Punctuations and Predicates in Language Models

Sonakshi Chauhan[1]*, **Maheep Chaudhary**[2], **Koby Choy**[2],
Samuel Nellessen[3], **Nandi Schoots**[4]
[1]University of Glasgow [2]Independent
[3]Radboud University Nijmegen [4]University of Oxford

Abstract

In this paper we explore where information is collected and how it is propagated throughout layers in large language models (LLMs). We begin by examining the surprising computational importance of punctuation tokens which previous work has identified as attention sinks and memory aids. Using intervention-based techniques, we evaluate the necessity and sufficiency (for preserving model performance) of punctuation tokens across layers in GPT-2, DeepSeek, and Gemma. Our results show stark model-specific differences: for GPT-2, punctuation is both necessary and sufficient in multiple layers, while this holds far less in DeepSeek and not at all in Gemma. Extending beyond punctuation, we ask whether LLMs process different components of input (e.g., subjects, adjectives, punctuation, full sentences) by forming early static summaries reused across the network, or if the model remains sensitive to changes in these components across layers. Extending beyond punctuation, we investigate whether different reasoning rules are processed differently by LLMs. In particular, through interchange intervention and layer-swapping experiments, we find that conditional statements (if, then), and universal quantification (for all) are processed very differently. Our findings offer new insight into the internal mechanisms of punctuation usage and reasoning in LLMs and have implications for interpretability.

1 Introduction

Recent work has revealed that LLMs often perform tasks in ways that diverge significantly from human reasoning. In particular, punctuation which is often considered negligible in human processing appears to play a surprisingly active and complex role within these models. Studies have shown that punctuation tokens can act as attention sinks (Gu et al., 2025; Barbero et al., 2025), increase computational memory (Pfau et al., 2024), and even serve as information carriers (Tigges et al., 2023). Despite their seemingly minor status, punctuation marks appear to shoulder a disproportionately large computational burden.

In this work, we examine the role of punctuation in summarization and information propagation through the lens of necessity and sufficiency: how essential are these tokens for preserving meaning and reasoning within the model? We use zeroing out interventions to better understand how punctuation influences layerwise computation dynamics. To test whether the period acts as a boundary for information summarization in the context of reasoning, we also perform targeted activation interchange interventions.

Building on this, we take a mechanistic lens to ask a broader question: How do LLMs internally represent and process reasoning? Previous work on reasoning has investigated reasoning rules using graph-based approaches (Luo et al., 2024; Qu et al., 2021; Tang et al., 2024). In this paper, we investigate whether models interpret inputs compositionally, treating different elements of a sentence (such as subjects, adjectives, punctuation, and full syntactic structures) as functionally distinct units or whether they instead form a static summary early in the forward pass and reuse that across layers. This line of inquiry helps us understand not just whether a model can reason, but how that reasoning unfolds internally. This distinction is an effort to zoom-in on how closely model behaviour aligns or diverges from human reasoning patterns.

To understand how rules are processed by language models, we analyze which layers are responsible for reasoning over different types of logical and syntactic structures. Specifically, we aim to identify whether rule-consistent behavior is distributed across the model or concentrated in particular layers. We use part-of-sentence interventions and layer swaps, with the aim of characterizing how different layers contribute across syntactic and

*Correspondence to: sonakshichauhan1402@gmail.com

Findings of the Association for Computational Linguistics: EACL 2026, pages 5622–5636

logical dimensions.

Our main findings are:

- For GPT-2 we find that period and question mark tokens are necessary and sufficient in five out of twelve layers. However, in DeepSeek this holds for one in twentyfour layers, and in Gemma there is no layer for which this holds. See Section 5.1. We are the first to investigate sufficiency and necessity of punctuation tokens, which we do by selectively (non-)zeroing out tokens. Although interchange interventions have been done on tokens, they were not previously used to investigate attention sinks and memorization.

- We investigate which layers are swappable, and find different LLMs behave differently. For GPT-2 and Gemma we find that middle layers are unique, for DeepSeek we find that layers become more interchangeable, when prompted on statements containing reasoning rules. See Figure 5. We are the first to compare reasoning rules using layer swaps.

Across models there is a wide variation in necessity and sufficiency of punctuation, sensitivity to interventions on different reasoning rules, and non-uniform swappability of layers. This reveals sharp differences in how internal computations are organized across architectures or training paradigms, and suggests distinct underlying inductive biases in how models learn to represent and perform reasoning.

2 Related Work

Mechanistic interpretability seeks to reverse-engineer neural networks by linking neural components to specific behaviors and computations (Olah et al., 2017). In causal abstraction this linking is done by forming a causal model of the behaviors (Geiger et al., 2021). The following lines of research are most related to our work.

Layerwise Process Analyses. Previous work has shown earlier layers process initial structure of the input (Yang et al., 2025), and ablating them leads to substantial degradation (Zhang et al., 2024). Middle layers have been found to build intermediate representations (Yang et al., 2025), to be essential for a variety of reasoning tasks (Skean et al., 2025; Liu and Niehues, 2025), and they often operate in a shared representational space, swapping them has little effect for general tasks but this does not hold true for structured domains like math and logical reasoning (Sun et al., 2025). Later layers focus on task-specific specialization (Yang et al., 2025), contain attention heads specialized for generalization (Ye et al., 2025), and may primarily serve to consolidate and finalize already-inferred information (Ben-Artzy and Schwartz, 2024).

Attention Sinks emerge naturally during pretraining giving disproportionate attention to the first token (Gu et al., 2025). Such attention sinks may be functionally necessary to prevent representational collapse and ensure smooth information flow in long-context scenarios (Barbero et al., 2025). Tigges et al. (2023) demonstrate that punctuation symbols act as aggregation points for sentiment signals within the model's activation space, consistent with more recent mechanistic evidence that language models internally compress and reuse information through structured intermediate representations (Mishra, 2025). Punctuation tokens serve as structural anchors, influencing factual consistency, context retention, and reasoning (Razzhigaev et al., 2025; Zhu et al., 2025). Punctuation may also act as computation enablers, providing additional "thinking steps" irrespective of their linguistic function (Pfau et al., 2024).

Reasoning. Techniques like steering vectors (Venhoff et al., 2025), chain-of-thought prompting (Plaat et al., 2024; Dutta et al., 2024) and multi-hop reasoning analysis (Wang et al., 2024) aim to make the implicit reasoning steps of LLMs more interpretable and controllable. LLMs may construct internal reasoning trees, recoverable via attention-based probes (Hou et al., 2023), indicating genuine multi-hop reasoning beyond memorization. Benchmarks like Cladder (Jin et al., 2024) systematically evaluate a model's ability to extract and utilize causal structures from text. Bogdan et al. (2025) perform sentence-level reasoning analysis using thought vectors, revealing that a small subset of these vectors disproportionately governs the model's reasoning process, consistent with other findings that specific tokens or structures can exert outsized influence on model predictions (Servantez et al., 2024). Recent work has further identified causally necessary internal structures underlying logical reasoning in language models, demonstrating that specific components are responsible for syllogistic inference (Kim et al., 2025). Ibeling and Icard (2021); Chauhan and Geiger (2024) have

explored reasoning from a causal perspective. Prior work has also shown that strong performance on reasoning benchmarks does not necessarily reflect robust rule-based reasoning, motivating analyses that probe how such behavior is internally realized (Talmor et al., 2020).

