Towards a Mechanistic Understanding of Propositional Logical Reasoning in Large Language Models

Danchun Chen[1,2]* **and Qiyao Yan**[1,3]* **and Chenpeng Wang**[5] **and Liangming Pan**[1,3,4]†
[1]State Key Laboratory of Multimedia Information Processing, Peking University
[2]Singapore Management University
[3]School of Computer Science, Peking University
[4]Beijing Academy of Artificial Intelligence, Beijing, China
[5]YiXin-AILab, YIXIN, Beijing, China
dcchen.nexus@gmail.com, yqy.1@foxmail.com, liangmingpan@pku.edu.cn

Abstract

Understanding how Large Language Models (LLMs) perform logical reasoning internally remains a fundamental challenge. While prior mechanistic studies focus on identifying task-specific circuits, they leave open the question of what computational strategies LLMs employ for propositional reasoning. We address this gap with a causal mechanistic analysis on *PropLogic-MI*, a controlled benchmark of 11 propositional rules across one- and two-hop tasks, applied to three model families (*Qwen3*, *Llama-3.1*, *Mistral*). Rather than asking which components are necessary, we ask how the reasoning process is organized, and identify four interlocking mechanisms: *Staged Computation*, where early, middle, and late layers take on distinct functional roles; *Information Transmission*, where semantic content aggregates at boundary tokens; *Fact Retrospection*, where fact tokens stay causally active as binding lookup in middle layers and as sustained access in late layers under hard reasoning loads; and *Specialized Attention Heads* that structurally implement these patterns. A prompt-order control that places the query before the facts confirms that this organization is model-internal rather than input-layout-induced. These findings show that pretrained LLMs solve propositional reasoning through a structured, layer-organized process that recurs across models, rule categories, and reasoning hops. [1]

1 Introduction

The capability of Large Language Models (LLMs) to execute complex logical reasoning has improved significantly (Cheng et al., 2025; El-Kishky, 2024; DeepSeek-AI et al., 2025). State-of-the-art evaluations demonstrated this trend: *GPT-5.2* achieves 92% accuracy on *GPQA-Diamond* and *Gemini-3-Pro* reaches 76% on *SimpleBench*. However, the underlying computational mechanisms behind such high empirical performance remain poorly understood. This gap in understanding raises a fundamental question: are these models simply sophisticated pattern matchers relying on surface statistics (Bender et al., 2021; Dziri et al., 2023), or have they internalized robust algorithms that mirror logical deduction (Li et al., 2023; Nanda et al., 2023)? To address this ambiguity, the emerging field of *Mechanistic Interpretability* (MI) (Sharkey et al., 2025) has begun to reverse-engineer transformers to causally uncover the mechanisms underlying model behaviors. Recent efforts have applied MI to reasoning domains including arithmetic (Kantamneni and Tegmark, 2025; Stolfo et al., 2023; Yu and Ananiadou, 2024), syllogistic inference (Kim et al., 2024), and multi-hop composition (Ye et al., 2025; Xia et al., 2024). Their findings suggest that LLMs may employ structured, interpretable strategies when processing logical reasoning, rather than relying solely on opaque pattern matching.

Propositional logical reasoning, *i.e.*, the capacity to manipulate boolean operators and chain inference rules, represents a cornerstone of formal deduction that underpins more complex reasoning tasks. Despite its foundational role, this domain remains largely unexplored in MI. Behavioral studies reveal that model accuracy degrades significantly with increased proof depth (Dziri et al., 2023), yet the internal computational mechanisms responsible for these failures remain opaque. A prior mechanistic work, Hong et al. (2024), provides circuit-level analysis identifying four attention head families in *Mistral-7B*, *Gemma-2-9B*, and *Gemma-2-27B*. While they demonstrate that similar circuits emerge across these models, their study is constrained to a single operator structure (*i.e.*, implication chains) using simplified one-hop problems. Broader propositional rules, including *De Morgan's Laws*, *Dis-*

*Equal Contribution.
†Corresponding Author.
[1]Code is available at https://github.com/yanqiyao1/Propo-MI.

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing, pages 28708–28737
October 24-29, 2026 ©2026 Association for Computational Linguistics

tributivity, and *Commutativity*, as well as applying multiple different rules in a multi-step reasoning process, remain mechanistically unexplored. Moreover, their circuit-discovery paradigm focuses on identifying *which specific components* are necessary for a given model-task pair, rather than characterizing *how* the reasoning process in computational strategies that may transfer across model architectures and logical structures.

In this paper, we address this gap with a more comprehensive analysis. First, we shift analytical focus from model-specific circuit identification to generalizable mechanistic principles. As shown in Figure 1, this perspective reveals four interlocking mechanisms: (1) *Staged Computation*, where fact premises, logical expressions and query compute mainly at early, middle, and late layers; (2) *Information Transmission*, where semantic information aggregates at semantic boundary tokens, *i.e.*, positions that close or anchor a meaningful segment such as a factual clause, logical expression, or query suffix; (3) *Fact Retrospection*, in which fact tokens remain causally active beyond early-layer encoding — as a middle-layer binding lookup and, under hard reasoning loads, as a sustained late-layer signal; and (4) *Specialized Attention Heads*, where heads specialize into Splitting, Transmission, and Fact-Retrieval roles, respectively marking segment boundaries, routing information within a logical block, and looking up bound fact values. Second, we expand the scope of analysis to 11 propositional logic rule categories across one- and two-hop reasoning on four models (*i.e.*, *Qwen3-8B*, *Qwen3-14B*, *Llama-3.1-8B-Instruct*, and *Mistral-7B-Instruct*), and find the same four mechanisms *generalize* across these axes. Third, we run every analysis under two prompt orderings, one where the facts precede the query and one where they follow it, to test whether the staged organization tracks the surface prompt layout or the underlying logical content. The same staged template emerges in both orderings across all models, indicating that the organization is model-internal rather than input-order-induced. These analyses provide a more complete and generalizable picture of the mechanisms of propositional logical reasoning.

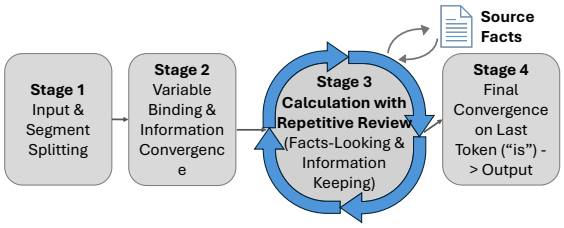
2 Related Work

A common three-level abstraction in *Mechanistic Interpretability* (MI) distinguishes features, circuits, and the algorithms they implement (Bereska and Gavves, 2024). Building on this view, we organize prior work on LLM reasoning along two complementary axes. A *component-level* lens asks *which* heads or layers implement a given sub-computation, while a *strategy-level* lens asks *how* those components are orchestrated into a network-wide reasoning pattern.

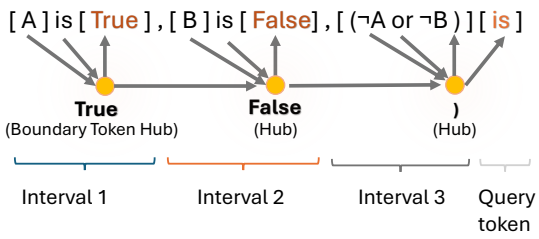
Component-Level MI. MI has identified specific components responsible for individual computational primitives: induction heads for in-context learning (Olsson et al., 2022), MLP layers that store factual associations (Meng et al., 2022; Geva et al., 2023; Ferrando and Voita, 2024), and binding ID vectors through which attention heads link entities to their attributes (Feng and Steinhardt, 2024; Wang et al., 2023b). Causal mediation analysis (Vig et al., 2020; Hase et al., 2023; Zhang and Nanda, 2024; Heimersheim and Nanda, 2024) and circuit-based methods (Elhage et al., 2021; Wang et al., 2023a) extend this by identifying minimal subgraphs causally necessary for specific tasks. For reasoning specifically, Kim et al. (2024) decompose syllogistic inference into modular circuits with premise-retrieval and conclusion-assembly components. These studies provide fine-grained causal evidence at the level of individual heads and layers.

However, as Tigges et al. (2024) observe, the specific attention heads realizing a functional component may shift across training and scale, yet the overarching algorithm they implement remains stable. Our work contributes primarily on the strategy-level axis while drawing its primitives from the component-level literature.

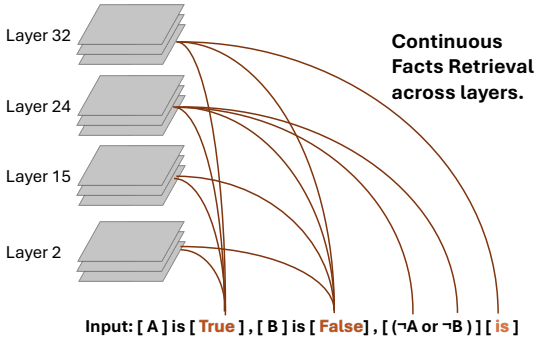
Strategy-Level MI. This line of research focuses on how individual components organize into network-wide reasoning strategies. Dutta et al. (2024) trace how *chain-of-thought* (CoT) prompting restructures internal information flow, showing that *explicit* intermediate tokens serve as anchor points for staged computation. Those tokens, however, are generated and re-enter the context as externalized working memory, so the organization built around them may belong to the scaffold instead of LLM itself. Moreover, the verbalized rationale itself need not faithfully reflect the computation that produces the answer (Turpin et al., 2023; Lanham et al., 2023; Paul et al., 2024). Therefore, we remove the scaffold and ask whether a similar organization appears when LLMs have to do all the reasoning *implicitly*. We find that models still stage



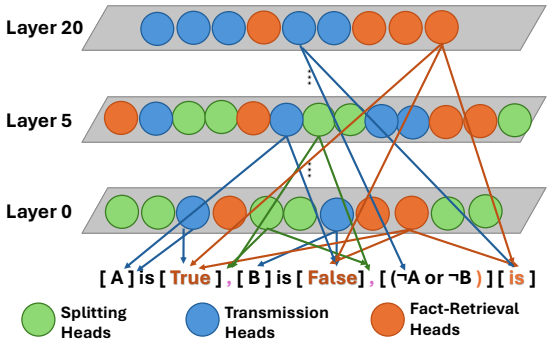
(a) **Staged Computation**



(b) **Information Transmission**



(c) **Fact Retrospection**



(d) **Specialized Attention Heads**

Figure 1: **Four interlocking mechanisms of propositional-logic reasoning.** (a) **Staged Computation**: early, middle, and late layers take on distinct functional roles. (b) **Information Transmission**: information aggregates at semantic boundary tokens. (c) **Fact Retrospection**: fact tokens used for middle-layer binding lookup and late-layer access under hard reasoning loads. (d) **Specialized Attention Heads**: Splitting, Transmission, and Fact-Retrieval heads implement the three patterns above.

the computation temporally across layers and spatially across token positions, anchored by boundary tokens instead of generated ones.

A strategy-level view also recasts primitives identified at the component level. Prior component-level studies show that attention heads link entities to attributes through *binding ID vectors* (Feng and Steinhardt, 2024) and that MLPs localize *single-fact* retrieval to specific layers (Meng et al., 2022). In the propositional setting, both observations resurface as part of a single *Fact Retrospection* pattern, in which fact tokens remain causally active beyond their initial encoding: middle layers carry a binding-flow lookup, and under hard reasoning loads (*e.g.*, two-hop reasoning) an additional sustained-access signal appears at late layers.

Mechanistic Studies of Propositional Logic. To the best of our knowledge, the only mechanistic study on propositional logic is Hong et al. (2024), which decomposes implication reasoning into four functional head types (locating, mover, processing, and decision heads) in *Mistral-7B* and *Gemma-2*, pinpointing the heads responsible for rule retrieval, information movement, and answer selection. Building on this *head-level* decomposition, we characterize *network-wide* computational patterns along three complementary axes: *temporal computation* across depth, *spatial information routing* across token positions, and the underlying *computational operators*. Combined with broader rule coverage and multi-hop scenarios, this strategy-level perspective reveals organizational principles that generalize across diverse logical structures, inference chain lengths, model scales and families.

3 PropLogic-MI: Task and Dataset

3.1 Task Formulation

We formulate propositional reasoning as *next-token prediction* over a *logical instance* $I = (F, E, Q)$, where F is a set of factual assignments, E is a set of logical expressions, Q is the queried expression, and the gold label $y \in \{\texttt{True}, \texttt{False}\}$ is determined by I. We study two prompt ordering, *i.e.*, π_{FF} listing F before E, and π_{EF} placing E first. Both preserve the same logical contents and the same answer, so differences between them reflect input order. The model is required to emit the truth value immediately after the query for analyzing the hidden computation that precedes the answer.

3.2 Dataset Construction

PropLogic-MI. *PropLogic-MI* is a controlled corpus for mechanistic analysis. It spans 11 propositional rule categories and two reasoning depths: (1) *one-hop* questions can be answered directly from the stated facts, and (2) *two-hop* reasoning, where an intermediate assignment $M = e_1$ must first be derived before predicting the final answer. Each logical sample is instantiated under both prompt orderings (*i.e.*, π_{FF} and π_{EF}) with matched clean/corrupted pairs, shared labels, and randomized variable remapping. Full rule definitions, prompt templates, and data generation details can be seen at Appendix A.

Data Statistics. Each prompt ordering contains 20,000 clean/corrupted pairs (9,523 one-hop and 10,477 two-hop), with near-balanced rule coverage; Table 1 summarizes clean accuracy and the *dual-correct* rate, namely the fraction of prompts that a model answers correctly on both the clean and corrupted runs. This dual-correct subset defines the sample pool for experiments. *Qwen3* yields larger pools than *Llama* and *Mistral*, and the same four mechanisms recur across families and analysis is restricted to these matched pools. Therefore, we only claim the strategy-level mechanism is shared conditionally on successful reasoning; how a model fails, and whether failure mechanisms coincide across models, is not addressed by this pool.

	π_{FF}		π_{EF}		
Model	Clean	Dual	Clean	Dual	Pool (π_{FF}/π_{EF})
Qwen3-8B	87.0	75.4	74.2	53.9	15,087 / 10,778
Qwen3-14B	87.4	76.6	74.4	52.8	15,310 / 10,562
Llama-3.1-8B	63.9	38.4	60.3	37.0	7,674 / 7,403
Mistral-7B	65.1	35.1	66.7	40.0	7,022 / 7,995

Table 1: **Data Statistics of *PropLogic-MI*.** Each prompt ordering contains 20,000 prompts. Clean = clean accuracy (%). Dual = dual-correct rate (%), *i.e.*, accuracy on both the clean and corrupted runs. The right column gives the pool size used for mechanistic analysis. Hop-wise and model-family breakdowns are deferred to Appendix C (including the cross-family summary in Table 7).

4 Methodology

Causal Interventions. We use two complementary intervention primitives. The intervention's exact scope (which layers, which token positions, which to average) varies per experiment and is described with experiments in §5. (1) **Activation patching.** Following Vig et al. (2020); Meng et al. (2022), we form a clean prompt x_{clean} with gold label y and a minimally corrupted (*e.g.*, flipping the assignment of a fact) prompt x_{corr} whose prediction flips. We cache an internal activation from the clean run at a chosen layer and token and restore it into the corrupted run, measuring how much the corrupted output moves back toward y. (2) **Mean ablation.** We replace a targeted output by a defined mean value computed over a clean reference set. The exact scope of the mean differs across experiments.

Analytical Metrics. Let w_y denote the token for the correct label $y \in \{\texttt{True}, \texttt{False}\}$ and $w_{\neg y}$ its opposite. We define **Probability Difference** as

$$\mathrm{PD}(x, y) = \mathrm{P}(w_y \mid x) - \mathrm{P}(w_{\neg y} \mid x), \quad (1)$$

where $\mathrm{P}(w \mid x)$ is the model probability at the answer position. Larger PD means stronger support for the correct answer. For an intervention $\tilde{x}$, we report **Probability Difference Shift** as

$$\mathrm{dPD} = \mathrm{PD}(\tilde{x}, y) - \mathrm{PD}(x_{\text{baseline}}, y), \quad (2)$$

where x_{baseline} is the input of the unintervened run (the corrupted prompt for activation patching; the clean prompt for mean ablation). A positive dPD indicates that the intervention restores the model toward the correct answer.

Semantic Aggregation. To localize computation along the prompt, we take two semantically aggregated granularities (Figure 2): (1) *Coarse logical blocks* partition each prompt into three contiguous, role-defined spans — the `Fact Block` (*e.g.*, `A is True, B is False`), the `Expression Block` (*e.g.*, `C = (¬A or ¬B)`), and the `Query Block` (*e.g.*, `C is`) — and are intervened on as a whole (used in §5.1). (2) *Fine structural categories* further subdivide each block into syntactically meaningful token classes — the boundary tokens `fact_value`, `expr_last`, and `query_last` that close each block, vs. interior tokens such as `fact_var_ref` or `operator` — and are scored at the token level (used in §5.2). Both granularities share a single per-layer aggregation. Let $\mathcal{T}_c$ denote the units belonging to category c; the mean dPD at layer l is

$$\overline{\mathrm{dPD}}_c^{(l)} = \frac{1}{|\mathcal{T}_c|} \sum_{t \in \mathcal{T}_c} \mathrm{dPD}^{(l)}(t), \quad (3)$$

where the unit t is set by the granularity: under (1), t is an entire logical block and $|\mathcal{T}_c| = 1$, so $\overline{\mathrm{dPD}}_c^{(l)}$ reduces to the block-level dPD; under (2), t ranges over token positions inside structural category c and $|\mathcal{T}_c|$ is the number of such tokens.

