct/Incorrect	Right/Wrong
GPT-2 Small	1.8399	1.7738	0.6958	2.1221	2.0309
$oldsymbol{C}_{SS}$	1.9234	1.9940	1.1584	1.6785	2.1599

Table 5: Transferability of C_{SS} to other binary token pairs

	Original	Good/Bad	Pos/Neg	Correct/Incorrect	Right/Wrong
GPT-2 Small	1.2632	2.1163	3.0032	0.7986	1.3469
$oldsymbol{C}_{OS}$	1.3136	1.7136	1.0113	0.8142	1.2481

Table 6: Transferability of C_{OS} to other binary token pairs

C Disentangling MLP Contributions via Patching

To assess the contribution of MLPs to the model's output, we perform path patching both with and without attention restored. Figure 10b shows that early-layer MLPs—particularly MLP0—appear to significantly affect the logits when patched in isolation. This aligns with prior observations that MLP0 functions as an extended embedding layer, especially when attention is absent (McDougall et al., 2023; Wang et al., 2022).

However, once attention is also restored, the influence of these early MLPs sharply diminishes. This suggests their apparent impact in the no-attention condition is largely an artifact of missing context, rather than a reflection of GPT2 semantic ability to complete syllogisms.

For this reason, in all subsequent experiments analyzing MLP effects, we report results with attention paths patched in. This allows us to isolate the true downstream influence of MLPs under more realistic model conditions.

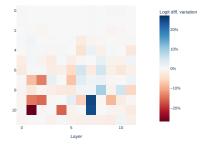
D Extension to Larger Models

507

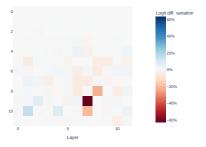
519

To assess whether the findings observed in GPT-2 Small generalize across model scale and architecture, we extend our experiments to several larger models: GPT-2 XL, Pythia 1.4B, Qwen3-1.7B, and LLaMA3.2-1B.

Across all models, we continue to observe empirical signatures of binary behavior: heads relevant to the simple and opposite syllogism tasks tend to exert opposing effects on the logits. MLP layers remain important in the opposite syllogism task for all models except Pythia 1.4B, mirroring the behavior observed in GPT-2 Small. Notably, Table 7 shows that performance on the simple syllogism



(a) Simple Syllogism with Good/Bad



(b) Opposite Syllogism with Good/Bad

Figure 8: Binary task results of Good/Bad



(a) MLP effects with attention context

(b) MLP effects without attention context

Figure 9: Path patching MLPs in the opposite syllogism task. (a) shows effects when MLPs are patched with attention context preserved; (b) shows isolated MLP contributions without attention context.

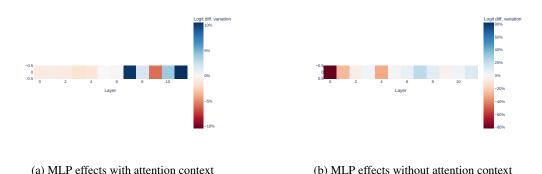


Figure 10: Path patching MLPs in the simple syllogism task. (a) shows effects when MLPs are patched with attention context preserved. No MLPs have significant importance; (b) shows isolated MLP contributions without attention context. Early MLPs, specifically MLP0, appear relevant for the task

format degrades significantly in larger models, suggesting that task generalization does not uniformly scale with model size.

All models retain some attention heads exhibiting negative-copy behavior. However, the influence of these heads on output logits is more muted compared to GPT-2 Small. In particular, the heads most responsible for enabling opposite syllogism performance in the larger models are not the negative heads. Qwen3-1.7B, for instance, contains relatively few negative heads, and those it has do not drive logit differences in either task. An exception is Pythia 1.4B, whose success on the opposite task remains closely tied to the activity of its negative-copy heads.

Interestingly, across all models, the heads most influential on model output tend to exhibit strong induction behavior (e.g., $ABA \rightarrow B$), regardless of whether they also contribute to the task-relevant distinction. Yet despite this variability in attention head dynamics, the consistent involvement of MLPs in the opposite task—and their near absence in the simple task—suggests a robust division of labor: negation appears to depend more heavily on the feedforward path than on attention alone. This may help constrain future hypotheses about the mechanistic implementation of logical inversion and contextual negation in transformer models.

These findings remain empirical and exploratory. Figures 11-14 illustrate the direct effects of attention heads and MLPs across the syllogism tasks. A deeper investigation into how architectural scale affects circuit behavior remains a promising direction for follow-up work.

	GPT-2 XL	Qwen3-1.7B	LLaMA 3.2-1B	Pythia 1.4B
Simple Syllogism	0.1112	0.5322	-0.4357	1.0105
Opposite Syllogism	2.6114	1.5257	-0.1807	2.1098

Table 7: Average logit difference across models and tasks.

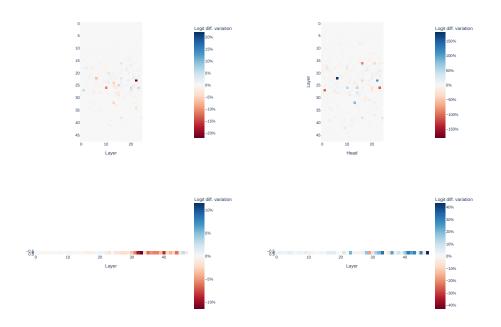


Figure 11: Direct effects of attention heads and MLPs for GPT-2 XL across syllogism tasks.

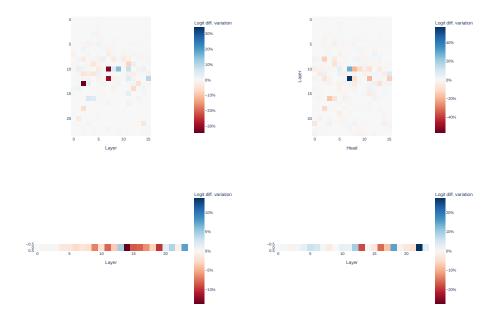


Figure 12: Direct effects of attention heads and MLPs for Pythia 1.4B across syllogism tasks.

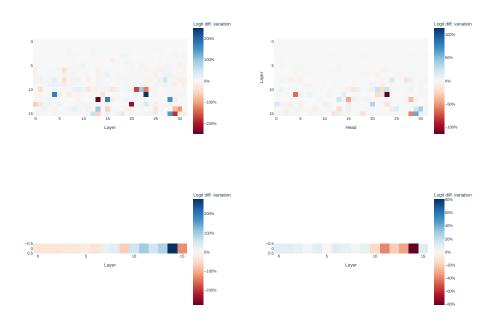


Figure 13: Direct effects of attention heads and MLPs for LLaMA 3.2B across syllogism tasks.

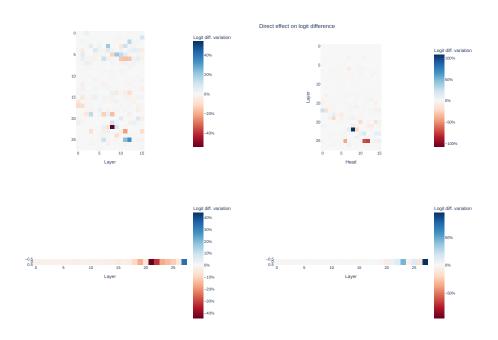


Figure 14: Direct effects of attention heads and MLPs for Qwen 1.7B across syllogism tasks.