3 Models and Datasets

Models: We conduct our experiments using three models of varying sizes and architectures: GPT-2 Small (Lee and Hsiang, 2019), Gemma (Team et al., 2024), and DeepSeek (Guo et al., 2024). A clear distinction in the characteristics of the models can be seen in Table A in the appendix.

We use these models to answer questions based on the context provided; each question can have three possible answers 1. *True*, 2. *False*, 3. *Unknown*. Our main aim of having an experimental setup like this is to check how the model processes context and then check the understanding of the context with the help of questions.

RuleTaker Dataset (Clark et al., 2020) tests the reasoning and implication abilities of LLMs. It includes facts and rules, followed by questions that assess whether the rules are correctly applied. Answers to these questions are labeled as True, False, or Unknown. An example prompt is: *"Harry is tall. Tall people are round. Is Harry round?"* In the above example, the first sentence is a fact, the second sentence is a (universal quantification) rule, followed by a question that the model answers.

Fine-tuning on RuleTaker Dataset: We began with vanilla models to minimize confounding factors, including training effects. Across Gemma, DeepSeek, and GPT-2, performance remained near random, when using unfinetuned or in-context learning strategies(Appendix A).Therefore, we fine-tuned GPT2, DeepSeek, and Gemma on the ruletaker dataset. We fine-tune all models using Adam optimizer with 1e-3 learning rate to specialize their knowledge for our targeted reasoning task by taking a 80-20% split of the data, which involves structured logical rules beyond the scope of general pre-training. Furthermore, since our data set is formulated as a supervised classification problem with three labels, task-specific fine-tuning is necessary to align the outputs of the models with the label space. While GPT-2 is fine-tuned using full parameter training, we apply LoRA (Hu et al., 2021) for Gemma and DeepSeek due to their larger sizes, allowing efficient adaptation with minimal parameter updates. After finetuning, GPT-2 and Gemma achieve 96% classification accuracy on a held-out validation set, while DeepSeek achieves 93%.

Interchange Intervention Dataset: We curate subsets of the original RuleTaker dataset to assess model predictions and interchange intervention effectiveness. These datasets follow the format: *base prompt*, *override prompt*, *base answer*, *override answer*, *question*. The model is prompted with ' `<base> Question: <question >` '. The *base answer* is the model's original response, while the *expected answer* is what it should output after a successful interchange intervention. Questions are designed based on the type of interchange intervention performed, where we have questions that check the base information is removed and questions that check whether the information from the override prompt has been introduced. Here is an idealized example (without distractor sentences):

> **Base prompt:** Rabbit like squirrel. If something like squirrel then squirrel chase rabbit.
> **Override prompt:** Anne is young. If someone is quiet then they are young.
> **Question:** Squirrel chase rabbit?
> **Base Answer:** *True* **Override Answer:** *Unknown*

We design separate subsets targeting Conditional Rules and Universal Quantification.

4 Methodology

4.1 Interventions

Below we describe our interventions, which we use different terms for, even when they can be seen as special cases of each other. For each of these interventions, we intervene on the residual blocks. In particular, we intervene on the output from one residual block component, which acts as the input to the next residual block. We do this because the combined effect of attention and MLP components are captured in the block output, and we want to intervene on a "macro" layer behaviour as opposed to smaller subcomponents of the layer.

Zeroing-out Intervention: Here we selectively replace token activations with zero. Prior work has shown that such interventions can trigger compensatory behavior elsewhere in the network, leading

to emergent self-repair and potentially masking the causal importance of individual components (McGrath et al., 2023). To account for this, we perform two versions of the experiment. In one version, which we call *zero* (or necessity) intervention, a token activation z is zeroed out, and all other activations are left untouched. In a second version, which we call *non-zero* (or sufficiency) intervention, we leave a token activation z untouched, and zero out all other activations.

Interchange Intervention (Geiger et al., 2022) involves replacing the internal activation of the *base prompt* with that of the *override prompt* at a specific intervention layer and token position. Formally, let $\mathbf{z}^{(l)}_{\text{base},t}$ denote the output of the transformer block at layer l and token position t for the base prompt. We replace this with $\mathbf{z}^{(l)}_{\text{override},t}$ the corresponding activation from the override prompt and continue the forward pass using this modified representation.

This allows us to examine how information originating from the override prompt influences the model's response when introduced partway through computation on the base prompt. By performing interchange interventions across layers $l = 0, 1, ..., L$, we identify which parts of the network are most responsible for reasoning over logical structures. We use the NNsight library (Fiotto-Kaufman et al., 2024) to execute the interchange interventions.

Layer Swap Intervention: (Lad et al., 2025) We take the logit with the highest value, and check if it corresponds to True, False, or Unknown. We only considered datapoints where the model originally answers questions correctly, i.e. the logit for the correct class (True, False, Unknown) was highest. To do a layer swap intervention, we replace the entire layer activation of the intervention layer, with the entire activation of the swap layer.

4.2 Evaluation Metrics

Zeroing Out Accuracy After zeroing out tokens as described in Section 4.1, we obtain a predicted label by applying argmax over the output logits. Accuracy for each layer is computed as a fraction of samples for which predicted label matches the ground-truth label.

Interchange Intervention Accuracy (IIA) measures whether a model's response aligns with the expected outcome after an interchange intervention. For each input, we perform interchange intervention as defined in Section 4.1. The model is then queried with a follow-up question, and its answer is compared to the *expected answer* (i.e., the answer the model would give if it fully adopted the logic from the override prompt).

Let $y^{(i)}$ be the model's answer to the i-th intervened example, and let $\hat{y}^{(i)}$ be the corresponding *override answer*. Then IIA is defined as $\frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{\left[y^{(i)}=\hat{y}^{(i)}\right]}$, where $\mathbf{1}_{[\cdot]}$ is the indicator function and N is the number of examples.

This metric captures whether the interchange intervention successfully transfers logical structure from the override prompt into the base context. By computing IIA across layers, we identify where in the model such logical reasoning is most causally encoded.

Layer Swap Accuracy: We calculate the impact of the layer swap interchange intervention by taking the difference between the logit of the correct class (True, False, Unknown) when using the original layer's activations during inference, and the logit for the correct class when using the swapped activation during inference.

4.3 Intervention Targets

4.3.1 Punctuation Zeroing Out

targets are all the periods in a prompt, and/or the question mark in a prompt.