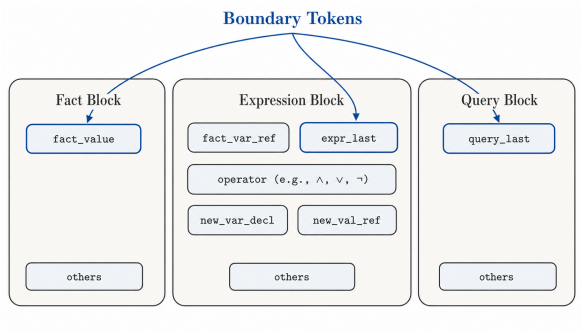


Figure 2: **Two token-categorization granularities.** Coarse logical blocks (`Fact`, `Expression`, `Query`) nest the fine structural categories; arrows mark the boundary tokens (`fact_value`, `expr_last`, `query_last`). Two-hop prompts additionally introduce `new_var_decl` and `new_val_ref` inside the Expression Block.

5 Experiments

We analyze propositional reasoning using coarse-to-fine intervention granularity. We first use mean ablation on MLP and Attention (§5.1) to when each *logical block* (*i.e.*, Facts, Expression, Query) is processed temporally. Next we zoom in to the *structural categories* and ask *how* information routes (§5.2). We finally identify *which* attention heads implement the above patterns as discrete *computational operators* (Wang et al., 2023a) (§5.3.1). We run the full pipeline on four LLMs: *Qwen3-8B/14B*, *Llama-3.1-8B-Instruct*, and *Mistral-7B-Instruct*. Main-text figures mainly use *Qwen3* as representative cases.

5.1 Coarse-grained Analysis: Staged Computation

We begin with a fundamental question: *whether LLMs organize propositional reasoning into computational stages given different logical blocks?* Therefore, we mean-ablate each logical block with the token-wise mean computed over the input prompt at every layer in three complementary pathways (MLP, Attention, and the joint Attention+MLP) to remove that block's information without injecting any outlier signal and isolate the layers at which the block carries necessity load.

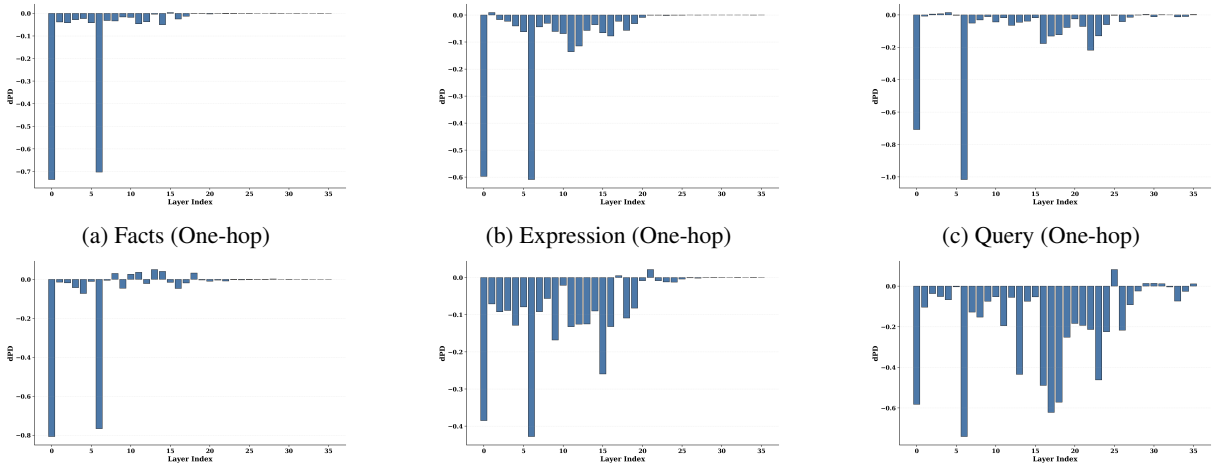
The same staged pattern appears across all three ablation pathways. Figure 3 shows the representative *Qwen3-8B* joint ablation. `Fact_Block` concentrates in the early layers, while `Expression_Block` dominates in the middle layers and `Query_Block` in the late layers. Notably, few early layers carry significant effects for all logical blocks simultaneously, so we concentrated distributions rather than peaks, and the meaningful signal is the *relative order* of the three distributions instead of absolute layer cutoff. Concretely, the `Expression` distribution lies after the `Fact` distribution, and the `Query` lies after the `Expression`; two-hop reasoning broadens the later computation but preserves this computational ordering. Results of MLP ablation and Attention ablation are in (Figure 17, 24; Appendix D.3).

The same ordering survives prompt-order reversal and model change. Under π_{EF} the three distributions keep the same order as under π_{FF}, but the staging is more compact: the `Fact` concentration arrives roughly one to two layers later, while the `Expression` and `Query` concentrations move few layers earlier, so the three stages occupy a narrower depth window (Appendix D.1; compare Figures 7 and 8). Scaling to *Qwen3-14B* reproduces the same template (Figure 9; Appendix D). The ordering also holds across model families, with two coexisting family differences: *Llama-3.1-8B* and *Mistral-7B* place the `Fact` and `Expression` stages at earlier layers than *Qwen3*, while devoting more overall mass to the `Query` block and keeping a broader late-integration tail (Figures 11, 13; Appendix D.1).

5.2 Fine-grained Analysis: Information Routing

This section investigates *how information routes between computational stages* through fine-grained activation patching on residual stream, as it serves as the primary information pathway and accumulates representations across layers, By patching residual stream activations at specific token positions, we can trace where stage-relevant information resides as it propagates through the network.

For each item we build a clean/corrupt pair that flips the label. The clean prompt is, *e.g.*, `A is True, B is False, C = (¬A or ¬B). C is` (answer `True`); the corrupt is `A is True, B is True, C = (¬A or ¬B). C is`" (answer `False`). We use metric 3 to aggregate token-wise causal influence with each structural categories (Figure 4a).



(a) Facts (One-hop) (b) Expression (One-hop) (c) Query (One-hop)

(d) Facts (Two-hop) (e) Expression (Two-hop) (f) Query (Two-hop)

Figure 3: **Joint Attention+MLP mean ablation on *Qwen3-8B* under π_{FF}.** Per-layer dPD when the activations of each logical block (Facts/Expression/Query) are mean-replaced. Facts effects concentrate in early layers, Expression in middle layers, and Query in late layers; two-hop preserves the ordering while broadening the Query(§5.1).

5.2.1 Information Converges at Boundary Tokens

Figure 4a shows the representative *Qwen3-8B* run under π_{FF}. The three boundary-token structural categories dominate their respective computational stages: `fact_value` in early layers, `expr_last` in middle layers, and `query_last` in late layers, while the remaining categories (`fact_var_ref`, `operator`, `others`) stay negligible. Causal influence concentrates on a small set of *boundary tokens* that close or anchor each logical block, rather than spreading uniformly across the block. The token-level transport runs Fact → Expression → Query through these positions, mirroring the layer-level staging in Figure 3.

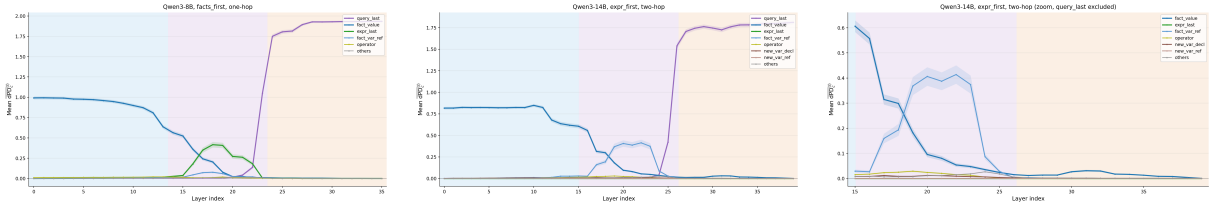
Under π_{EF}, the clean and corrupt prompts differ only at the `fact_value` position (the flipped truth assignment) and the model is autoregressive, so patching the clean residual of any expression token (*e.g.*, `expr_last`) into the corrupt forward pass transfers no information and produces no dPD by construction. The other two boundary tokens continue to dominate their respective layers under π_{EF} (Figure 4b, Appendix E), confirming that boundary-token convergence generalizes across different prompt orderings.

The same boundary-token rule survives two-hop reasoning and replication across all four models. The relative weight of each boundary differs by family: *Qwen3* keeps the strongest fact-side signal, *Llama* hands off to query tokens earliest, and *Mistral* lies in between. Details are in Appendix E.

5.2.2 Fact Retrospection

The `query_last` in late layers peak visually compresses what happens in the later layers. To reveal the information flow during expression processing, where expression-variable tokens bind to fact values, we exploit the autoregressive suppression of `expr_last` under π_{EF} and zoom in middle and late layers with `query_last` excluded.

As shown in figure 4c, `fact_value` and `fact_var_ref` exchange dynamically across the middle layers: as the layer index advances, `fact_value`'s dPD declines from its early-layer peak while `fact_var_ref` rises to a middle-layer peak within the same layer range. This indicates information routing during fact binding: at these depths, attention heads use `fact_var_ref` as the query position and copy the truth value from `fact_value`, shifting causal effects from `fact_value` to `fact_var_ref` across middle layers. The exchange between `fact_value` and `fact_var_ref` generalizes across reasoning hop, prompt order, and models, although magnitude differs. *Qwen3* shows the clearest middle-layer exchange, *Llama-3.1* is weaker, and *Mistral* is negligible (Appendix E). A complementary yet counterintuitive pattern emerges as tasks get harder: factual influence outlasts the binding step. Specifically, in the *late* layers, `fact_value` stays well above the other structural categories excepts `query_last`, and is markedly significant under two-hop compared to one-hop for the same model and prompt order (Appendix E). We call the pattern as *Fact Retrospection* that LLMs keep attending back to



(a) *Qwen3-8B*, π_{FF}, one-hop (b) *Qwen3-14B*, π_{EF}, two-hop (c) *Qwen3-14B*, π_{EF}, two-hop (zoom)

Figure 4: **Per-token mean** dPD **by structural category.** Residual-stream activation-patching effects averaged per token within each category; shading marks Early/Middle/Late layers. (a) Under π_{FF}, boundary tokens dominate their respective layers—`fact_value` early, `expr_last` in middle, `query_last` late—while the remaining categories stay near zero. (b) Under π_{EF}, `expr_last` is pinned to zero by autoregressive construction, but `fact_value` and `query_last` still dominate the early and late layers, confirming that boundary-token convergence persists under prompt reordering. (c) Zoom of (b) into middle and late layers with `query_last` excluded: the binding-flow signature during expression process between `fact_value` and `fact_var_ref`.

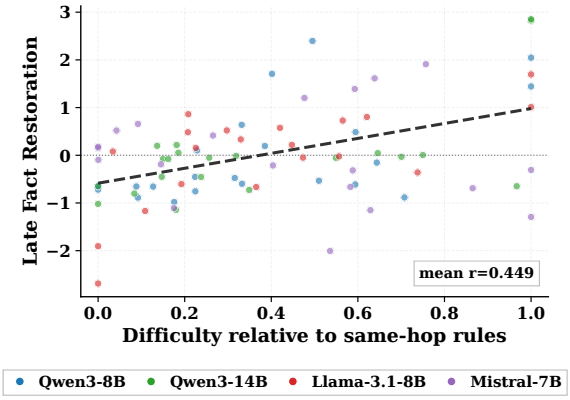


Figure 5: **Harder tasks show stronger Fact Retrospection.** Each point is one rule × hop group, colored by model and plotted against its difficulty relative to other rules at the same hop. Both axes are rescaled separately for each model and prompt order. The dashed line is the mean trend.

`fact_value` positions even after most information has already routed forward to `query_last`.

To quantify the relationship between task difficulty and fact retrospection, we group tasks by hop and logical rule within each model and prompt order. A group's difficulty is one minus the average of its clean and corrupted accuracy, centered within hop so that each group is scored against other rules at the same depth. *Late Fact Restoration* (LFR) is the mean dPD from patching clean `fact_value` activations into the paired corrupted run, averaged over the fact-value positions and over the late layers of each model. Using 1,000 patching examples per model and prompt order, difficulty is positively associated with LFR (Figure 5; $\bar{r} = 0.449$ over the 8 model × prompt-order conditions, stratified permutation $p = 0.0004$). Therefore, *Fact Retrospection* strengthens as tasks get harder.

5.3 Computational Operaters Analysis: Specialized Attention Heads

5.3.1 Head Taxonomy

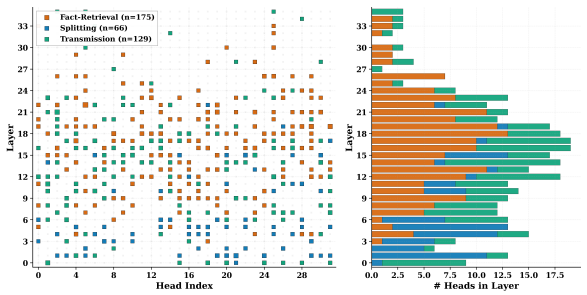


Figure 6: **Layer-wise distribution of the three head roles in *Qwen3-8B* under π_{FF}.** Counts of heads classified as Splitting, Transmission, and Fact-Retrieval at each layer. Splitting Heads concentrate in early layers, Transmission Heads in middle layers, and Fact-Retrieval Heads peak from the middle into late layers. Scale, prompt-order, and cross-family extensions are in Appendix F.

With the macroscopic computational patterns characterized in the preceding sections, the last core question remains: *which attention heads implement these patterns as* computational operators*?* Analyzing the attention patterns, We find Specialized Attention Heads, *i.e.*, a consistent, layer-wise division of labor in which heads at different layers assume specialized, stable functional roles. We group these heads into three main categories, each corresponding to a macroscopic mechanism identified earlier. Detailed attention patterns of these heads could refer to figure 38. We further compute the layer-wise distribution of these heads (Figure 6). The depth ranges where each role concentrates mirror the macroscopic staging of §5.1–§5.2.2.

		Qwen3								Llama-3.1-8B				Mistral-7B			
		8B-FF		8B-EF		14B-FF		14B-EF		FF		EF		FF		EF	
k	**Role**	Acc	dPD	Acc	dPD	Acc	dPD	Acc	dPD	Acc	dPD	Acc	dPD	Acc	dPD	Acc	dPD
–	Baselines	96.6	–	92.8	–	94.8	–	97.6	–	60.2	–	69.2	–	99.4	–	99.8	–
8	Fact-Retrieval	74.6	−0.40	77.4	−0.26	83.2	−0.24	86.0	−0.23	55.2	−0.16	58.4	−0.20	41.6	−0.21	65.4	−0.13
	Transmission	80.4	−0.29	78.0	−0.26	76.8	−0.34	79.8	−0.33	55.2	−0.17	58.4	−0.19	45.2	−0.14	77.6	−0.10
	Splitting	84.6	−0.21	81.2	−0.19	90.4	−0.09	93.4	−0.09	55.2	−0.18	58.4	−0.24	47.0	−0.11	50.4	−0.19
	Mixed-All	74.6	−0.40	77.6	−0.27	74.2	−0.39	81.6	−0.34	55.2	−0.18	58.4	−0.22	41.6	−0.23	47.4	−0.24
	Random-Outside	97.1	+0.01	94.7	+0.02	96.5	+0.02	98.0	+0.00	62.4	+0.03	72.6	+0.03	95.9	+0.02	94.5	+0.02
16	Fact-Retrieval	67.4	−0.55	70.0	−0.41	76.8	−0.37	81.4	−0.31	55.2	−0.16	58.4	−0.22	41.2	−0.24	44.6	−0.21
	Transmission	69.4	−0.51	75.0	−0.32	72.2	−0.43	76.0	−0.44	55.2	−0.18	58.4	−0.20	41.6	−0.19	56.4	−0.14
	Splitting	83.4	−0.23	78.2	−0.25	88.2	−0.14	90.6	−0.14	55.0	−0.19	58.4	−0.23	42.2	−0.15	43.6	−0.24
	Mixed-All	68.4	−0.54	71.2	−0.40	65.8	−0.53	76.2	−0.43	55.2	−0.17	58.4	−0.22	41.4	−0.25	43.4	−0.27
	Random-Outside	97.4	+0.02	95.0	+0.01	96.9	+0.03	97.9	+0.00	65.5	+0.07	74.1	+0.05	92.8	+0.04	90.7	+0.04
32	Fact-Retrieval	57.0	−0.74	63.2	−0.54	63.8	−0.58	77.2	−0.42	55.2	−0.16	58.4	−0.21	41.2	−0.28	43.8	−0.25
	Transmission	69.2	−0.52	73.2	−0.36	69.2	−0.51	76.6	−0.44	55.2	−0.18	58.4	−0.21	41.2	−0.23	44.8	−0.19
	Splitting	80.4	−0.30	73.2	−0.36	84.8	−0.21	86.8	−0.23	55.2	−0.19	58.4	−0.25	41.4	−0.18	43.2	−0.26
	Mixed-All	62.0	−0.66	67.4	−0.48	61.4	−0.62	62.0	−0.68	55.2	−0.17	58.4	−0.22	41.2	−0.28	43.2	−0.31
	Random-Outside	97.2	+0.02	92.5	−0.02	98.3	+0.05	98.1	+0.01	72.0	+0.12	79.2	+0.09	88.4	+0.07	85.2	+0.07
64	Fact-Retrieval	50.6	−0.88	58.4	−0.61	57.2	−0.74	78.0	−0.41	55.2	−0.16	58.4	−0.22	41.2	−0.29	43.2	−0.30
	Transmission	62.6	−0.65	60.0	−0.62	57.4	−0.72	69.4	−0.56	55.2	−0.17	58.4	−0.22	41.2	−0.28	43.2	−0.24
	Splitting	70.4	−0.48	69.2	−0.43	75.2	−0.39	64.0	−0.66	55.2	−0.20	58.4	−0.23	41.2	−0.23	43.2	−0.31
	Mixed-All	52.2	−0.83	57.4	−0.67	53.0	−0.81	56.8	−0.85	55.2	−0.16	58.4	−0.21	41.2	−0.31	43.2	−0.33
	Random-Outside	91.1	−0.09	86.7	−0.11	97.1	+0.04	98.3	+0.01	78.9	+0.16	84.4	+0.17	79.5	+0.14	77.6	+0.12

Table 2: **Post-ablation accuracy and mean** dPD **shift after top-k head ablation, across four models and both prompt orders.** For each setting we report **Acc**, the post-ablation top-1 accuracy on held-out clean prompts (%), and **dPD**, the mean shift in the two-answer probability difference $\mathrm{PD}_{\texttt{True}} - \mathrm{PD}_{\texttt{False}}$ relative to the unintervened baseline (more negative = stronger push away from the correct answer). FF = π_{FF}, EF = π_{EF}. *Mixed-All* ablates the union of the top heads of all three roles; *Random-Outside* control is sampled from heads not assigned to any role.