Punctuation Interchange Interventions targets are:

- **First Sentence**: For a base prompt, consider the first full sentence excluding the period. Swap the activations of that entire base sentence with those of an override.
- **First Period**: For a base prompt, consider the first period. Swap the activations at that period between a base sentence and an override sentence.

Let the base be *"Dave is nice. Fiona is grey. If someone is happy, they are cool."* and the override be *"Ben is purple. Adam is cool. All happy are sad."* In the first-period swap, we exchange the activation at the period following *"Dave is nice"* with that from the period after *"Ben is purple"*. We then ask a question derived from the override context, such as *"Is Ben purple?"* A correct response indicates that the information from the base context has been transferred via the period token — suggesting it

acts as a summarization. Additionally, we also experimented on additional punctuations,“
n”, and “END”, results of which are given in Appendix A.

Conditional Statements This rule structure captures logical implication, typically in the form “If A, then B". For example, given the base sentence *“Dave is nice. If Dave is nice then he is happy."* and a override sentence *“Ram is cool. If Ram is cool then he is great."*, we replace the activation of the consequent token *“happy"* in the base with *“great"* from the override. We then query the model with *“Is Dave great?"* to see whether it adopts the altered consequent, effectively testing whether it continues to reason over the modified rule.

Universal Quantification This rule type involves generalized statements about all members of a category, typically structured as *“All [adjective] things are [predicate]"*. These constructions are common in both natural language and formal logic and require the model to generalize from an adjective-noun combination to a property.

To study this, we use a base sentence such as *“All blue things are nice."* and a override sentence like *“All green things are great."* We intervene by replacing the activation of the predicate token *“nice"* in the base with the corresponding token *“great"* from the override. We then query the model with “Are all blue things great?" to test whether it adopts the altered predicate and violates or preserves the original generalization.

Layer Swaps target the activations of a prompt.

5 Results

We summarize our empirical findings below:

5.1 Punctuation Analysis

Sufficiency of Periods and Question Mark In Figure 1a, we selectively keep some tokens non-zero and zero out all other tokens for GPT2. Here we see that for later layers (layer 7 to 11) the performance remains high even when almost all tokens are zeroed out, as long as period and question tokens are non-zero. This means that the period and question mark tokens in these layers contain all information that is needed to answer the question, i.e. they are sufficient for answering the question.

Necessity of Periods and Question Mark We also find in Figure 1d for GPT2, that when period and question mark are both zeroed out (and all other tokens are non-zero), the performance is dramatically low, which means these tokens are necessary for answering the question.

DeepSeek and Gemma. For DeepSeek punctuation is sufficient in the first and last layer, and necessary in the first. For Gemma punctuation is sufficient in layer 12, 15 and 17, but never both necessary and sufficient.

We further dissect these results for GPT2 below

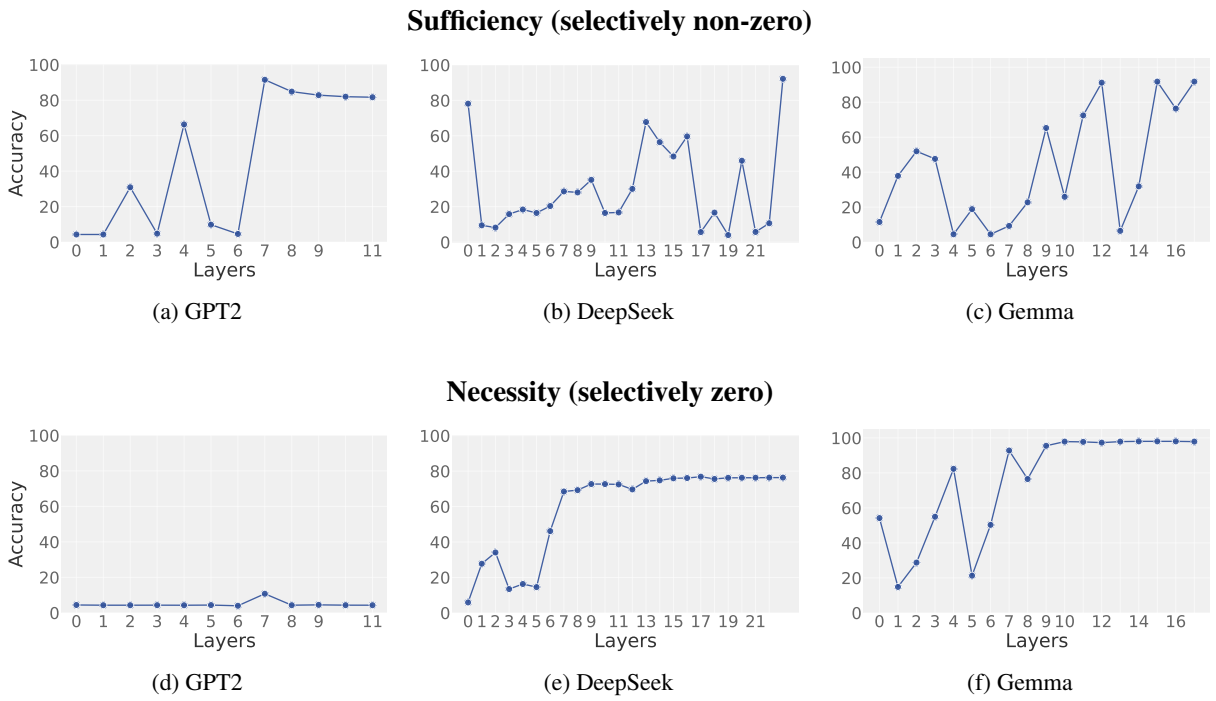
Question Mark is Necessary and Sufficient in Some Layers When question mark is the only non-zero token in layer 4 or in the last five layers (Figure 3b) performance is high, but when it is the only zeroed out token in layer 4 or the last five layers (Figure 3d) the performance is low. In contrast, period is necessary in layers 0 to 4 (Figure 3c), but not sufficient in those layers (Figure 3a).

Tokens Can “Dilute” Question Mark In Figure 3b we find that if question mark is the only non-zero token in layer 4, then performance is high. However, we also find in Figure 3c that if period tokens are the only zero tokens (so in particular question mark is non-zero) in layer 4 then performance is low. We hypothesize that this is because the many other non-zero tokens in Figure 3c “dilute” the question mark token. We investigate this hypothesis in Figure 2. In this figure question mark is always nonzero, periods are always zero, and we respectively nonzero one in 15, one in 5, one in 2, four in 5 other tokens. We find that performance is high when question mark is one of the few nonzero tokens (see Figure 2a) and becomes lower as we increase the proportion of nonzero tokens.

Transfer from Period to Question Mark We find that when period is non-zero in layer 7 (Figure 3a) or if question mark is non-zero in layers 7 to 11 (Figure 3b) then accuracy is high. We hypothesize this is because information can be “transferred” from period to question mark in layer 7.