Splitting Heads. In early layers, these heads concentrate on comma tokens (Figure 38a), which act as boundary markers between logical blocks. They materialize Staged Computation: by marking clause boundaries, they separate the factual premises from the expression to be solved.

Transmission Heads. These heads aggregate information within a logical block and route it toward the block's terminal token. Their attention maps form lower-triangular blocks inside the Facts and Expression blocks (Figure 38b), matching the boundary-token convergence pattern of §5.2.1.

Fact-Retrieval Heads. These heads route attention from the query token or expression tokens back toward truth-value tokens (`True`/`False`) (Figure 38c), and they concentrate in middle-to-late layers (Figure 6). This is the head-level counterpart of the information routing in §5.2.2: the same depth range where fact-value tokens remain above zero is occupied by heads which retrieve those fact tokens.

5.3.2 Ablation Study

To verify whether the head taxonomy genuinely implements the above mechanisms, we perform targeted mean-ablation of the top-k heads of each role and compare them with a *Random-Outside* control sampled from heads not assigned to any role (Appendix B.4). We report the model's **post-ablation accuracy** on held-out clean prompts together with the **mean** dPD **shift**, the change in the two-answer probability difference $\mathrm{PD}_{\texttt{True}} - \mathrm{PD}_{\texttt{False}}$ relative to the unintervened baseline.

Table 2 reports both metrics across $k \in \{8, 16, 32, 64\}$ for all models and both prompt orders. In *Qwen3*, role-targeted ablations consistently drop accuracy well below the *Random-Outside* control, with the ordering Fact-Retrieval $\geq$ Transmission $>$ Splitting robust across sizes and prompt orders; the cross-family rows show two distinct failure modes, with *Llama3.1* saturating by $k = 8$ (roles not cleanly separated) and *Mistral* degrading sharply under both role-targeted and *Random-Outside* ablations (globally sensitive).

6 Discussion

The most consequential finding concerns *Fact Retrospection*: how the model keeps fact tokens causally active beyond their initial encoding, in two sub-patterns. (1) A *binding-flow lookup* in middle layers that transfers from `fact_value`

to expression-side variable references, with Fact-Retrieval heads routing from query and expression positions to the bound truth-value tokens. This sub-pattern is universal across our models and hops, and is the propositional-logic counterpart of the binding-ID mechanism of Feng and Steinhardt (2024). (2) A *sustained late-layer access* that emerges only as the task nears the model's capacity limit (*e.g.*, on two-hop setting), where `fact_value` stays causally active in parallel with forward routing toward the answer, showing the model holding onto premise tokens when it cannot fully compress them into a single forward path.

The cross-family pattern further ties this mechanism to model capacity. The staged template recurs without exception across families, but how cleanly the three head roles separate from the *Random-Outside* control varies considerably: *Qwen3* shows the cleanest separation, *Mistral* is globally sensitive (both role-targeted and random ablations damage accuracy), and *Llama* saturates at small k. The same weaker models also produce the strongest late-layer sub-pattern of Fact Retrospection on multi-hop. Combined, we suppose that the mechanism we describe runs in all four models, but it is implemented more cleanly in models with capacity headroom for the task. While the link between the mechanism and task difficulty is established in §5.2.2, a more fine-grained analysis remains necessary, particularly regarding individual logical rules.

7 Conclusion

In this paper, we present a mechanistic analysis of propositional logical reasoning and adopt a top-down analytical strategy from macroscopic patterns to microscopic implementations. Our investigation uncovers a coherent computational architecture: these LLMs organizes reasoning into temporally distinct phases, routes information through semantic boundary tokens, maintains active fact retrieval, and implements these strategies through specialized attention heads. These four phenomena form an interlocking system where each mechanism supports and validates the others. Furthermore, these mechanisms generalize across models, rule categories, and reasoning hops, indicating a robust property of LLM propositional reasoning rather than model- or task-specific artifacts. By characterizing *how* models reason rather than *which components* are necessary, we contribute a strategy-level mechanistic understanding of propositional reasoning in LLMs.

Limitations

Our analysis has three main limitations. First, while we causally validate attention-head specialization in detail, the internal computations within MLP layers remain largely uncharacterized. Our MLP patching reveals *when* MLPs are causally important, but not *what* transformation they implement at each stage — whether they encode fact values, execute rule-specific Boolean operations, or integrate bound truth values at the answer position — nor how these transformations emerge during training.

Second, the task setting is deliberately controlled: prompts are short, symbolic, and limited to propositional rules with one- and two-hop chains; our four models span three families (*Qwen3*, *Llama-3.1*, *Mistral*). Whether the same staged organization persists with longer chains, natural-language phrasings, richer logical systems, or explicit intermediate-step formats remains an open empirical question, and generality to other LLM families remains to be established.

Third, all mechanistic analyses are restricted to the *dual-correct* pool, namely prompts that each model answers correctly on both the clean and corrupted runs. Retention rates differ across families (about 75% for *Qwen3* versus 35–40% for *Llama-3.1* and *Mistral*), so our claims describe the computation underlying *successful* propositional reasoning on these matched pools, not failure-case behavior or absolute cross-family difficulty.

Acknowledgments

This work was supported in part by the Beijing Major Science and Technology Project under Contract No. Z251100008125054. This work was supported by the Beijing Academy of Artificial Intelligence (BAAI). This work was supported by Yixin Group Limited. We gratefully acknowledge their provision of the essential computing resources required for our experiments.

References

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *FAccT '21: 2021 ACM Conference on Fairness, Accountability, and Transparency, Virtual Event / Toronto, Canada, March 3-10, 2021*, pages 610–623. ACM.

Leonard Bereska and Efstratios Gavves. 2024. Mechanistic interpretability for ai safety - a review. *ArXiv*, abs/2404.14082.

Fengxiang Cheng, Haoxuan Li, Fenrong Liu, Robert van Rooij, Kun Zhang, and Zhouchen Lin. 2025. Empowering llms with logical reasoning: A comprehensive survey. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2025, Montreal, Canada, August 16-22, 2025*, pages 10400–10408. ijcai.org.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Jun-Mei Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiaoling Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 179 others. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645:633 – 638.

Subhabrata Dutta, Joykirat Singh, Soumen Chakrabarti, and Tanmoy Chakraborty. 2024. How to think step-by-step: A mechanistic understanding of chain-of-thought reasoning. *Trans. Mach. Learn. Res.*, 2024.

Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Sean Welleck, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D. Hwang, Soumya Sanyal, Xiang Ren, Allyson Ettinger, Zaïd Harchaoui, and Yejin Choi. 2023. Faith and fate: Limits of transformers on compositionality. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Ahmed El-Kishky. 2024. Openai o1 system card. *ArXiv*, abs/2412.16720.

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, and 6 others. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2021/framework/index.html.

Jiahai Feng and Jacob Steinhardt. 2024. How do language models bind entities in context? In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Javier Ferrando and Elena Voita. 2024. Information flow routes: Automatically interpreting language models at scale. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 17432–17445. Association for Computational Linguistics.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 12216–12235. Association for Computational Linguistics.

Peter Hase, Mohit Bansal, Been Kim, and Asma Ghandeharioun. 2023. Does localization inform editing? surprising differences in causality-based localization vs. knowledge editing in language models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Stefan Heimersheim and Neel Nanda. 2024. How to use and interpret activation patching. *ArXiv*, abs/2404.15255.

Guan Zhe Hong, Nishanth Dikkala, Enming Luo, Cyrus Rashtchian, Xin Wang, and Rina Panigrahy. 2024. A implies b: Circuit analysis in llms for propositional logical reasoning. *Advances in Neural Information Processing Systems 38*.

Subhash Kantamneni and Max Tegmark. 2025. Language models use trigonometry to do addition. *CoRR*, abs/2502.00873.

Geonhee Kim, Marco Valentino, and André Freitas. 2024. Reasoning circuits in language models: A mechanistic interpretation of syllogistic inference. In *Annual Meeting of the Association for Computational Linguistics*.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson E. Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, John Kernion, Kamil.e Lukovsiut.e, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, and 11 others. 2023. Measuring faithfulness in chain-of-thought reasoning. *ArXiv*, abs/2307.13702.

Kenneth Li, Aspen K. Hopkins, David Bau, Fernanda B. Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023. Emergent world representations: Exploring a sequence model trained on a synthetic task. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems 35*.

Neel Nanda and Joseph Bloom. 2022. Transformerlens. `https://github.com/TransformerLensOrg/TransformerLens`.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. Progress measures for grokking via mechanistic interpretability. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, and 7 others. 2022. In-context learning and induction heads. *CoRR*, abs/2209.11895.

Debjit Paul, Robert West, Antoine Bosselut, and Boi Faltings. 2024. Making reasoning matter: Measuring and improving faithfulness of chain-of-thought reasoning. In *Conference on Empirical Methods in Natural Language Processing*.

Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeff Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Bloom, Stella Biderman, Adrià Garriga-Alonso, Arthur Conmy, Neel Nanda, Jessica Rumbelow, Martin Wattenberg, Nandi Schoots, Joseph Miller, Eric J. Michaud, and 10 others. 2025. Open problems in mechanistic interpretability. *ArXiv*, abs/2501.16496.

Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. 2023. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. In *Conference on Empirical Methods in Natural Language Processing*.

Curt Tigges, Michael Hanna, Qinan Yu, and Stella Biderman. 2024. LLM circuit analyses are consistent across training and scale. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart M. Shieber. 2020. Causal mediation analysis for interpreting neural NLP: the case of gender bias. *CoRR*, abs/2004.12265.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2023a. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 2023b. Label words are anchors: An information flow perspective for understanding in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 9840–9855. Association for Computational Linguistics.

Tingyu Xia, Bowen Yu, Yuan Wu, Yi Chang, and Chang Zhou. 2024. Language models can evaluate themselves via probability discrepancy. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, volume ACL 2024 of *Findings of ACL*, pages 4889–4901. Association for Computational Linguistics.

Jiaran Ye, Zijun Yao, Zhidian Huang, Liangming Pan, Jinxin Liu, Yushi Bai, Amy Xin, Weichuan Liu, Xiaoyin Che, Lei Hou, and Juanzi Li. 2025. How do transformers learn implicit reasoning? In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*.

Zeping Yu and Sophia Ananiadou. 2024. Interpreting arithmetic mechanism in large language models through comparative neuron analysis. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 3293–3306. Association for Computational Linguistics.

Fred Zhang and Neel Nanda. 2024. Towards best practices of activation patching in language models: Metrics and methods. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Appendix Roadmap

The appendix supplies (i) full dataset specification and prompt templates (§A), (ii) implementation details for the causal probes (§B), (iii) hop-wise and rule-wise behavioral baselines (§C), and three results sections that supplement the body experiments along the same backbone: staged computation (§D), information routing (§E), and specialized attention heads (§F). Each results section pools all models, prompt orders, and hops; the model-by-model comparison summary is in Table 7.

A Dataset Details

PropLogic-MI is released in two prompt-order variants (π_{FF} and π_{EF}) generated from the same seed

and the same underlying logical rows. Each file contains 20,000 unique clean/corrupted pairs and ends with the answer suffix `"Answer with one word only: True or False."` Table 3 lists the eleven rule categories and Table 4 illustrates the one-hop and two-hop prompt templates.

Generation, corruption, and filtering. For one-hop rows, we instantiate a symbolic rule template, randomly remap the template variables to fresh uppercase symbols (excluding `F`/`T`), sample a Boolean assignment, and evaluate the resulting expression. For two-hop rows, we additionally introduce an intermediate variable $M = e_1$ and a gate variable G, with the final query being either $M \wedge G$ or $M \vee G$. The generation budget targets a 50/50 hop split, but the uniqueness constraint under our corruption semantics yields a final 9,523 one-hop / 10,477 two-hop allocation that is deterministic across the two prompt-order variants. Corruption performs an exact label-flipping search over the factual variables only (19,808/20,000 rows); the remaining 192 one-hop rows whose expressions are constant under the available facts fall back to a label-flipping expression substitution. The raw dataset is not filtered by model behavior. Each mechanistic experiment uses the model-specific *dual-correct* subset, namely rows that the model answers correctly on both the clean and corrupted runs (Table 1). All rows carry rule, hop, prompt-order, and corruption metadata so that subsets can be reproduced from the released JSONL.

B Methodology and Implementation Details

B.1 Hardware and Software

All experiments run on NVIDIA H100 GPUs with 96 GB memory, using PyTorch 2.0 with CUDA 11.8. Causal probes are implemented with the TransformerLens library (Nanda and Bloom, 2022). All forward passes use `torch.no_grad()` with `float16` weights and activations. A single residual-stream token-by-layer patching pass takes about 30 s on *Qwen3-8B* (36 layers $\times \approx$16 token positions); a single head-by-layer mean-ablation pass takes about 2 min (36 layers $\times$32 heads). The full paper-scale pipeline (all four models, both prompt orders, all interventions) takes roughly 50 GPU-hours, excluding replotting from saved CSV/JSON/PKL artifacts.

B.2 Probability-Difference Metric

All three intervention branches score with the same probability-difference metric introduced in §4. For an answer label $y \in \{\texttt{True}, \texttt{False}\}$ and a softmax restricted to those two answer tokens, we define

$$\mathrm{PD}(x) = p_\theta(\texttt{True} \mid x) - p_\theta(\texttt{False} \mid x), \quad (4)$$

and its label-aligned form $\mathrm{PD}_{\mathrm{label}}(x, y) = \mathrm{PD}(x)$ if $y = \texttt{True}$, else $-\mathrm{PD}(x)$, so larger values always indicate stronger support for the correct answer. For an intervention producing $\tilde{x}$, the shift is

$$\mathrm{dPD} = \mathrm{PD}_{\mathrm{label}}(\tilde{x}, y) - \mathrm{PD}_{\mathrm{label}}(x_{\mathrm{baseline}}, y), \quad (5)$$

where x_{baseline} is the corrupted prompt for activation patching and the clean prompt for mean ablation. Throughout the paper, we report this shift rather than raw logit differences.

B.3 Layer Partition and Sample Budgets

For Early/Middle/Late summaries we use fixed token-level boundaries Early $= [0, 13]$, Middle $= [14, 23]$, Late $= [24, L-1]$ for the *Qwen3* family and *Mistral-7B* (with $L \in \{36, 40, 32\}$). For *Llama-3.1-8B* ($L = 32$), whose staged computation peaks earlier in depth, we use Early $= [0, 8]$, Middle $= [9, 15]$, Late $= [16, L-1]$ so that the labels remain aligned with the model's empirical phases. The logical-block analyses split the full depth into three approximately equal contiguous spans with `numpy.array_split`. Sample budgets per setting are: 3000 rows for logical-block ablation, 1000 rows for token patching, 2000 rows for head discovery / classification, 1000 rows for the head-validation dPD curves, and 500 rows for the head-validation accuracy tables.

B.4 Intervention Scopes

Logical-block ablation (MLP / Attention / Joint). For a logical block $R \in \{\text{Facts}, \text{Expression}, \text{Query}\}$ and layer l, the **MLP block ablation** runs the clean prompt and replaces `mlp_out` for all positions $i \in R$ with the within-sequence mean MLP vector at that layer:

$$\tilde{m}_i^{(l)} = \frac{1}{T} \sum_{j=1}^{T} m_j^{(l)}, \qquad i \in R, \quad (6)$$

where T is the sequence length. The **attention block ablation** applies the same within-sequence

Category	Formal Definitions	Example Prompt Templates (One-Hop)
Identity	$P \wedge \top \equiv P, \quad P \vee \bot \equiv P$	"A is T, A and T is", "A is F, A or F is"
Domination	$P \wedge \bot \equiv \bot, \quad P \vee \top \equiv \top$	"A is T, A and F is", "A is F, A or T is"
Idempotent	$P \wedge P \equiv P, \quad P \vee P \equiv P$	"A is T, A and A is", "A is F, A or A is"
Dbl. Negation	$\neg(\neg P) \equiv P$	"A is T, (¬(¬A)) is"
Excluded Mid.	$P \vee \neg P \equiv \top$	"A is T, A or ¬A is"
Contradiction	$P \wedge \neg P \equiv \bot$	"A is F, A and ¬A is"
Commutative	$P \wedge Q \equiv Q \wedge P$	"A is T, B is F, A and B is"
Associative	$(P \wedge Q) \wedge R \equiv P \wedge (Q \wedge R)$	"A is T, B is T, C is F, (A and B) and C is"
Distributive	$P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$	"A is T, B is F, C is T, A and (B or C) is"
De Morgan	$\neg(P \wedge Q) \equiv \neg P \vee \neg Q$	"A is T, B is F, (¬A or ¬B) is"
Absorption	$P \wedge (P \vee Q) \equiv P$	"A is T, B is F, A and (A or B) is"

Table 3: **Rule coverage.** The 11 rule categories in *PropLogic-MI*, each with its formal Boolean-algebra definition and an illustrative one-hop template. Placeholder letters {A,B,C} are compact stand-ins; in the actual corpus, variables are randomly remapped to fresh uppercase symbols (excluding T/F).

Depth	Prompt Template & Reasoning Process
One-Hop	"A is True, B is False, (A and B) is" ↪ *Logic:* $A \wedge B$
	"A is True, (¬(¬A)) is" ↪ *Logic:* $\neg(\neg A) \equiv A$
	"A is True, B is False, (¬A or ¬B) is" ↪ *Logic: De Morgan* $(\neg A \vee \neg B)$
Two-Hop	"A is T, B is A and T, C is F, B and C is" ↪ *Chain:* $(A \wedge \top) \to B; \quad (B \wedge C) \to$ *Ans*
	"A is T, B is F, C is ¬(A and B), D is T, C or D is" ↪ *Chain: NAND*$(A, B) \to C; \quad (C \vee D) \to$ *Ans*
	"A is T, B is F, C is A and B, D is T, C or D is" ↪ *Chain:* $(A \wedge B) \to C; \quad (C \vee D) \to$ *Ans*

Table 4: **Probing prompts across reasoning depths.** One-hop queries apply a rule directly; two-hop queries require an inline intermediate assignment $M = e_1$. Each logical row is realized under both π_{FF} and π_{EF}.

mean replacement to attn_out instead. The **joint attention+MLP block ablation** replaces both attn_out and mlp_out inside R during one forward pass. All three branches reuse the same three-block parser (logical blocks are defined by character-span alignment and then mapped back to token indices) and the same dPD scoring.

Token-wise residual patching. For each clean/corrupted pair we patch resid_pre one layer and one token at a time from the clean prompt into the corrupted prompt, recording the layer×token matrix $\mathrm{dPD}_{l,i}$. Raw prompt positions are first aligned to model-input positions using tokenizer offset mappings; when offset-based alignment fails, the implementation falls back to a contiguous body-span mapping inside the model input. Tokens are then assigned to a prompt-order-sensitive taxonomy: fact_value, fact_var_ref, query_last, operator, expr_last, new_var_decl, new_val_ref, others. The classifier is prompt-order-sensitive because the same lexical token plays different structural roles under π_{FF} vs. π_{EF} (*e.g.*, the query token is the first is in π_{EF} and the last is in π_{FF}). For each category c and stage $s \in \{\mathrm{Early}, \mathrm{Middle}, \mathrm{Late}\}$, the per-token causal density reported in figures is

$$M_c^{(s)} = \frac{1}{|c|} \sum_{i \in c} \frac{1}{|s|} \sum_{l \in s} |\mathrm{dPD}_{l,i}|, \tag{7}$$

averaged across samples. The $|c|$ normalization controls for category size, so frequent token classes do not dominate merely because they occupy more positions.

Attention-head mean ablation and role classification. We replace the output $z_{l,h}$ of head h at layer l by its dataset mean computed over a balanced clean sample set, at all token positions. Heads are ranked by

$$\mathrm{Sel}(l, h) = -\mathbb{E}[\mathrm{dPD}_{\mathrm{label}}(l, h)], \tag{8}$$

so larger values correspond to heads whose ablation most strongly reduces correct-answer support. We first run a small probe over all heads, keep the top candidates globally together with the top heads from each layer, and then re-evaluate the candidate pool on a larger balanced set.

Retained heads are classified into Splitting (SP), Transmission (TR), and Fact-Retrieval (FR) using six attention-pattern statistics measured on clean prompts: fact-focus ratio f, boundary-focus ratio b, intra-clause ratio i, attention entropy e, top-1

mass t, and split-half stability s. After feature standardization, each head receives a rule-based score for each role,

$$\begin{aligned}\text{Score}_{\text{FR}} = f - 0.20\,i - 0.15\,b + 0.20\,t \\ - 0.10\,e + 0.08\,s,\end{aligned} \tag{9}$$

$$\begin{aligned}\text{Score}_{\text{SP}} = b - 0.15\,i - 0.10\,f + 0.15\,t \\ - 0.08\,e + 0.10\,s,\end{aligned} \tag{10}$$

$$\begin{aligned}\text{Score}_{\text{TR}} = i - 0.20\,b - 0.10\,f + 0.18\,t \\ - 0.08\,e + 0.12\,s,\end{aligned} \tag{11}$$

and the winning label is assigned. The Splitting / Transmission / Fact-Retrieval taxonomy used throughout the paper is the output of this classifier.

For validation, we mean-ablate the top-k heads of each role on held-out clean prompts and report both post-ablation accuracy and the mean dPD shift. The *Random-Outside* control samples heads from layers used in the validation stage but outside the selected head pool; *Mixed-All* ablates the union of the top heads from all three roles.

B.5 Reproducibility

Dataset construction is handled by `src.data.generate_dataset`; inference, dual-correct filtering, and metrics by `src.eval`; block ablations by `src.mlp_analysis`, `src.attn_analysis`, and `src.attn_mlp_analysis`; token patching by `src.token_analysis`; head discovery and validation by `src.heads_analysis`. Wrapper scripts in `scripts/` run the per-model pipelines, while `scripts/replot_all_figures.sh` regenerates figures from saved report files. All dataset generation and sampling entry points expose a `-seed` argument with default 42. Generated datasets, predictions, filtered pools, reports, figures, and logs are written to ignored output directories. Reproducing the *Llama-3.1* runs requires the usual Hugging Face access to the gated model weights.

C Behavioral Performance on PropLogic-MI

Table 5 reports overall clean and corrupted accuracy, dual-correct rate, and hop-wise clean accuracy for the two *Qwen3* models that anchor the main text. Table 6 breaks down clean accuracy by rule. The same statistics for *Llama-3.1-8B-Instruct* and *Mistral-7B-Instruct* are reported alongside their mechanistic profile in Table 7.

D Supplementary Results: Staged Computation

This section supplements §5.1 along the three ablation pathways used in the body. Within each pathway we show all available model, prompt-order, and hop conditions so that the staged template can be read off the same pathway across settings, rather than per family.

D.1 MLP-Block Ablation

Figures 7–14 show MLP-block mean-ablation dPD for all four models under both prompt orders, following the body figure template (Figure 3 in §5.1). The early-Facts / middle-Expression / middle-to-late-Query order persists across scales, prompt orders, reasoning depths, and model families. *Qwen3* produces the sharpest stage boundaries; *Llama* and *Mistral* preserve the same coarse ordering while allocating a heavier Query Block throughout depth.

D.2 Attention-Block Ablation

Figures 15–22 show attention-block mean-ablation dPD in the same format. Compared with the MLP branch, the attention branch shifts more mass into middle-layer Expression and later Query integration, consistent with attention's role in cross-token routing. The coarse staged ordering is preserved across all (model, prompt-order) cells.

D.3 Joint Attention+MLP Block Ablation

Figures 23–29 show the joint attention+MLP-block mean-ablation dPD in the same format. The body Figure 3 is the *Qwen3-8B* π_{FF} cell of this grid. The joint branch preserves the same coarse staged ordering as the MLP branch while sharpening early-Facts dominance.

E Supplementary Results: Information Routing

This section supplements §5.2 with the full cross-model per-token causal-density panels. All figures use the per-token aggregation $M_c^{(s)}$ defined in Eq. 7; layers are partitioned into Early/Middle/Late per family as in §B.3. Two robust points emerge.

Boundary-token convergence generalizes. Figures 30–31 (under π_{FF}) and Figures 34–35 (under π_{EF}) show the same boundary-token transport in all four families: `fact_value` dominates the early layers, `expr_last` (where present) dominates the

Model	Prompt Order	Clean	Corrupted	Dual-Correct Rate	One-Hop	Two-Hop
Qwen3-8B	π_{FF}	86.98	88.17	75.44% (15087/20000)	86.35	87.55
Qwen3-14B	π_{FF}	87.42	89.11	76.55% (15310/20000)	92.64	82.68
Qwen3-8B	π_{EF}	74.22	79.25	53.89% (10777/20000)	79.09	69.79
Qwen3-14B	π_{EF}	74.39	77.91	52.80% (10561/20000)	88.34	61.72

Table 5: **Overall zero-shot performance of Qwen3 on *PropLogic-MI*.** All values except the dual-correct column are accuracies (%). The dual-correct subset defines the candidate pool for the mechanistic analyses.

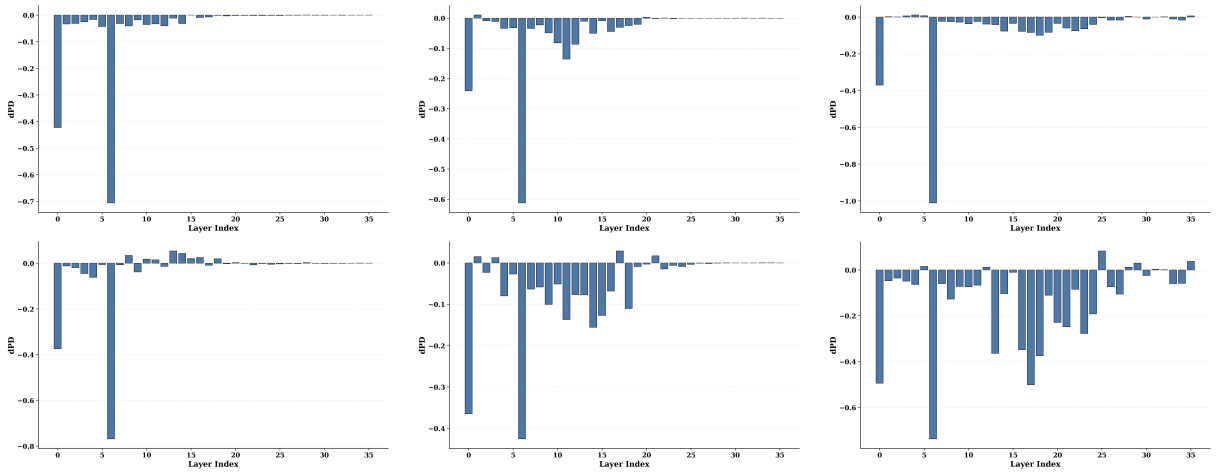


Figure 7: **MLP-block mean ablation on *Qwen3-8B* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

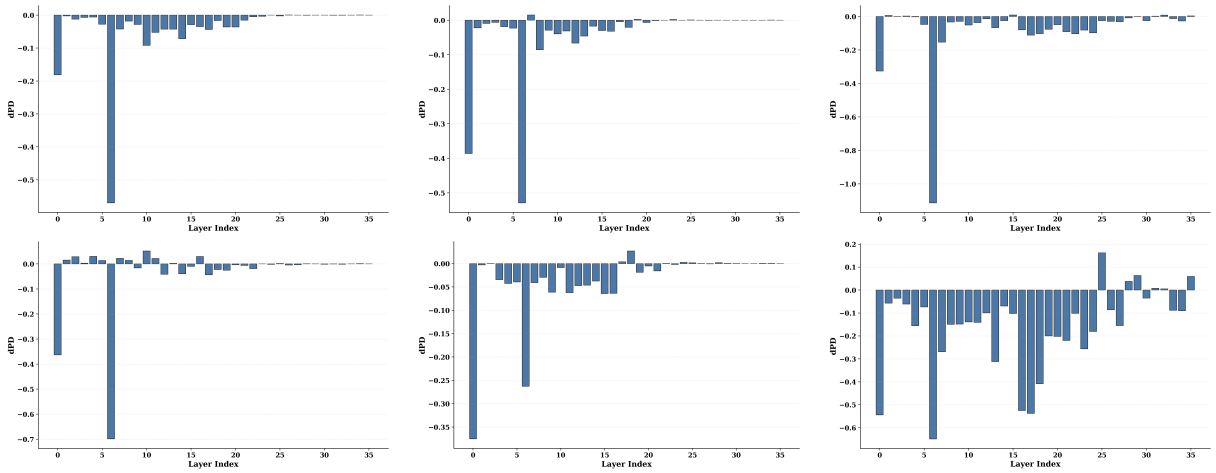


Figure 8: **MLP-block mean ablation on *Qwen3-8B* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

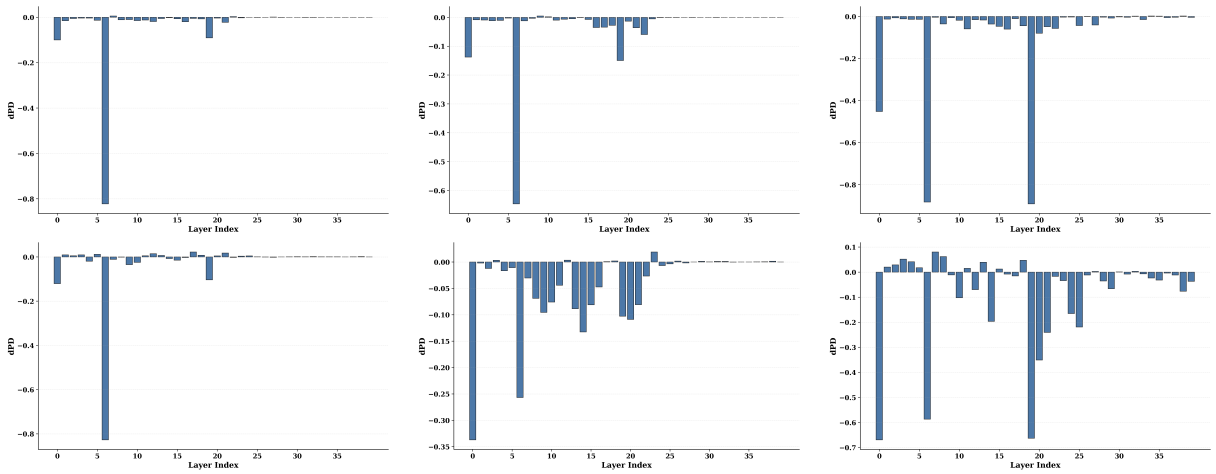


Figure 9: **MLP-block mean ablation on *Qwen3-14B* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

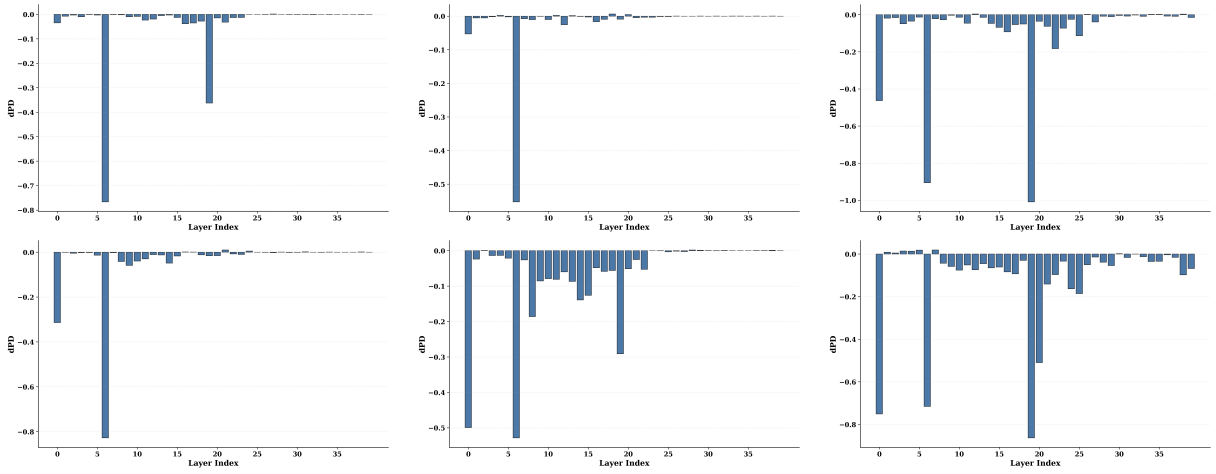


Figure 10: **MLP-block mean ablation on *Qwen3-14B* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

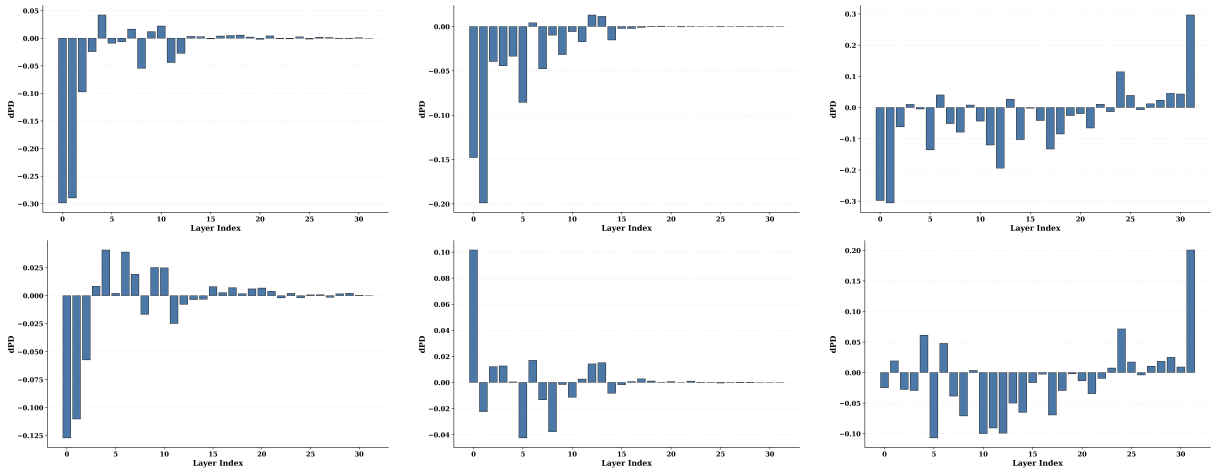


Figure 11: **MLP-block mean ablation on *Llama-3.1-8B-Instruct* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

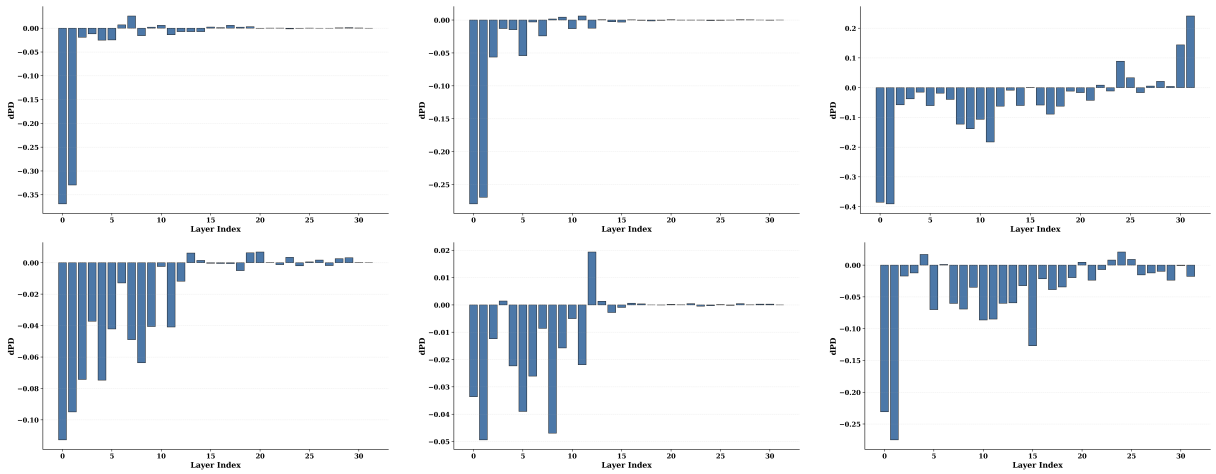


Figure 12: **MLP-block mean ablation on *Llama-3.1-8B-Instruct* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