First Full Sentence Interchange Intervention Accuracy (IIA) For all three models we find in Figure 4 that when we perform interchange intervention on the first full sentence of the prompt, the IIA starts out high, and then dramatically decreases.

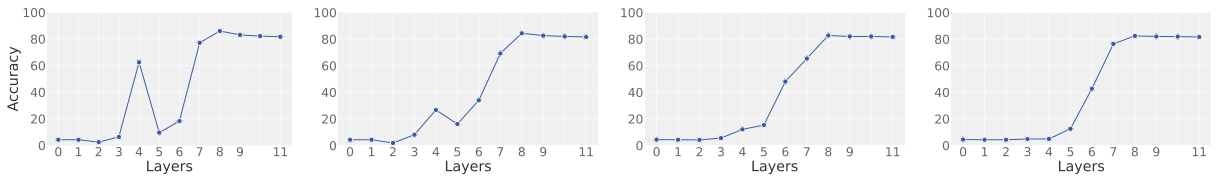
First Period IIA The results shown in Figure 4, also indicate that for GPT-2 the IIA peaks in layer 4. DeepSeek shows a similar but more abrupt



(a) GPT2 (b) DeepSeek (c) Gemma

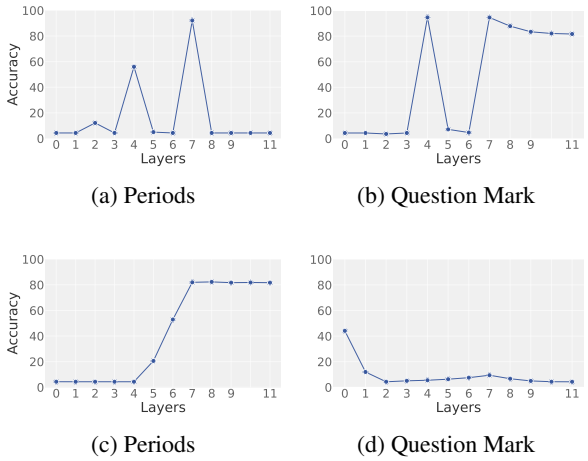
(d) GPT2 (e) DeepSeek (f) Gemma

Figure 1: Selectively either zeroing out period and question mark tokens or non-zeroing out only these tokens for GPT-2, DeepSeek and Gemma.



(a) One in 15 (b) One in 5 (c) One in 2 (d) Four in 5

Figure 2: Non-zeroing question mark and different proportions of extra tokens (while keeping periods zero) for GPT-2.



(a) Periods (b) Question Mark

(c) Periods (d) Question Mark

Figure 3: Selectively (non-)zeroing out either the period token activations or the question mark activation for GPT-2.

pattern, starting out at an IIA of just above 20%, remaining constant, suddenly peaking at layer 4 to around 60%, and then dropping to 20% again. Gemma, in contrast, shows negligible sensitivity to interchange interventions for first full period. In line with previous findings (Barbero et al., 2025), we hypothesize that this difference is related to the context window size of the models: GPT-2 has the smallest window, followed by DeepSeek and then Gemma, or due to Gemma being a distilled model.

First Sentence IIA Drop Coincides with First Period Peak for GPT2 and DeepSeek For both GPT-2 and DeepSeek, the layer where the IIA for the first period peaks (layer 4 in DeepSeek) coincides with a pronounced drop in the IIA for the first sentence. We hypothesize that in the first four layers information from the first sentence is "transferred" to the first period. This is corroborated by Figure 3c, where we find that periods contain important information in the first four layers. However, we expected the IIA to either be constantly

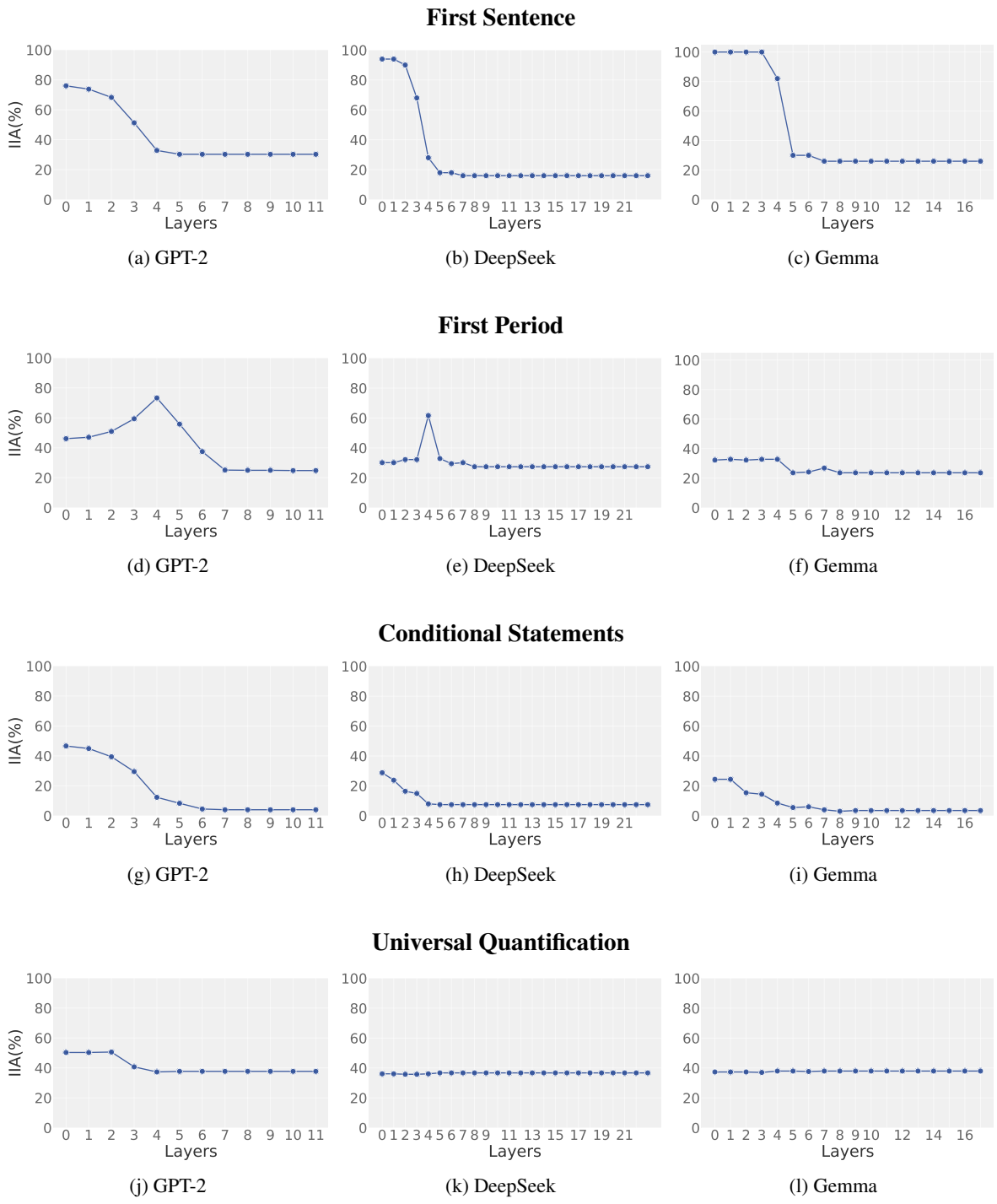
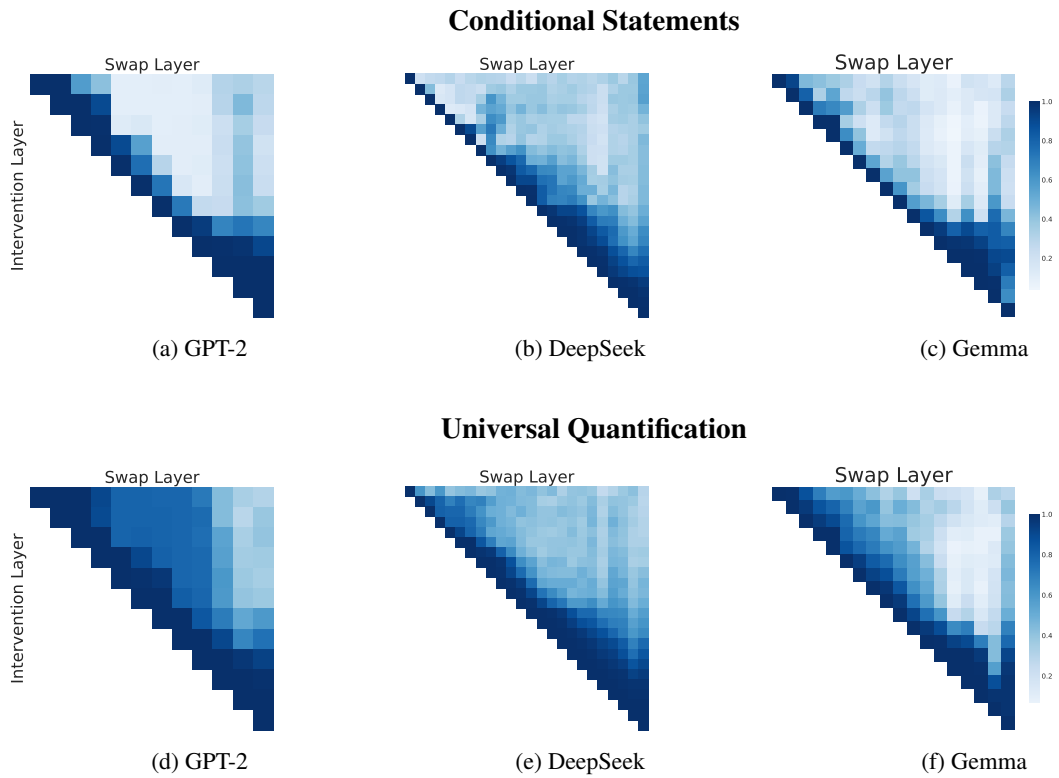


Figure 4: Layer-wise sensitivity of GPT-2, DeepSeek, and Gemma to interchange interventions applied to different intervention targets (first sentence, first period, conditional statements, universal quantification). Each row represents an intervention target; each column corresponds to a model. When applying interchange intervention we take a sentence and replace a sentence component z with an alternative z' and check whether the model's response aligns with the response we would expect if z' was in the prompt, if so we say the IIA for this datapoint is 1, if not it is 0. See the Evaluation Metrics section for more details on how IIA is calculated and the Intervention Targets section for more details on the intervention targets.

high, or the zeroing out accuracy to decrease across the first four layers. In Figure 6 in the appendix we find a similar pattern for second sentence and second period, but with a lower final IIA.

5.2 Reasoning Rules

Based on previous work (Sun et al., 2025; Yang et al., 2025) and before starting interchange intervention experiments on reasoning rules we had



(a) GPT-2 (b) DeepSeek (c) Gemma

(d) GPT-2 (e) DeepSeek (f) Gemma

Figure 5: Comparison of layer swap heatmaps across different language models. Each heatmap shows the correlation between layers when performing layer swapping experiments. Higher correlation values indicate that swapping those layers has minimal impact on model performance, while lower values suggest significant performance degradation.

expected to see the model collecting information in early layers and applying reasoning rules in later layers. However, for GPT-2 our results instead show that similar to first sentence, second sentence, adjective and subject interventions (Figure 4 and Figure 6 in the appendix) intervening on reasoning rules leads to high IIA in the early layers and low IIA in later layers.

Conditional Statements In Figure 4 we plot the IIA for conditional rules. This means that we take a sentence with "If x, then z" and replace the activations of the consequent z with the activations of z'. We find that GPT-2 has a high IIA in the first layer of around 50%, but this has drastically dropped by layer 6, where the IIA is around 5%, and the IIA remains low for all following layers. We interpret this as the model actively processing information about the consequent (z or z') in the first 5 layers, but after that (from layer 6 onward), the activations in previous layers already determine which consequent the model works with. For DeepSeek and Gemma we find a similar pattern, but here the IIA respectively starts around 30% and 25%, and respectively drops around layer 4 and layer 7. We find that proportionally DeepSeek processes the consequent very quickly, namely in 4 layers (out of 22), whereas GPT-2 has only completed processing the consequent after the middle layer.

Universal Quantification We plot the IIA for universal quantification in Figure 4. In a sentence like "All x are z", we replace the predicate z with z'. The surprising finding here is that for all models, the lowest IIA is still very high, around 40%. This means that for all layers replacing the predicate z with z' leads to the model answering as if the prompt said z' in 40% of the datapoints. In other words, the model does not "stop processing" the predicate after some layer (which is unlike how the models handled conditional rules).

IIA Drop for Conditional Statements Comparing adjective, subject, universal quantification and conditional statements, we find that conditional statements is the only intervention where the IIA drops to a very low point. Out of these universal quantification is the only target where the the IIA is constant for two models (DeepSeek and Gemma). We interpret this as the model revisiting the universal quantification in equal measure in every layer.

Layer Swaps In Figure 5 each entry represents the impact of a layer swap, where the x-axis represents the the layer we intervene on, and the y-axis represents layer we swap it with. The swap layer plot, see Figure 5, indicates that for conditional statements layers have specific functions, and can not replace each other, whereas for universal quantification layers are more replaceable. We speculate this may indicate that conditional statements are more difficult for the model to solve as compared to universal quantification.

6 Discussion

Our findings suggest that layer-wise reasoning behavior in language models is jointly shaped by the logical structure of the task and by architectural or training choices, rather than following a uniform early-to-late processing pipeline. Across models, we observe systematic differences in how punctuation and logical rules influence computation, reflecting variation in how information is compressed, propagated, and reused across layers.