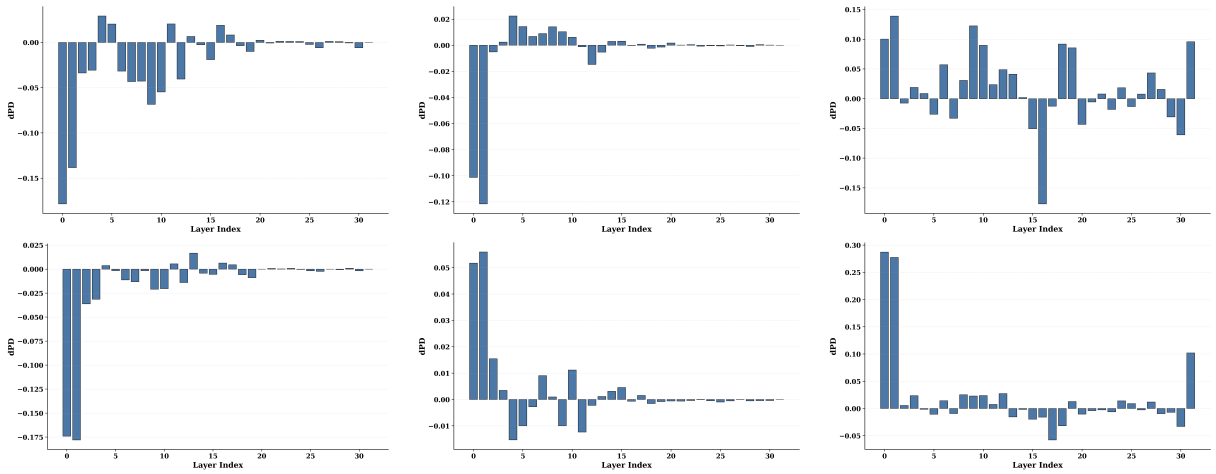


Figure 13: **MLP-block mean ablation on *Mistral-7B-Instruct* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

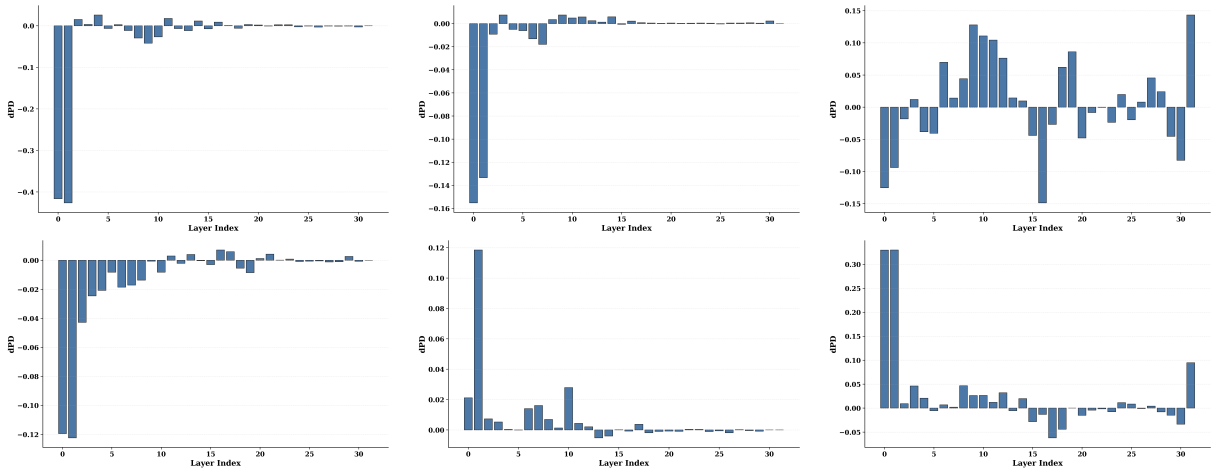


Figure 14: **MLP-block mean ablation on *Mistral-7B-Instruct* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

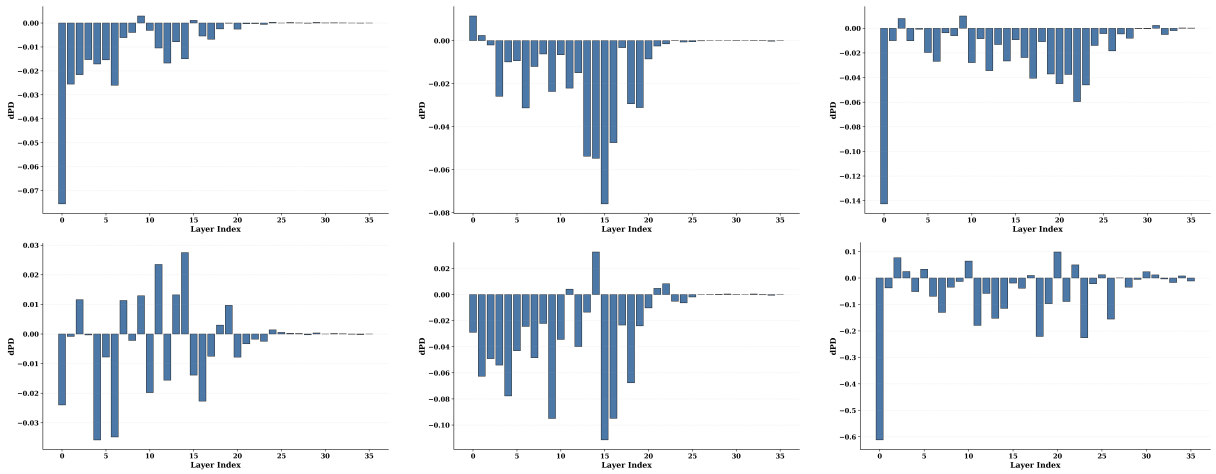


Figure 15: **Attention-block mean ablation on *Qwen3-8B* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

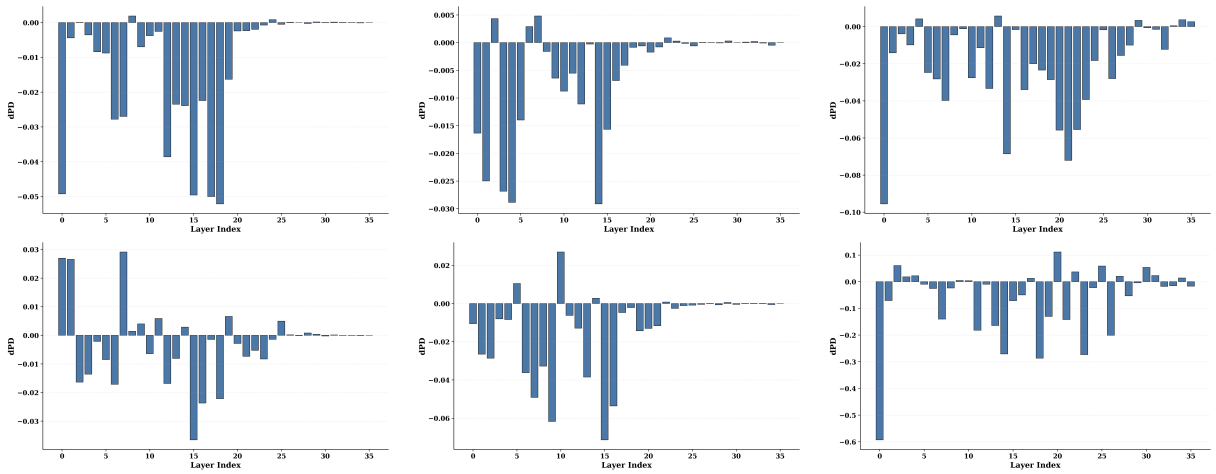


Figure 16: **Attention-block mean ablation on *Qwen3-8B* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

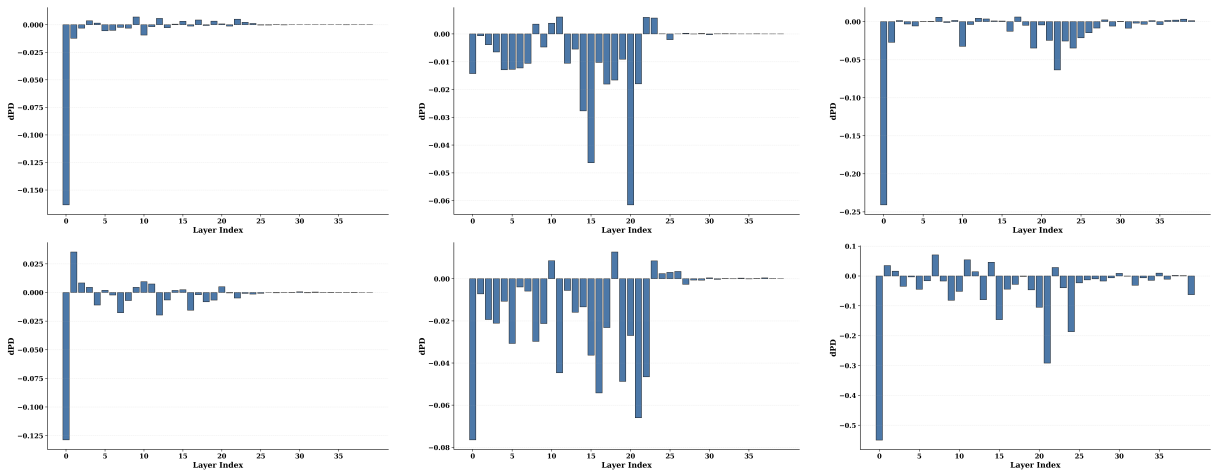


Figure 17: **Attention-block mean ablation on *Qwen3-14B* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

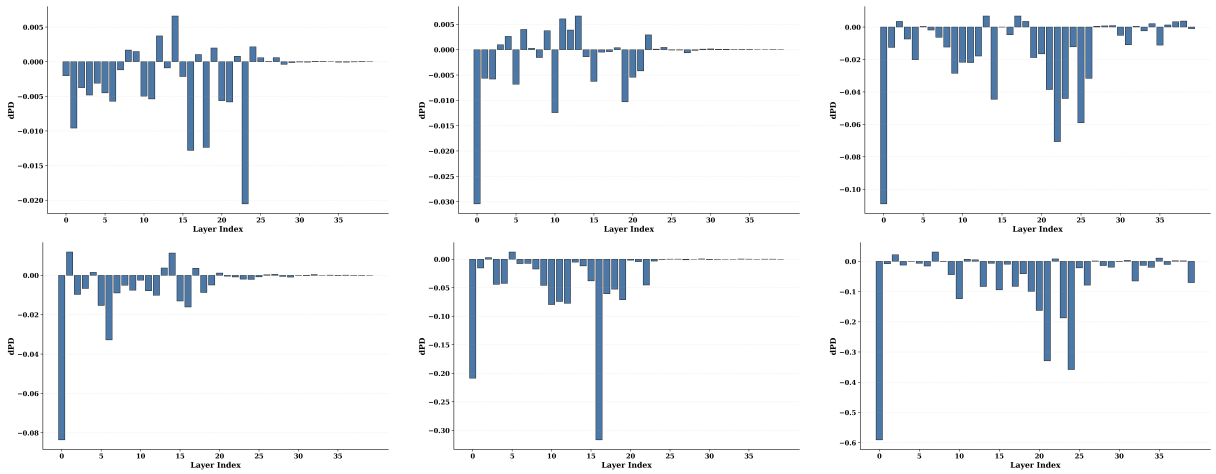


Figure 18: **Attention-block mean ablation on *Qwen3-14B* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

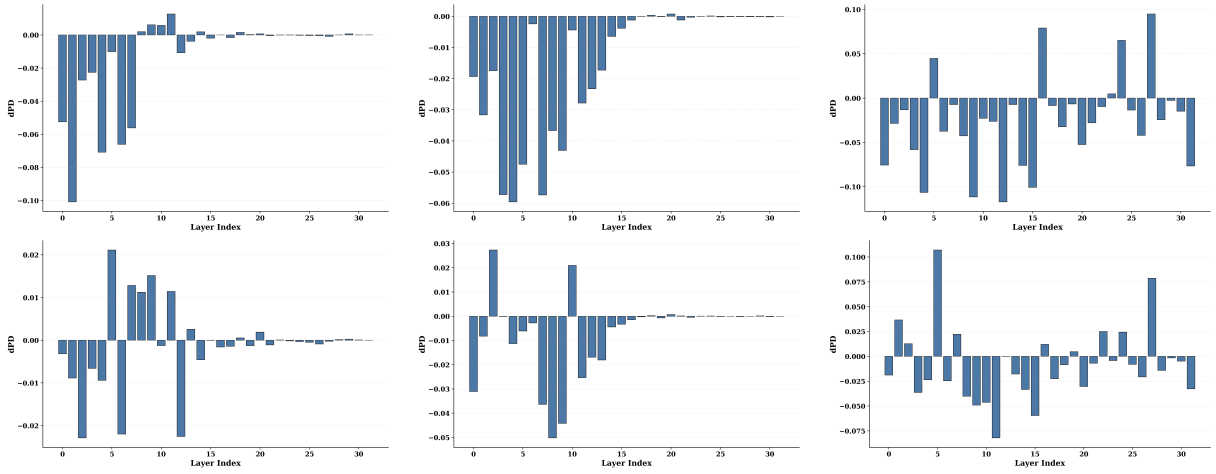


Figure 19: **Attention-block mean ablation on *Llama-3.1-8B-Instruct* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

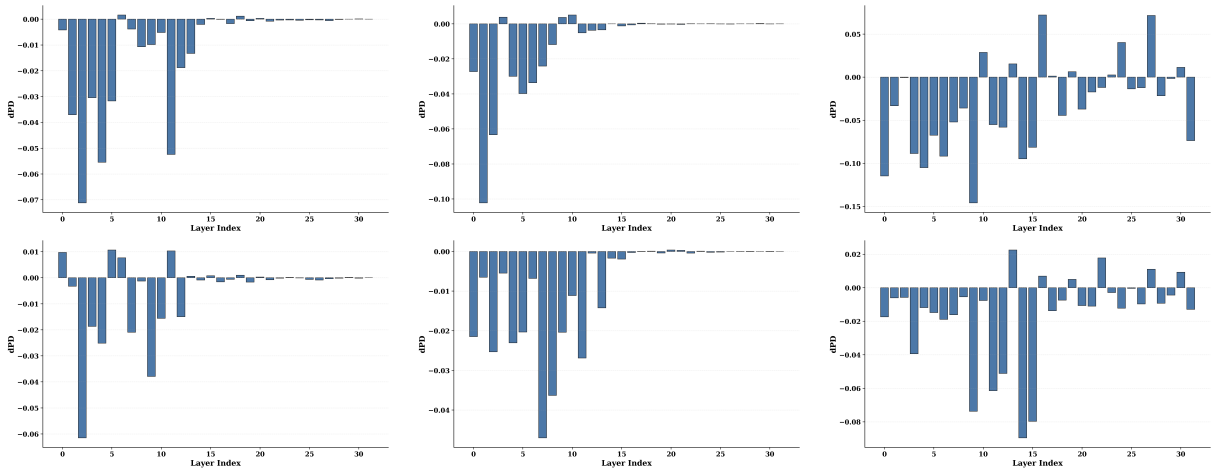


Figure 20: **Attention-block mean ablation on *Llama-3.1-8B-Instruct* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

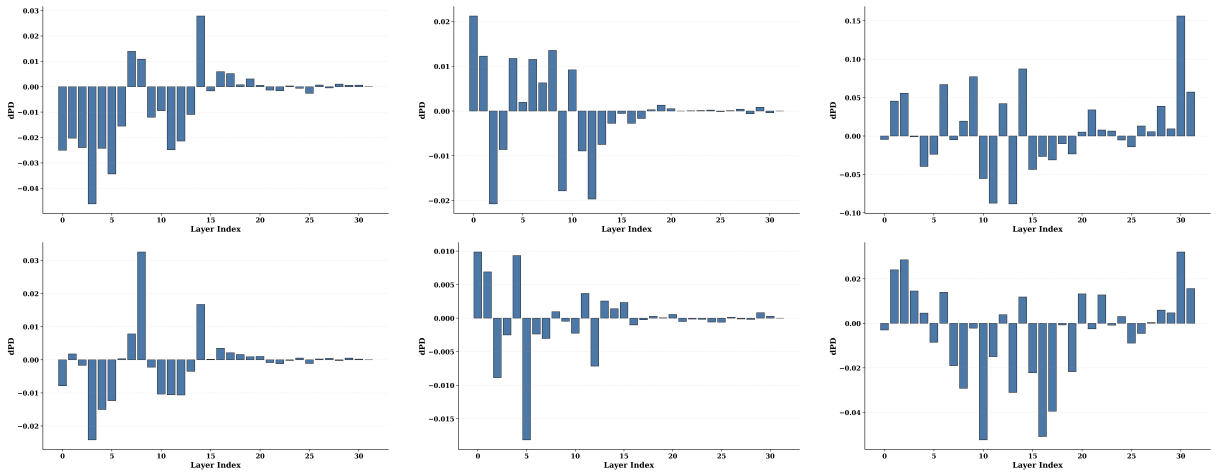


Figure 21: **Attention-block mean ablation on *Mistral-7B-Instruct* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