One possible explanation for the strong causal role of punctuation in GPT-2 is that smaller models with shorter context windows rely more on early information compression, with punctuation acting as compact aggregation points that summarize preceding content into representations influential in later layers. By contrast, the weaker sensitivity to punctuation in Gemma and DeepSeek suggests a more distributed encoding of contextual information, potentially encouraged by larger context windows, normalization schemes, or training objectives that promote smoother representations.

The contrasting layer-wise behavior of conditional statements and universal quantification points to differences in computational structure: conditional reasoning involves early stage-specific commitments after which the consequent is no longer revisited, whereas universal quantification maintains persistent sensitivity across layers, suggesting that predicates are repeatedly maintained or re-applied. These patterns indicate that different forms of logical reasoning rely on qualitatively different internal strategies.

We emphasize that these interpretations are speculative, as our analysis does not directly identify underlying mechanisms; nonetheless, the consistent patterns we observe offer concrete hypotheses for future work on how architectural design and training regimes shape the internal organization of reasoning in language models.

7 Conclusion

Using necessity, sufficiency, and interchange interventions, we analyze the causal role of punctuation and logical rules in language model reasoning. We find clear model-specific differences, with GPT-2 relying strongly on punctuation across layers, while DeepSeek and Gemma show little or no such dependence, alongside systematic rule-level differences: conditional statements exhibit a rapid decline in interchange intervention accuracy, whereas universal quantification maintains more stable layer-wise sensitivity, indicating distinct inductive biases

Overall, our results indicate that reasoning behavior is shaped by both task structure and model design, and does not follow a uniform layer-wise progression. By revealing systematic variation in causal sensitivity across models and rule types, this work lays groundwork for future studies of how architectural and training choices influence internal reasoning dynamics.

8 Limitations and Future Work

Our necessity and sufficiency analyses are limited to punctuation tokens and to GPT-2, and future work could extend these interventions to other token types and models. We also restrict interventions to token-level manipulations at each layer; more fine-grained analyses, including subspace-level interventions, may yield deeper insight.

In our reasoning experiments, we intervene on parts of sentences by targeting consequents or predicates. While this reveals differences in interchange intervention accuracy, it remains unclear whether these effects reflect general syntactic processing or reasoning-specific mechanisms. Future work could address this by intervening on entire rules, for example by replacing "if–then" statements with "if and only if–then" constructions.

Acknowledgments

We would like to thank AI Safety Camp for hosting and supporting this project, and for providing essential compute resources. Sonakshi Chauhan is grateful to Atticus Geiger for his valuable research guidance, feedback, and support during the initial development of this project.

References

Federico Barbero, Álvaro Arroyo, Xiangming Gu, Christos Perivolaropoulos, Michael Bronstein, Petar Veličković, and Razvan Pascanu. 2025. Why do llms attend to the first token? *Preprint*, arXiv:2504.02732.

Amit Ben-Artzy and Roy Schwartz. 2024. Attend first, consolidate later: On the importance of attention in different llm layers. *Preprint*, arXiv:2409.03621.

Paul C. Bogdan, Uzay Macar, Neel Nanda, and Arthur Conmy. 2025. Thought anchors: Which llm reasoning steps matter? *Preprint*, arXiv:2506.19143.

Sonakshi Chauhan and Atticus Geiger. 2024. GPT-2 small fine-tuned on logical reasoning summarizes information on punctuation tokens. In *MINT: Foundation Model Interventions*.

Peter Clark, Oyvind Tafjord, and Kyle Richardson. 2020. Transformers as soft reasoners over language. *Preprint*, arXiv:2002.05867.

Subhabrata Dutta, Joykirat Singh, Soumen Chakrabarti, and Tanmoy Chakraborty. 2024. How to think step-by-step: A mechanistic understanding of chain-of-thought reasoning. *Preprint*, arXiv:2402.18312.

Jaden Fiotto-Kaufman, Alexander R Loftus, Eric Todd, Jannik Brinkmann, Caden Juang, Koyena Pal, Can Rager, Aaron Mueller, Samuel Marks, Arnab Sen Sharma, Francesca Lucchetti, Michael Ripa, Adam Belfki, Nikhil Prakash, Sumeet Multani, Carla Brodley, Arjun Guha, Jonathan Bell, Byron Wallace, and David Bau. 2024. Nnsight and ndif: Democratizing access to foundation model internals. .

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. 2021. Causal abstractions of neural networks. *Preprint*, arXiv:2106.02997.

Atticus Geiger, Zhengxuan Wu, Hanson Lu, Josh Rozner, Elisa Kreiss, Thomas Icard, Noah D. Goodman, and Christopher Potts. 2022. Inducing causal structure for interpretable neural networks. *Preprint*, arXiv:2112.00826.

Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. 2025. When attention sink emerges in language models: An empirical view. *Preprint*, arXiv:2410.10781.

Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Y. Wu, Y. K. Li, Fuli Luo, Yingfei Xiong, and Wenfeng Liang. 2024. Deepseek-coder: When the large language model meets programming – the rise of code intelligence. *Preprint*, arXiv:2401.14196.

Yifan Hou, Jiaoda Li, Yu Fei, Alessandro Stolfo, Wangchunshu Zhou, Guangtao Zeng, Antoine Bosselut, and Mrinmaya Sachan. 2023. Towards a mechanistic interpretation of multi-step reasoning capabilities of language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Duligur Ibeling and Thomas Icard. 2021. On open-universe causal reasoning. *Preprint*, arXiv:1907.02170.

Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng Lyu, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, and Bernhard Schölkopf. 2024. Cladder: Assessing causal reasoning in language models. *Preprint*, arXiv:2312.04350.

Geonhee Kim, Marco Valentino, and André Freitas. 2025. Reasoning circuits in language models: A mechanistic interpretation of syllogistic inference. *Preprint*, arXiv:2408.08590.

Vedang Lad, Jin Hwa Lee, Wes Gurnee, and Max Tegmark. 2025. The remarkable robustness of llms: Stages of inference? *Preprint*, arXiv:2406.19384.

Jieh-Sheng Lee and Jieh Hsiang. 2019. Patent claim generation by fine-tuning openai gpt-2. *Preprint*, arXiv:1907.02052.

Danni Liu and Jan Niehues. 2025. Middle-layer representation alignment for cross-lingual transfer in fine-tuned llms. *Preprint*, arXiv:2502.14830.

Linhao Luo, Jiaxin Ju, Bo Xiong, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. 2024. Chatrule: Mining logical rules with large language models for knowledge graph reasoning. *Preprint*, arXiv:2309.01538.

Thomas McGrath, Matthew Rahtz, Janos Kramar, Vladimir Mikulik, and Shane Legg. 2023. The hydra effect: Emergent self-repair in language model computations. *Preprint*, arXiv:2307.15771.