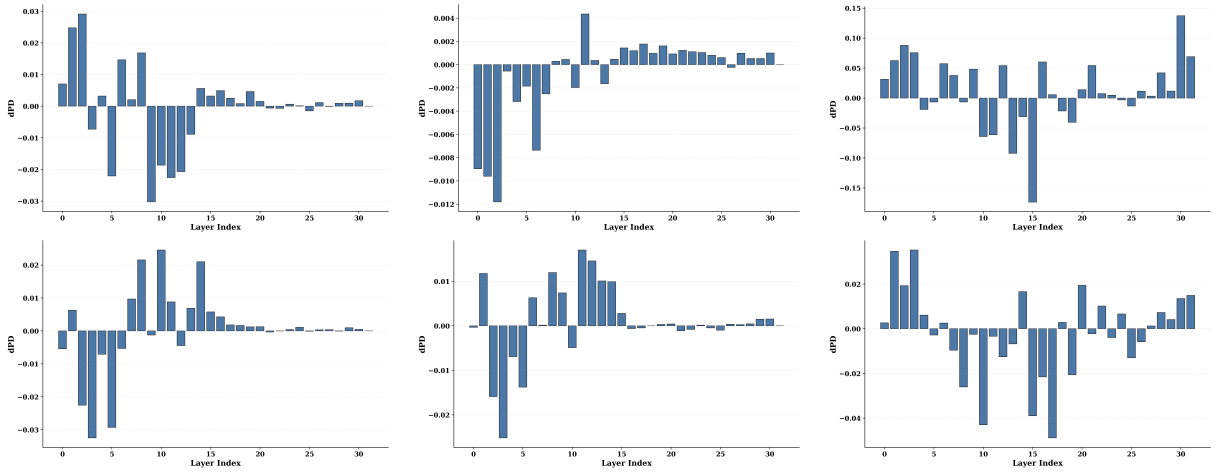


Figure 22: **Attention-block mean ablation on *Mistral-7B-Instruct* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

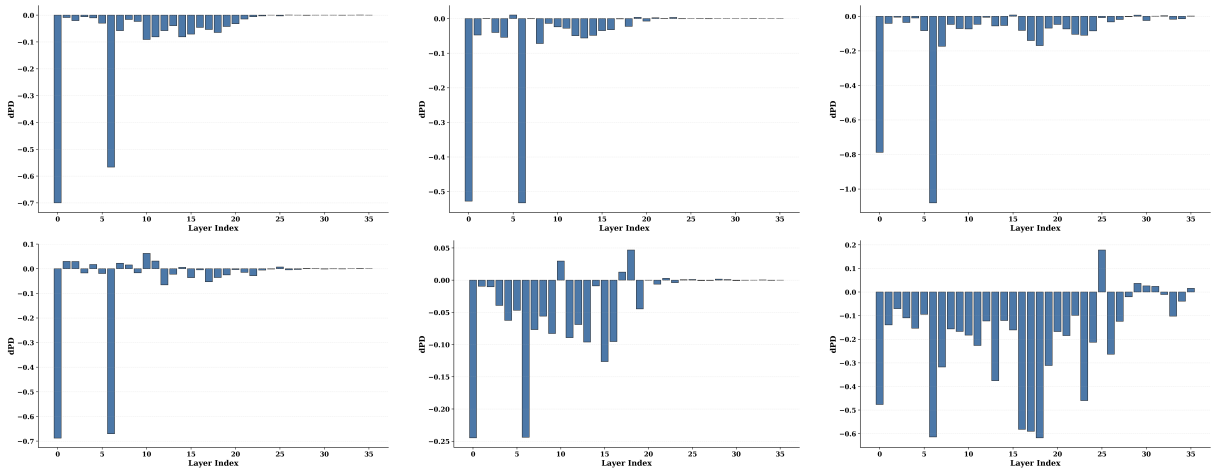


Figure 23: **Joint attention+MLP-block mean ablation on *Qwen3-8B* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

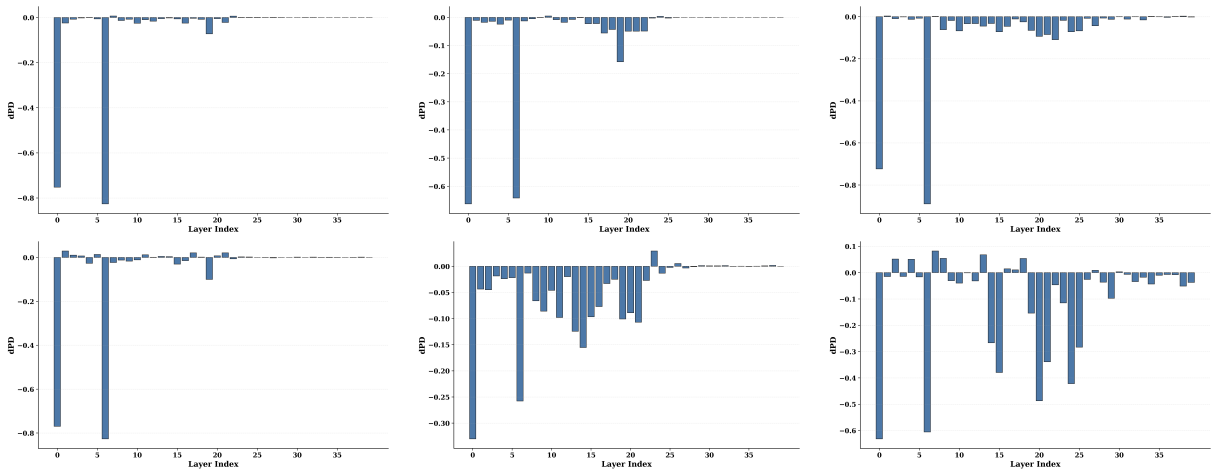


Figure 24: **Joint attention+MLP-block mean ablation on *Qwen3-14B* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

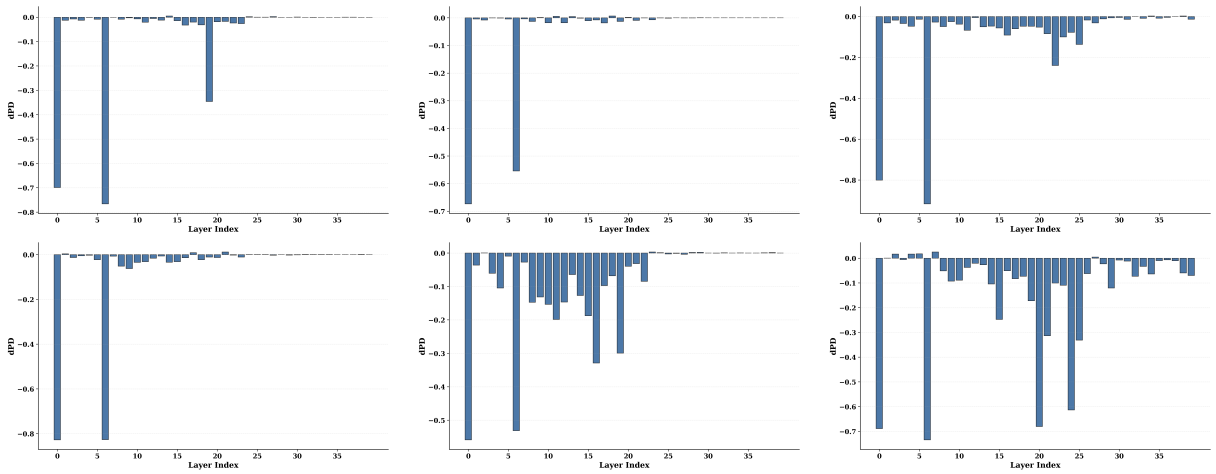


Figure 25: **Joint attention+MLP-block mean ablation on *Qwen3-14B* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

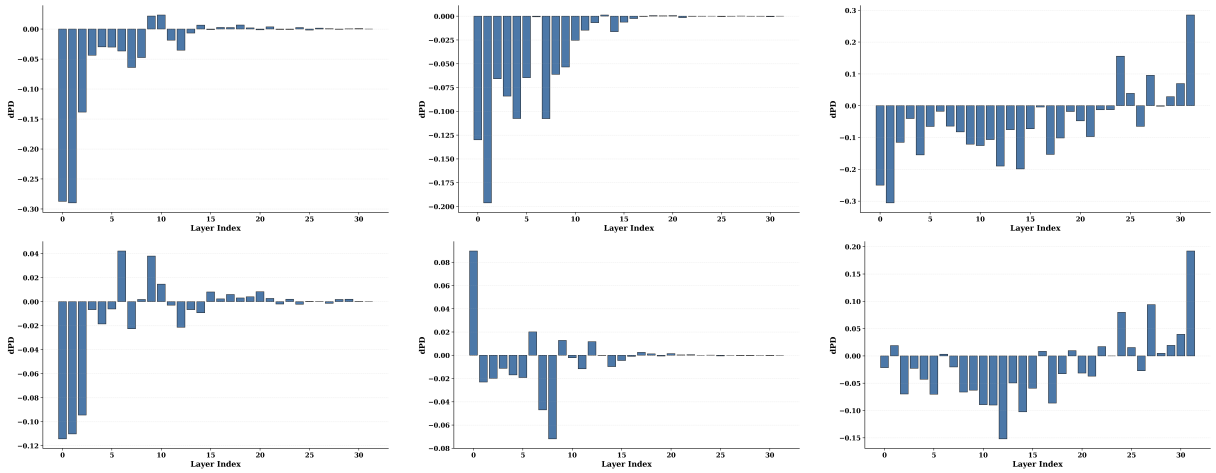


Figure 26: **Joint attention+MLP-block mean ablation on *Llama-3.1-8B-Instruct* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

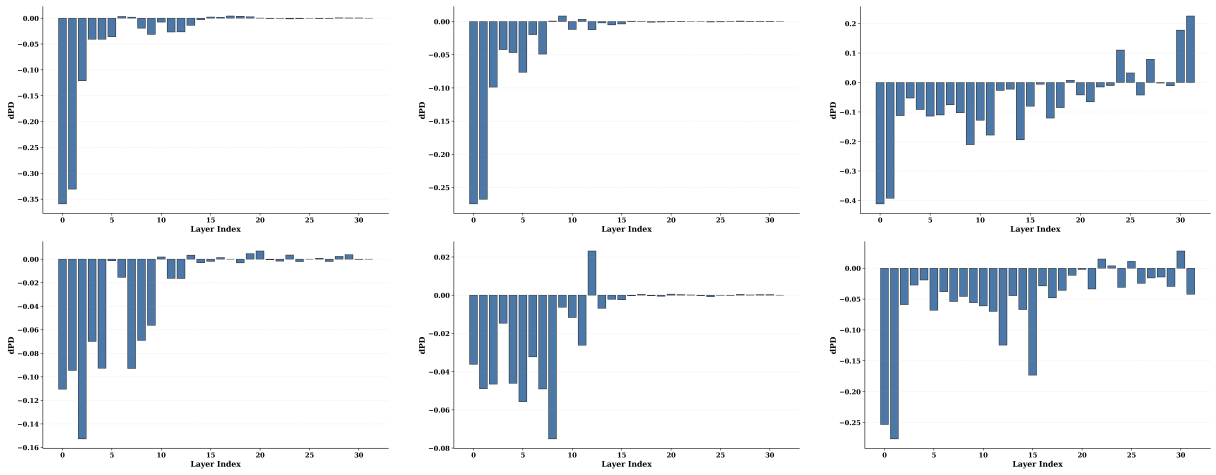


Figure 27: **Joint attention+MLP-block mean ablation on *Llama-3.1-8B-Instruct* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

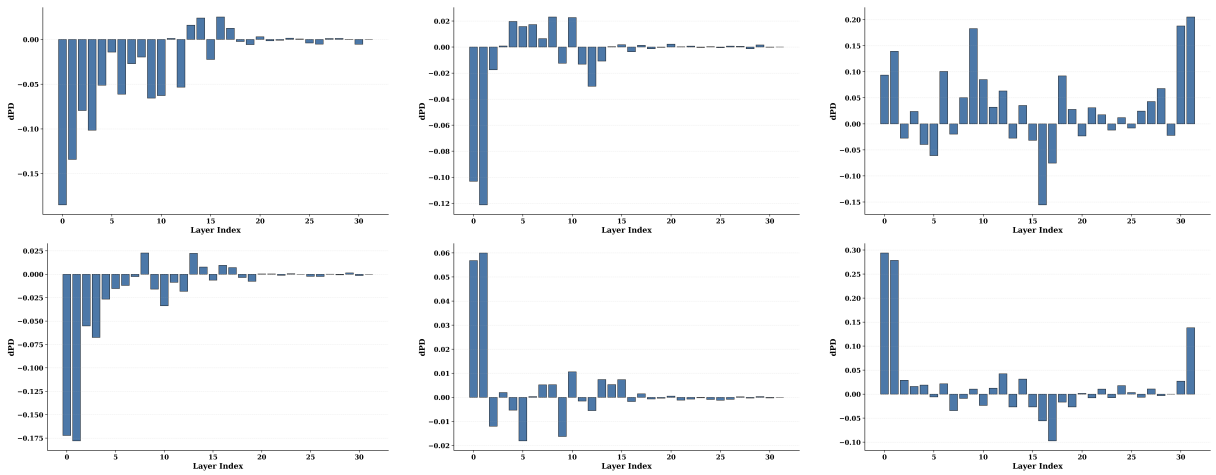


Figure 28: **Joint attention+MLP-block mean ablation on *Mistral-7B-Instruct* under π_{FF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

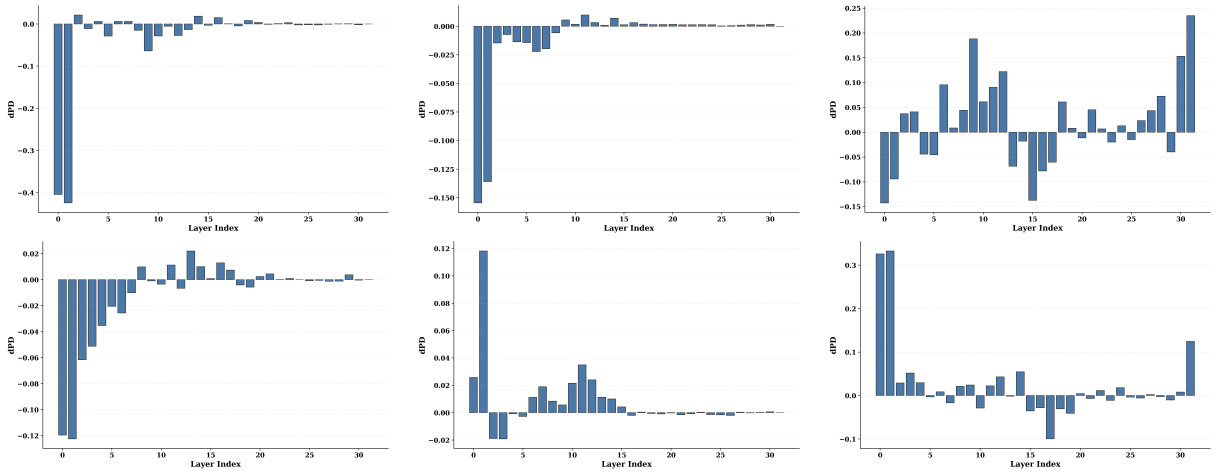


Figure 29: **Joint attention+MLP-block mean ablation on *Mistral-7B-Instruct* under π_{EF}.** Per-layer dPD when each logical block is mean-ablated. Top row: one-hop. Bottom row: two-hop. Columns: Facts / Expression / Query.

Rule	8B FF	14B FF	8B EF	14B EF
Absorption	84.8	95.0	84.4	85.4
Associative	92.1	99.0	61.8	76.0
Commutative	99.8	98.1	92.0	94.4
Contradiction	64.0	54.1	49.4	48.1
De Morgan	80.4	100.0	80.9	100.0
Distributive	72.8	71.5	76.8	88.8
Domination	95.9	88.6	62.5	50.3
Double Negation	84.8	87.1	72.7	70.3
Excluded Middle	98.7	86.1	72.9	51.4
Idempotent	92.1	85.6	83.5	76.5
Identity	91.4	96.7	79.4	77.2

Table 6: **Rule-wise clean accuracy (%) of Qwen3 on *PropLogic-MI*.** Prompt order has a large impact on several rules (Associative, Domination, Excluded Middle), while Commutative remains consistently easy and Contradiction consistently difficult.

middle layers, and `query_last` dominates the late layers. Under π_{EF}, `expr_last` is pinned to zero by autoregressive construction, so the middle-layer carrier shifts to `fact_value`; the early-fact and late-query dominance survive across families.

Middle-layer binding flow. The zoom views in Figures 32–33 and Figures 36–37 (`query_last` excluded) make the middle-layer exchange between `fact_value` and `fact_var_ref` visible. The signal is strongest under π_{EF} two-hop: *Qwen3-14B* reaches a `fact_var_ref` middle-layer peak around $\mathrm{dPD} \approx 0.41$ that overtakes `fact_value`; *Qwen3-8B* shows a smaller but clearly above-floor peak (≈ 0.15); *Llama-3.1* and *Mistral* retain the same shape with weaker magnitudes. The complementary late-layer `fact_value` persistence is most visible in the harder hop $\times$ family conditions (two-hop in *Llama-3.1* and *Mistral*), consistent with the late-layer sub-pattern reported in §5.2.2.

F Supplementary Results: Specialized Attention Heads

This section supplements §5.3.1. We first show the representative attention patterns of the three head roles (Figure 38), then per-model 4-panel layouts of the layer-wise role distributions and validation curves under both prompt orders (Figures 39–42). Numerical role counts: under π_{FF}, $66/129/175$ (Splitting/Transmission/Fact-Retrieval) in *Qwen3-8B* and $87/169/217$ in *Qwen3-14B*; under π_{EF}, $116/114/167$ and $170/142/200$. *Llama* yields 323 classified heads under π_{FF} $(119/106/98)$ and 377 under π_{EF} $(139/146/92)$; *Mistral* yields 314 under π_{FF} $(118/99/97)$ and 307 under π_{EF} $(123/109/75)$. The Splitting / Transmission / Fact-Retrieval taxonomy is recoverable in every cell; the family-level differences lie in the sharpness of the validation curves, not in the existence of the roles.

Model	FF Clean/Dual	EF Clean/Dual	Token-level profile	Head-ablation regime
Qwen3-8B	87.0 / 75.4	74.2 / 53.9	Most fact-centered at 8B scale: `fact_value` competitive with or above `query_last` in the per-token mean view.	Cleanly selective: role-targeted ablations are strongly harmful while *Random-Outside* stays near baseline through moderate k.
Qwen3-14B	87.4 / 76.6	74.4 / 52.8	Preserves the *Qwen3* fact-centered profile with cleaner middle-layer binding-flow exchange under π_{EF} two-hop.	Cleanly selective; the strongest Fact-Retrieval + Transmission coupling under π_{EF} two-hop.
Llama-3.1-8B	63.9 / 38.4	60.3 / 37.0	More query-centered and diffuse; `query_last` dominates `fact_value` under both prompt orders.	Weakly selective: role labels recoverable, but role-targeted ablations collapse to a common accuracy floor by moderate k.
Mistral-7B	65.1 / 35.1	66.7 / 40.0	Intermediate between *Qwen3* and *Llama*; less fact-centered than *Qwen3* but not as uniformly query-dominant as *Llama*.	Globally sensitive: role-targeted ablations are severe, but *Random-Outside* is also substantially damaging.

Table 7: **Cross-family summary.** Clean/Dual columns report clean accuracy and dual-correct rate (%) under π_{FF} (FF) and π_{EF} (EF). The mechanistic columns are detailed in §D–§F.

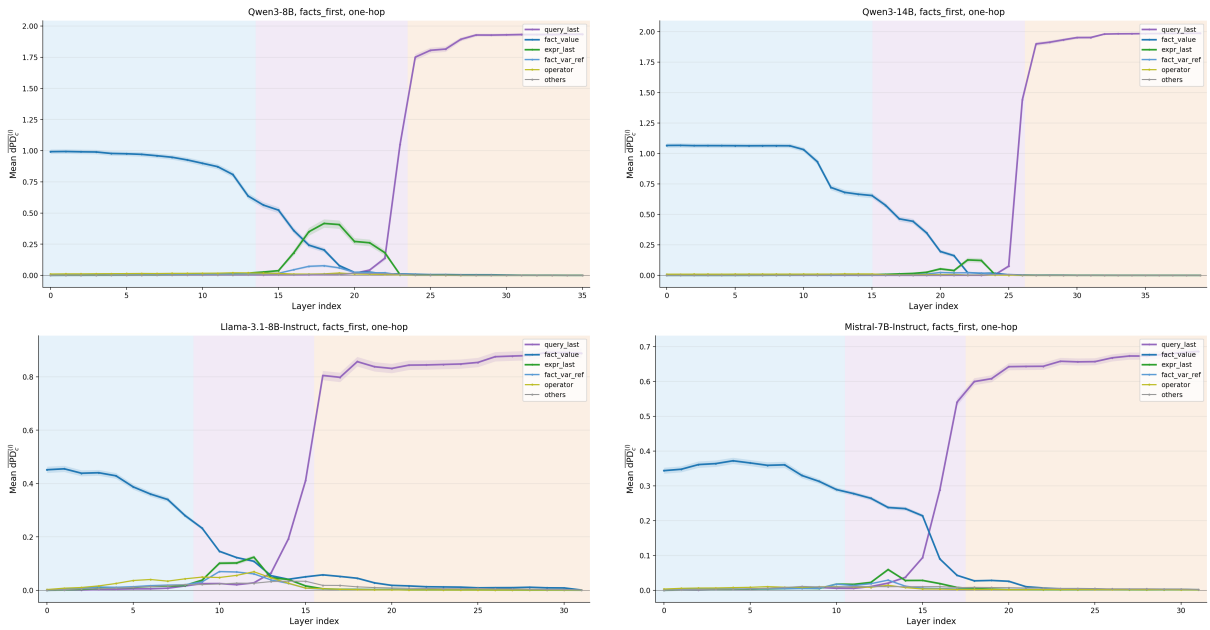


Figure 30: **Per-token mean** dPD**,** π_{FF}**, one-hop, full view.** Top-left: *Qwen3-8B*; top-right: *Qwen3-14B*; bottom-left: *Llama-3.1-8B-Instruct*; bottom-right: *Mistral-7B-Instruct*.

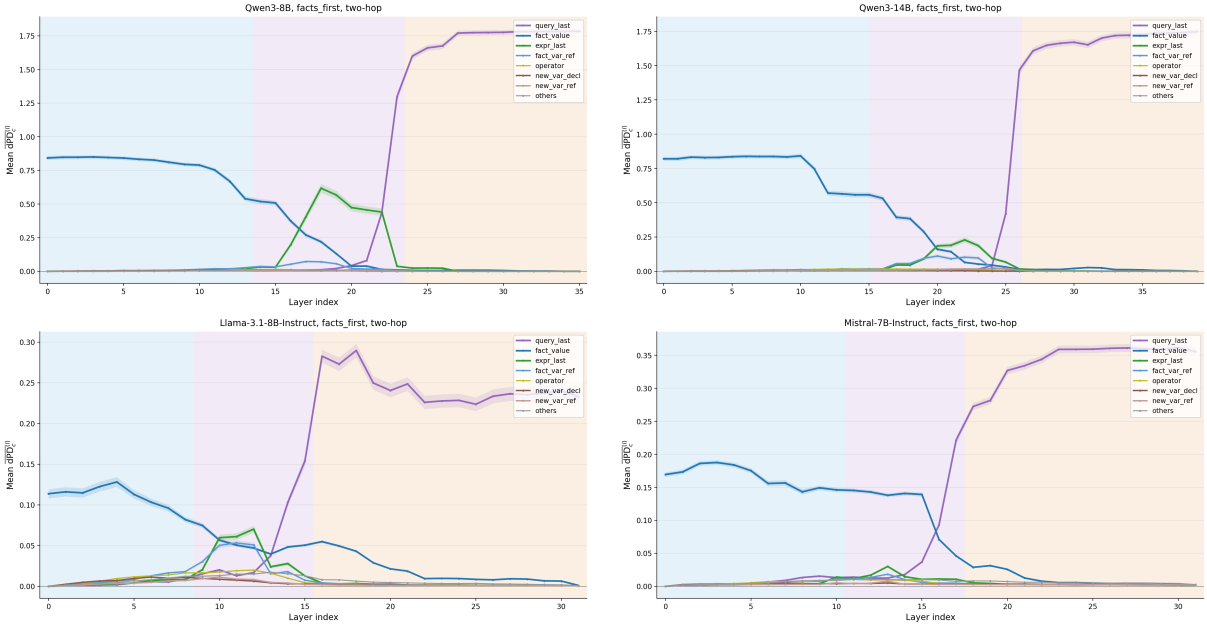


Figure 31: **Per-token mean** dPD**,** π_{FF}**, two-hop, full view.** Top-left: *Qwen3-8B*; top-right: *Qwen3-14B*; bottom-left: *Llama-3.1-8B-Instruct*; bottom-right: *Mistral-7B-Instruct*.

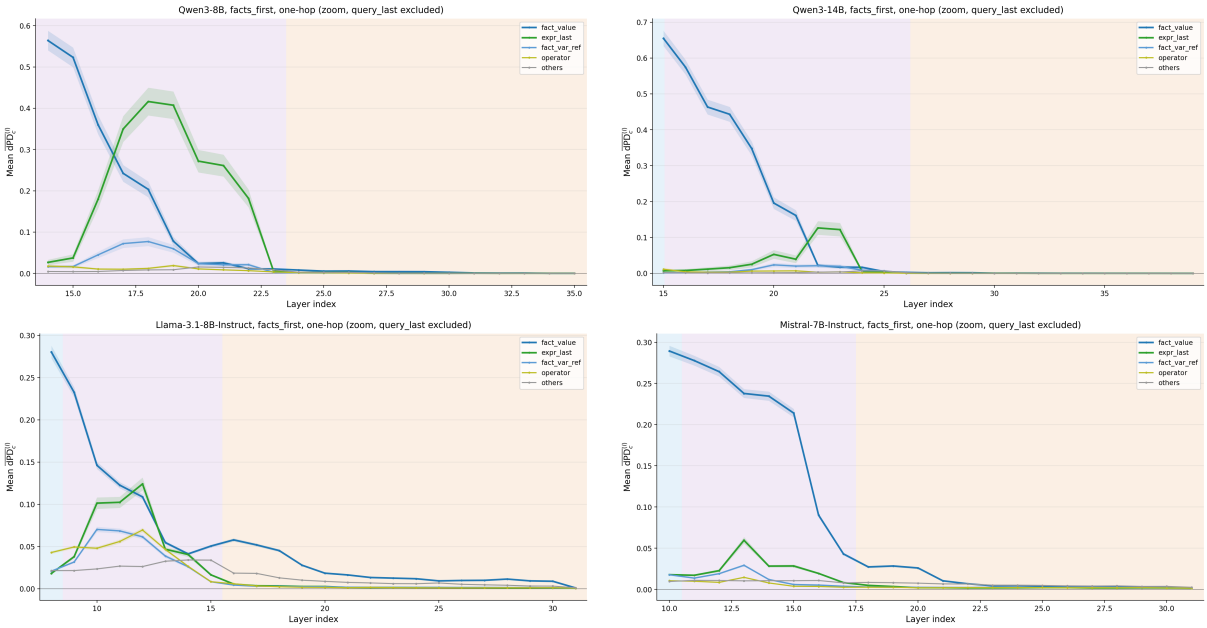


Figure 32: **Per-token mean** dPD**,** π_{FF}**, one-hop,** `query_last` **excluded.** Top-left: *Qwen3-8B*; top-right: *Qwen3-14B*; bottom-left: *Llama-3.1-8B-Instruct*; bottom-right: *Mistral-7B-Instruct*.

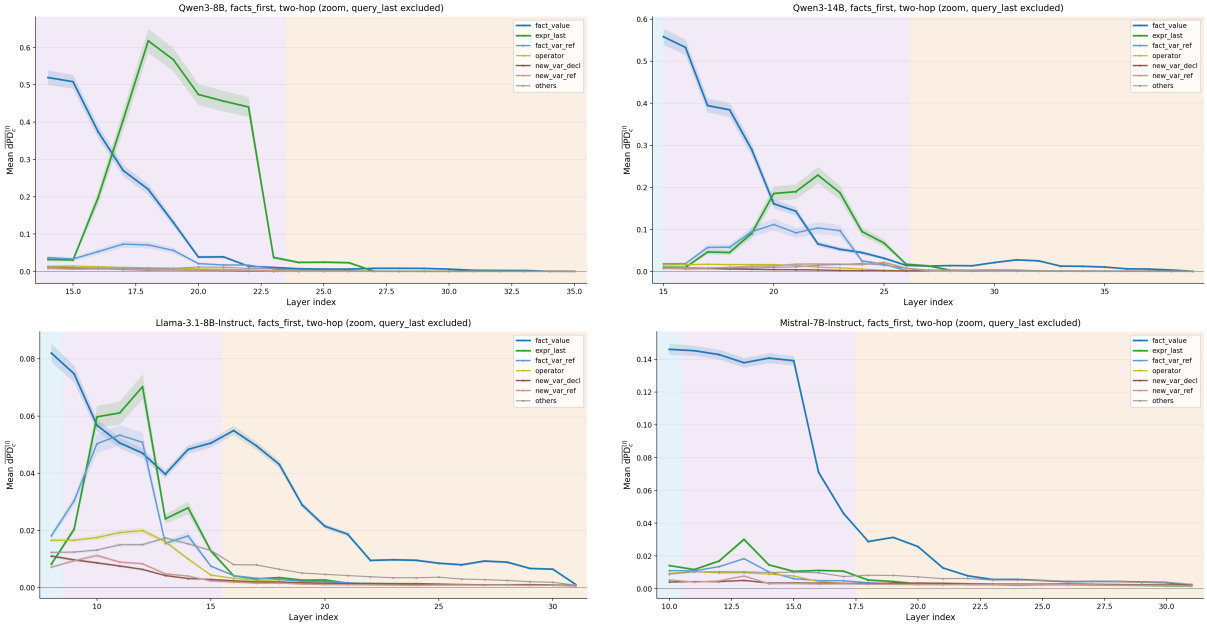


Figure 33: **Per-token mean** dPD**,** π_{FF}**, two-hop,** `query_last` **excluded.** Top-left: *Qwen3-8B*; top-right: *Qwen3-14B*; bottom-left: *Llama-3.1-8B-Instruct*; bottom-right: *Mistral-7B-Instruct*.

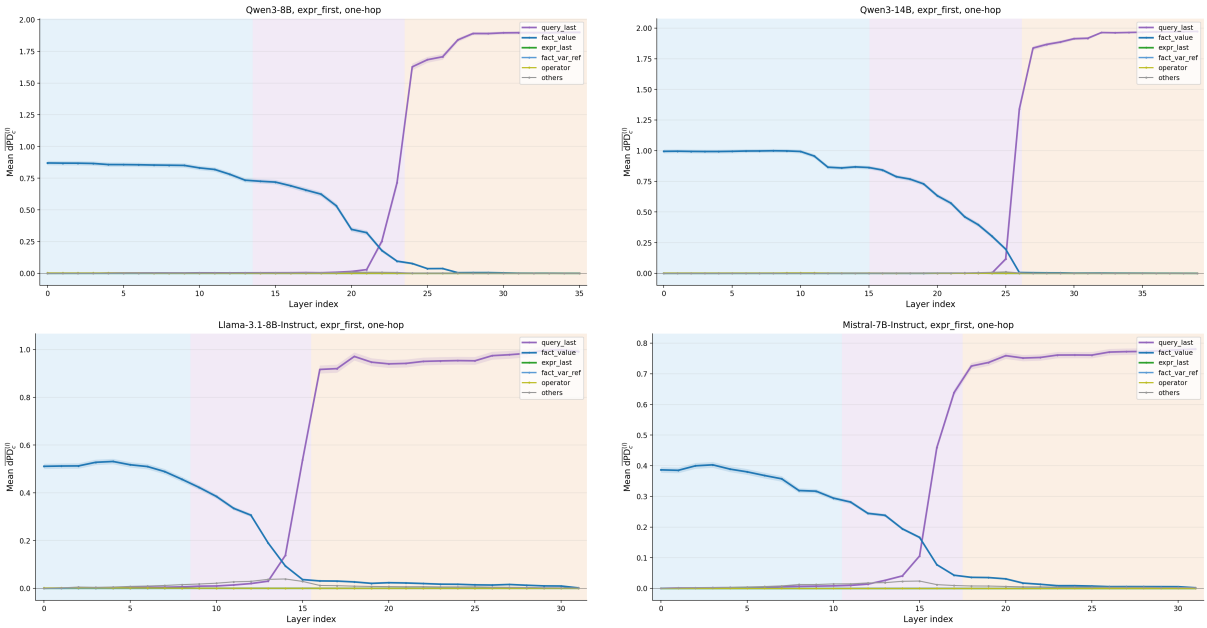


Figure 34: **Per-token mean** dPD**,** π_{EF}**, one-hop, full view.** Top-left: *Qwen3-8B*; top-right: *Qwen3-14B*; bottom-left: *Llama-3.1-8B-Instruct*; bottom-right: *Mistral-7B-Instruct*.

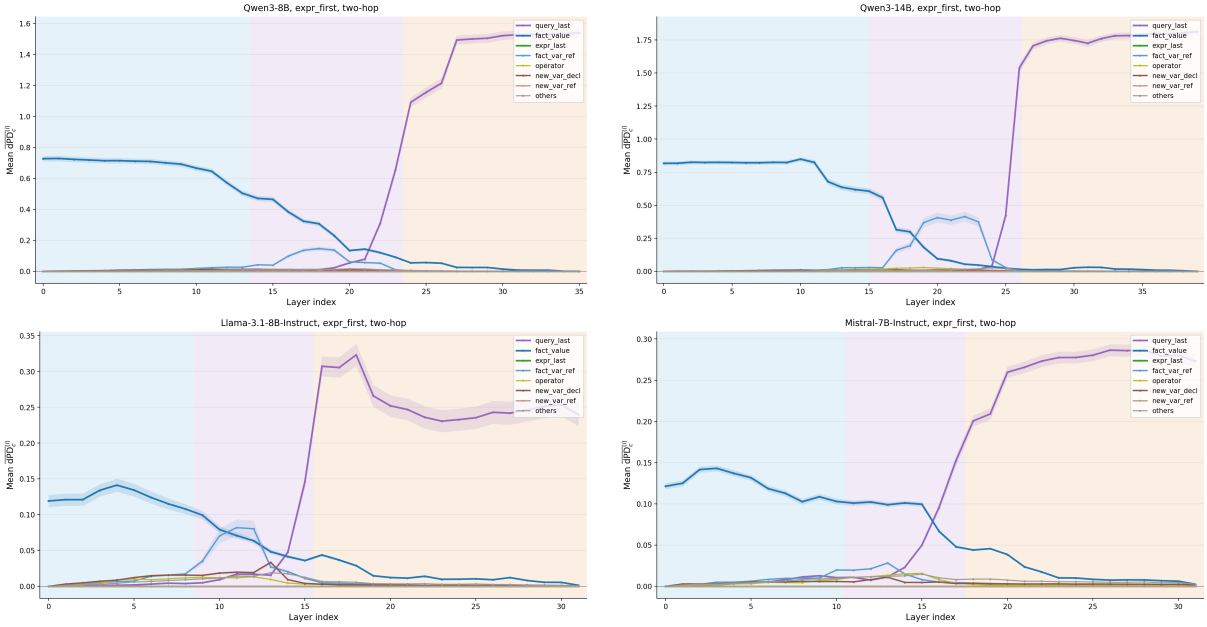


Figure 35: **Per-token mean** dPD**,** π_{EF}**, two-hop, full view.** Top-left: *Qwen3-8B*; top-right: *Qwen3-14B*; bottom-left: *Llama-3.1-8B-Instruct*; bottom-right: *Mistral-7B-Instruct*.