Anurag Mishra. 2025. Mechanistic interpretability of gpt-like models on summarization tasks. *Preprint*, arXiv:2505.17073.

Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. 2017. Feature visualization. *Distill*. Https://distill.pub/2017/feature-visualization.

Jacob Pfau, William Merrill, and Samuel R. Bowman. 2024. Let's think dot by dot: Hidden computation in transformer language models. *Preprint*, arXiv:2404.15758.

Aske Plaat, Annie Wong, Suzan Verberne, Joost Broekens, Niki van Stein, and Thomas Back. 2024. Reasoning with large language models, a survey. *Preprint*, arXiv:2407.11511.

Meng Qu, Junkun Chen, Louis-Pascal Xhonneux, Yoshua Bengio, and Jian Tang. 2021. Rnnlogic: Learning logic rules for reasoning on knowledge graphs. *Preprint*, arXiv:2010.04029.

Anton Razzhigaev, Matvey Mikhalchuk, Temurbek Rahmatullaev, Elizaveta Goncharova, Polina Druzhinina, Ivan Oseledets, and Andrey Kuznetsov. 2025. Llm-microscope: Uncovering the hidden role of punctuation in context memory of transformers. *Preprint*, arXiv:2502.15007.

Sergio Servantez, Joe Barrow, Kristian Hammond, and Rajiv Jain. 2024. Chain of logic: Rule-based reasoning with large language models. *Preprint*, arXiv:2402.10400.

Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Nikul Patel, Jalal Naghiyev, Yann LeCun, and Ravid Shwartz-Ziv. 2025. Layer by layer: Uncovering hidden representations in language models. In *Forty-second International Conference on Machine Learning*.

Qi Sun, Marc Pickett, Aakash Kumar Nain, and Llion Jones. 2025. Transformer layers as painters. *Preprint*, arXiv:2407.09298.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2020. olmpics – on what language model pre-training captures. *Preprint*, arXiv:1912.13283.

Xiaojuan Tang, Song-Chun Zhu, Yitao Liang, and Muhan Zhang. 2024. Rule: Knowledge graph reasoning with rule embedding. *Preprint*, arXiv:2210.14905.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, and 179 others. 2024. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Curt Tigges, Oskar John Hollinsworth, Atticus Geiger, and Neel Nanda. 2023. Linear representations of sentiment in large language models. *Preprint*, arXiv:2310.15154.

Constantin Venhoff, Iván Arcuschin, Philip Torr, Arthur Conmy, and Neel Nanda. 2025. Understanding reasoning in thinking language models via steering vectors. In *Workshop on Reasoning and Planning for Large Language Models*.

Zhiwei Wang, Yunji Wang, Zhongwang Zhang, Zhangchen Zhou, Hui Jin, Tianyang Hu, Jiacheng Sun, Zhenguo Li, Yaoyu Zhang, and Zhi-Qin John Xu. 2024. The buffer mechanism for multi-step information reasoning in language models. *Preprint*, arXiv:2405.15302.

Zhipeng Yang, Junzhuo Li, Siyu Xia, and Xuming Hu. 2025. Internal chain-of-thought: Empirical evidence for layer-wise subtask scheduling in llms. *Preprint*, arXiv:2505.14530.

Qinyuan Ye, Robin Jia, and Xiang Ren. 2025. Function induction and task generalization: An interpretability study with off-by-one addition. *Preprint*, arXiv:2507.09875.

Yang Zhang, Yanfei Dong, and Kenji Kawaguchi. 2024. Investigating layer importance in large language models. *Preprint*, arXiv:2409.14381.

Jingze Zhu, Yongliang Wu, Wenbo Zhu, Jiawang Cao, Yanqiang Zheng, Jiawei Chen, Xu Yang, Bernt Schiele, Jonas Fischer, and Xinting Hu. 2025. Layercake: Token-aware contrastive decoding within large language model layers. *Preprint*, arXiv:2507.04404.

A Appendix

Feature	GPT2-small	DeepSeek	Gemma
Architecture	Decoder-only	Decoder-only	Decoder-only
LayerNorm position	Post-LN	Pre-LN	Pre-LN
Positional embeddings	Absolute	RoPE	RoPE
Attention mechanism	MHA	GQA/MQA	MHA
Residual connection	Sequential	Parallel	Parallel
Activation function	GeLU	SwiGLU	GeLU
Model size	124M	1.3B	2B
Context length	1k	16k	8k
Training objective	Next-token prediction	Next-token + multi-token	Knowledge distillation
Memory optimization	Standard	Speed + memory efficient	Knowledge distill. + memory

Table 1: Architectural comparison of language models used in our experiments. All models follow the decoder-only transformer architecture with key differences in normalization, attention mechanisms, and optimization strategies.

Approx. Hours	GPU	Model	Memory
25 h	NVIDIA RTX 4090	Various	200GB

Table 2: Approximate compute resources used for experiments.

A.1 Reproducibility Statement

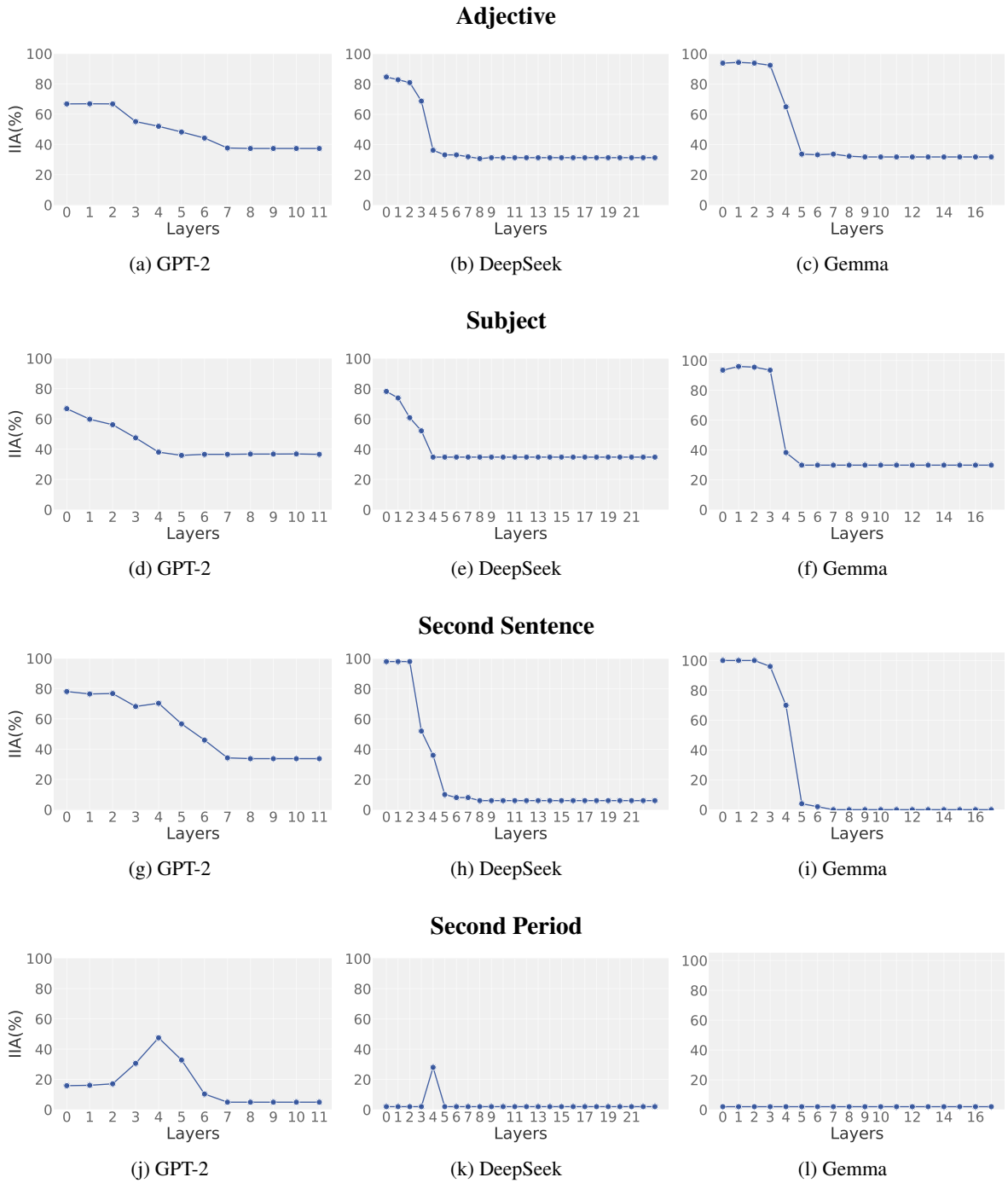
Our experiments were conducted using NVIDIA RTX 4090 GPUs. We evaluated a range of models across different families and sizes:

- **GPT-2 Small** (124M parameters) A fine-tuned version of OpenAI's GPT-2 model for patent claim generation, available under the MIT License: https://github.com/openai/gpt-2.
- **Gemma-2** (2B parameters) We used Gemma-2 2B released by Google DeepMind under the Gemma License: https://ai.google.dev/gemma/terms.
- **DeepSeek-Coder** (1.3B parameters) We used DeepSeek Coder 1.3B released by Hugging Face under the Apache 2.0 License: https://github.com/deepseek-ai/DeepSeek-Coder.

Our datasets and model configurations are described throughout this paper and in the Appendix to support reproducibility

Method	GPT2-small	DeepSeek	Gemma
Unfinetuned	27%	44%	45%
In-Context	32%	44%	44%

Table 3: Comparison of accuracy under ICL and unfinetuned settings across GPT-2, DeepSeek, and Gemma. We report overall task accuracy without any parameter updates to assess baseline model behavior independent of fine-tuning. All models achieve relatively low accuracy in both settings, motivating our focus on mechanistic comparisons rather than performance-driven analyses.



(a) GPT-2 (b) DeepSeek (c) Gemma

(d) GPT-2 (e) DeepSeek (f) Gemma

(g) GPT-2 (h) DeepSeek (i) Gemma

(j) GPT-2 (k) DeepSeek (l) Gemma

Figure 6: Layer-wise sensitivity to logical rule and period token interchange interventions across GPT-2, DeepSeek, and Gemma. Each row represents a rule type; each column corresponds to a model.

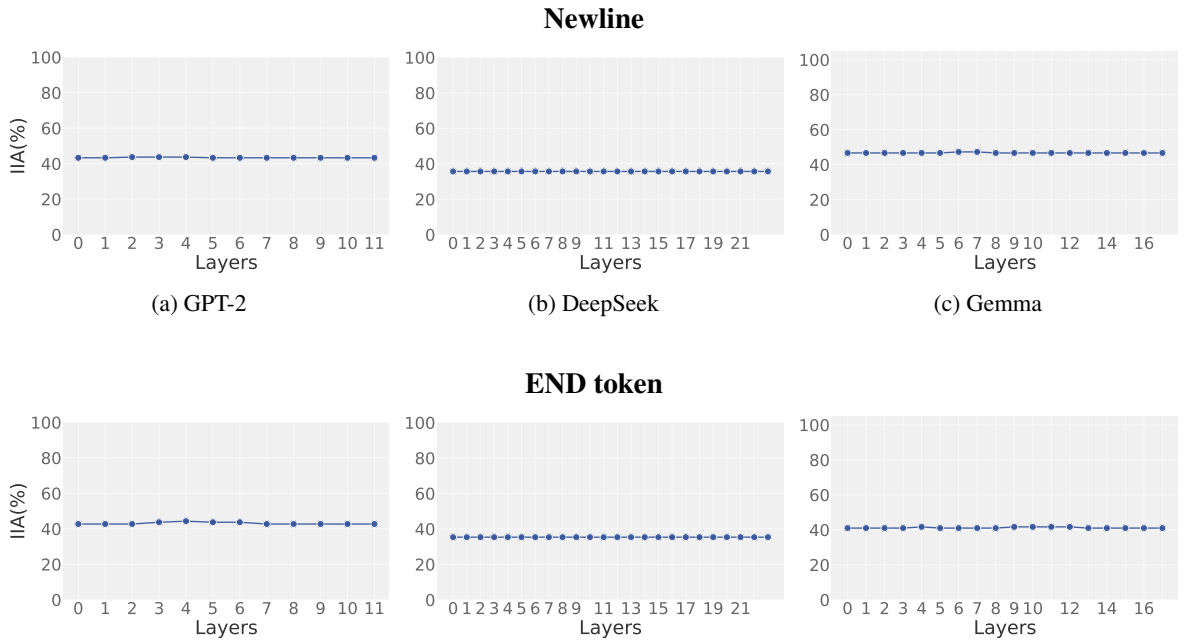


Figure 7: Layer-wise IIA for newline and END boundary tokens across GPT-2, DeepSeek, and Gemma. Each plot shows IIA as a function of the intervention layer when the boundary following the first sentence is marked using alternative tokens instead of periods.

B Usage of AI

This work made limited use of AI-assisted tools for secondary support tasks in writing and code review. Perplexity was occasionally used to obtain stylistic or organizational suggestions when revising the paper, such as rephrasing sentences for clarity or improving paragraph flow. All conceptual framing, argumentation, and interpretation of results were developed and written by the authors. Claude was used as a code-review assistant to identify potential implementation issues and improve code readability in experimental scripts. All experimental design choices, data processing steps, and final implementations were authored, verified, and executed solely by the research team. No text, analysis, or data generated by AI systems was included without human verification, and no AI system contributed original research ideas or interpretations. The use of AI assistants in this project conforms to standard publication ethics and authorship policies, which state that AI tools may aid researchers but cannot be listed as authors or credited with intellectual contributions.