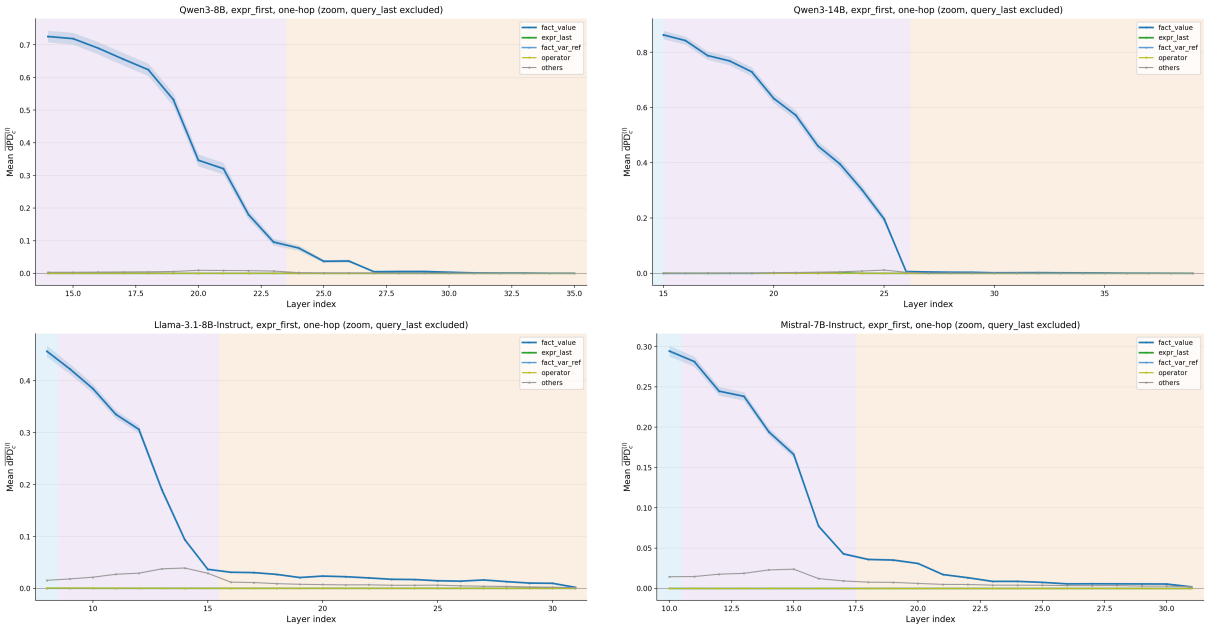


Figure 36: **Per-token mean** dPD**,** π_{EF}**, one-hop,** `query_last` **excluded.** Top-left: *Qwen3-8B*; top-right: *Qwen3-14B*; bottom-left: *Llama-3.1-8B-Instruct*; bottom-right: *Mistral-7B-Instruct*.

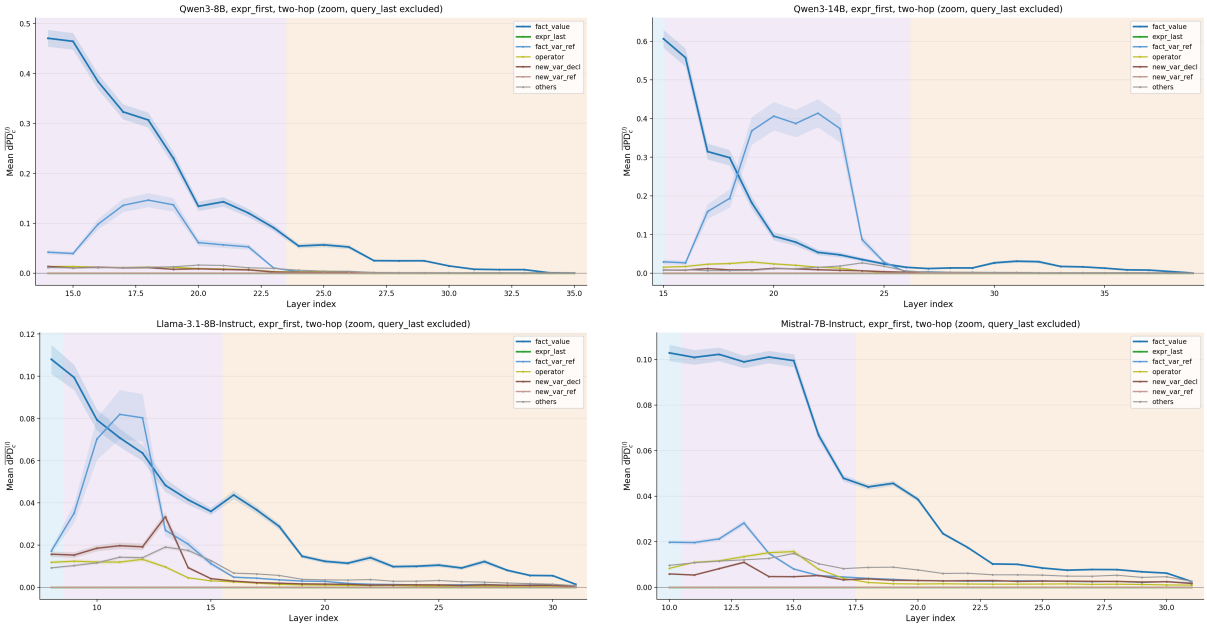
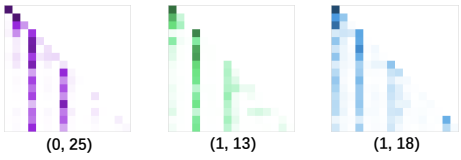
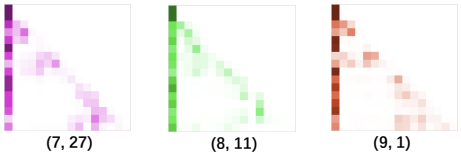


Figure 37: **Per-token mean** dPD**,** π_{EF}**, two-hop,** `query_last` **excluded.** Top-left: *Qwen3-8B*; top-right: *Qwen3-14B*; bottom-left: *Llama-3.1-8B-Instruct*; bottom-right: *Mistral-7B-Instruct*.



(a) Splitting heads



(b) Transmission heads



(c) Fact-Retrieval heads

Figure 38: **Representative attention maps for the three head roles in *Qwen3-8B*.** The prompt is `A is True, B is False, (¬A or ¬B) is`; (x, y) denotes the y-th head in the x-th layer.

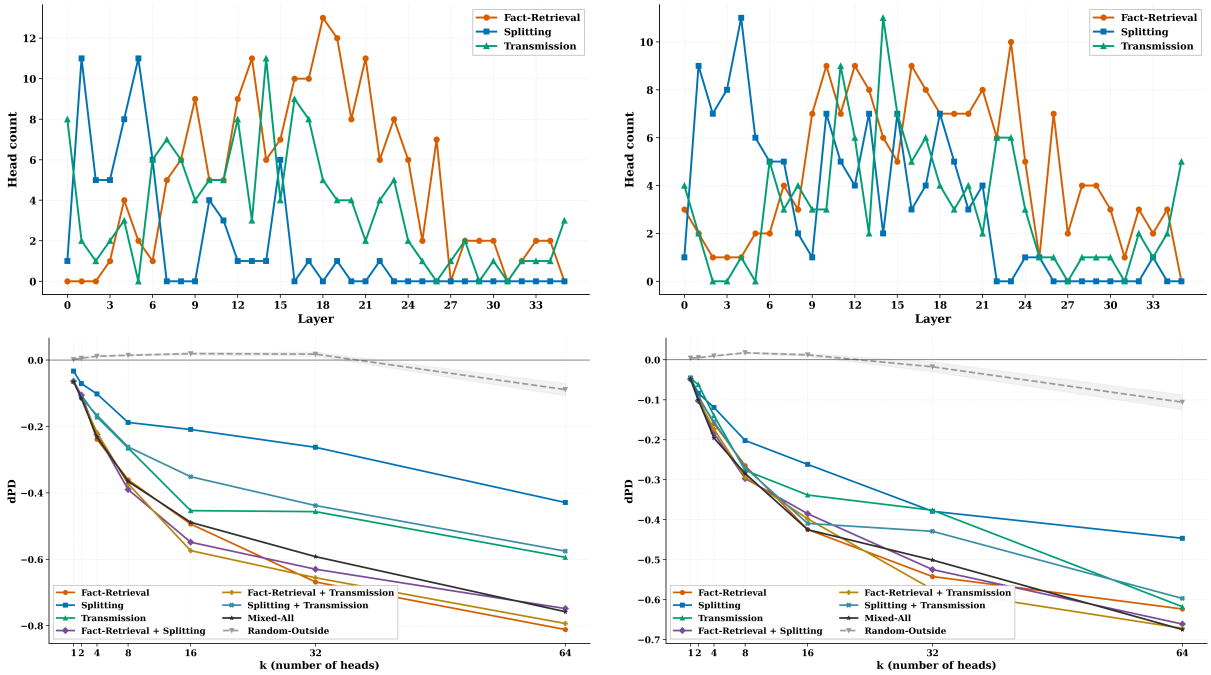


Figure 39: ***Qwen3-8B*: head roles under two prompt orders.** Top row: layer-wise role distributions (left: π_{FF}, right: π_{EF}). Bottom row: $\mathrm{dPD}/\mathrm{PD}_{\mathrm{clean}}$ validation curves under the same prompt orders.

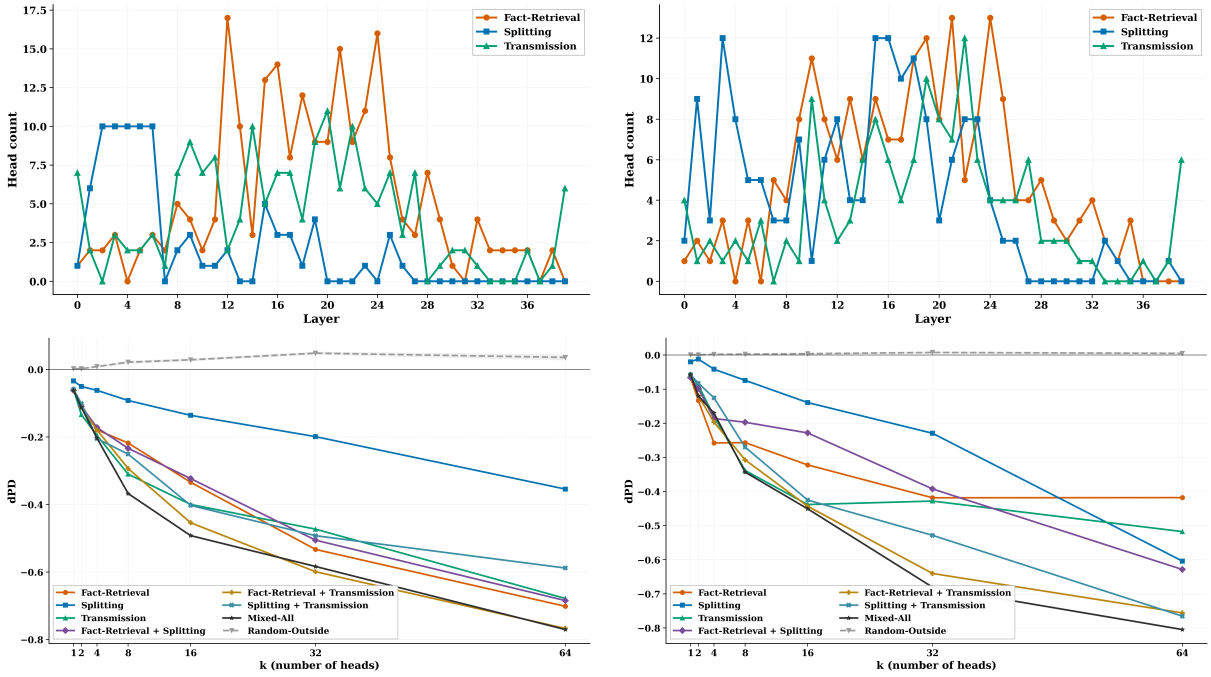


Figure 40: ***Qwen3-14B*: head roles under two prompt orders.** Top row: layer-wise role distributions (left: π_{FF}, right: π_{EF}). Bottom row: $\mathrm{dPD}/\mathrm{PD}_{\mathrm{clean}}$ validation curves under the same prompt orders.

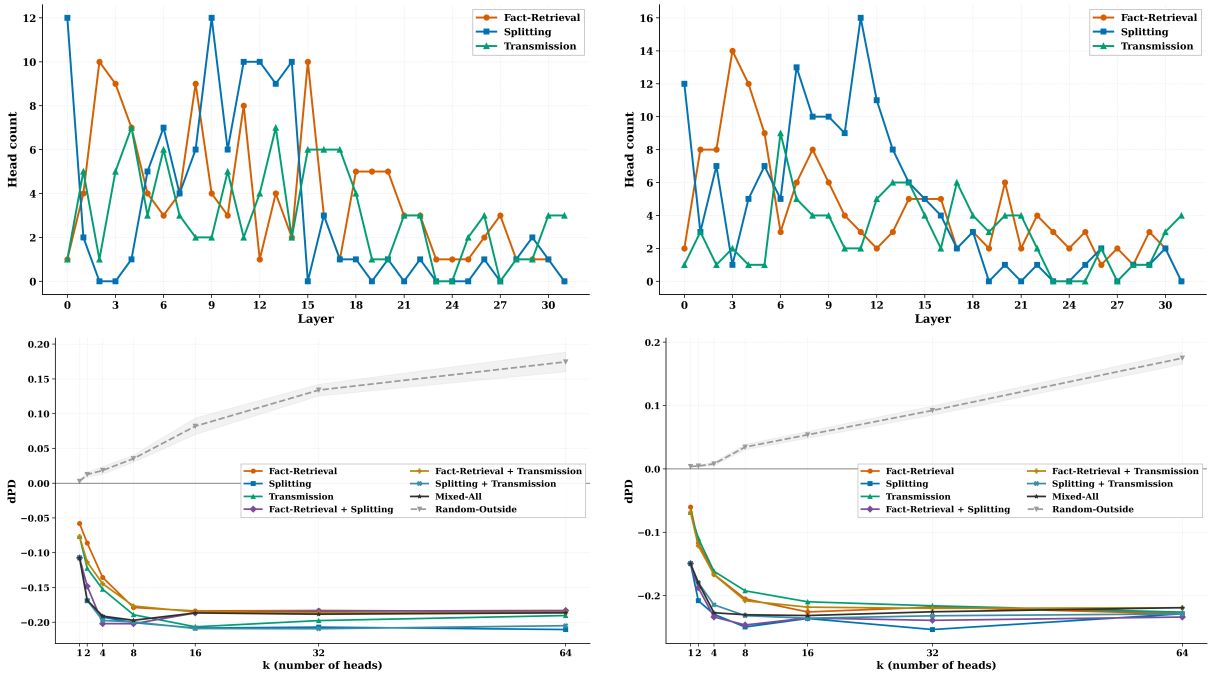


Figure 41: ***Llama-3.1-8B-Instruct*: head roles under two prompt orders.** Top row: layer-wise role distributions (left: π_{FF}, right: π_{EF}). Bottom row: $\mathrm{dPD}/\mathrm{PD}_{\mathrm{clean}}$ validation curves under the same prompt orders.

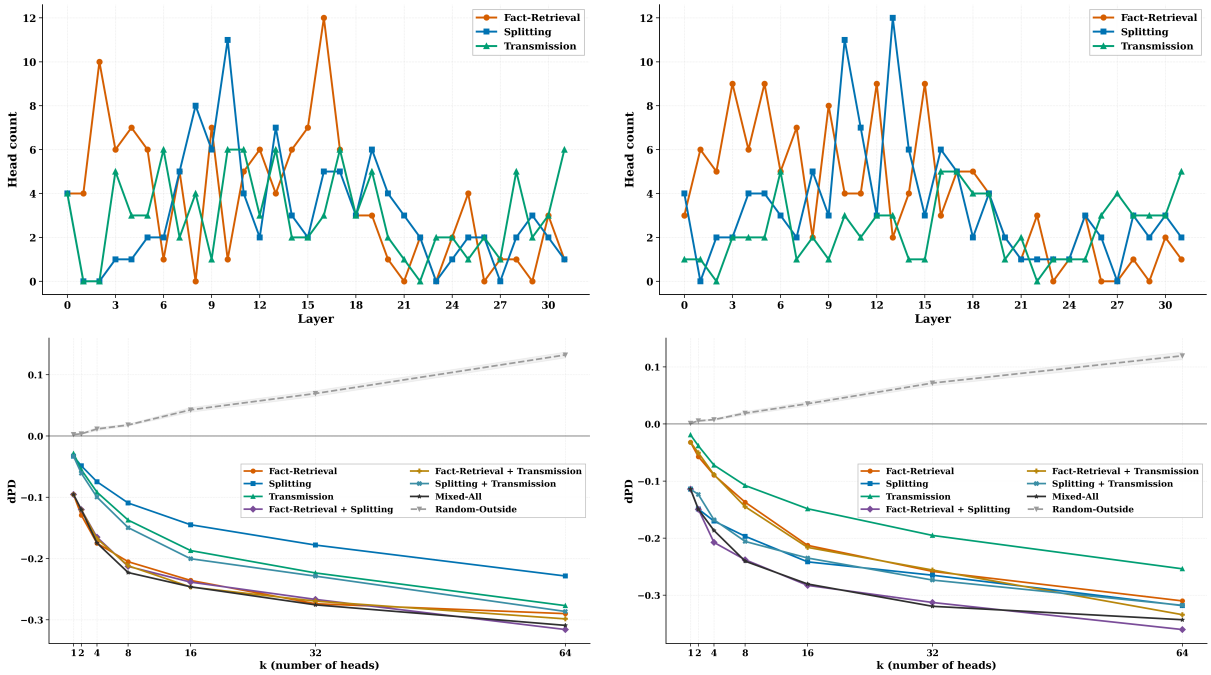


Figure 42: ***Mistral-7B-Instruct*: head roles under two prompt orders.** Top row: layer-wise role distributions (left: π_{FF}, right: π_{EF}). Bottom row: $\mathrm{dPD}/\mathrm{PD}_{\mathrm{clean}}$ validation curves under the same prompt orders.