Layer-Order Inversion: Rethinking Latent Multi-Hop Reasoning in Large Language Models

Anonymous ACL submission

Abstract

Large language models (LLMs) perform well on multi-hop reasoning, yet how they internally compose multiple facts remains unclear. Recent work proposes *hop-aligned circuit hypothesis*, suggesting that bridge entities are computed sequentially across layers before later-hop answers. Through systematic analyses on real-world multi-hop queries, we show that this hop-aligned assumption does not generalize: later-hop answer entities can become decodable earlier than bridge entities, a phenomenon we call *layer-order inversion*, which strengthens with total hops. To explain this behavior, we propose a *probabilistic recall-and-extract* framework that models multi-hop reasoning as broad probabilistic recall in shallow MLP layers followed by selective extraction in deeper attention layers. Through probing and causal intervention analyses, we further show that shallow recall of deeper-hop knowledge can act as an intuition-like signal for answer selection, with stronger effects in weaker models and higher-hop queries. Our framework further explains prior empirical phenomena and provides a mechanistic diagnosis of multi-hop failures despite correct single-hop knowledge. Code is available at `https://anonymous.4open.science/r/Layer-Order-Inversion/`.

1 Introduction

Large language models (LLMs) have demonstrated strong multi-hop reasoning capabilities (Plaat et al., 2025; Wu et al., 2025a; Li et al., 2024b), where answering a question requires integrating multiple pieces of factual knowledge. Such reasoning underlies many knowledge-intensive applications (Wei et al., 2025; Guo et al., 2024; Yu et al., 2024). Despite strong performance, how LLMs internally perform multi-hop reasoning remains poorly understood (Press et al., 2023; Hou et al., 2023).

Recent work has proposed that LLMs realize multi-hop reasoning through explicit *reasoning circuits* (Wang et al., 2024; Yao et al., 2024, 2025b), in which each hop is resolved sequentially across layers. Under this view, as illustrated in Figure 1, bridge entities (e.g., e_1) are computed at earlier layers and propagated forward, yielding a hop-aligned, layer-wise computation ($e_1 \rightarrow e_2'$). Empirical support for this hypothesis largely comes from analyses on two-hop or synthetic datasets, using techniques such as layer-wise decoding (Ghandeharioun et al., 2024; Belinkov, 2022).

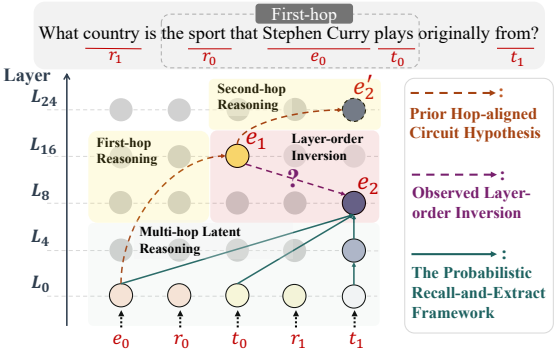


Figure 1: An illustration of latent multi-hop reasoning. The *layer-order inversion* contradicts hop-aligned circuits, while our *probabilistic recall-and-extract* framework interprets it as horizontal recall of deeper-hop knowledge that may act as an intuition-like signal.

However, whether LLMs' latent reasoning follows such hop-aligned circuits remains unclear beyond restricted settings (Biran et al., 2024; Yang et al., 2024). As illustrated in Figure 1, the reasoning-circuit view predicts a strict ordering in which bridge entities (e.g., e_1) emerge at earlier layers than later-hop entities (e.g., e_2'). In contrast, we observe clear counterexamples: the final answer entity (e.g., e_2) can become decodable at earlier layers than the corresponding bridge entity e_1. We refer to this phenomenon as *layer-order inversion*. Moreover, hop-aligned circuits also struggle to explain reasoning phenomena that are not primarily tied to model depth like shortcuts (Yuan et al., 2024). These limitations motivate us to re-examine the hop-aligned circuit hypotheses.

To this end, we systematically investigate layer-order inversion on MQuAKE (Zhong et al., 2023), a benchmark with up to four-hop queries and annotated intermediate facts. Using Patchscopes (Ghandeharioun et al., 2024), we show that layer-order inversion becomes increasingly pronounced in higher-hop queries, providing evidence that hop-aligned circuits are insufficient as a general explanation of latent multi-hop reasoning.

To explain these findings, we extend previous hypotheses (Geva et al., 2023) and propose a *probabilistic recall-and-extract* framework that views latent multi-hop reasoning as probabilistic knowledge recall and extraction rather than deterministic hop-by-hop computation. In this framework, knowledge is recalled not only vertically across layers but also horizontally across tokens. Supported by Patchscopes and hidden-state similarity analyses, horizontal recall explains layer-order inversion by allowing deeper-hop knowledge to become available before all bridge entities are explicitly decoded. Causal intervention further demonstrates that such shallow horizontal recall can act as an intuition-like signal for answer selection, while deeper attention layers selectively extract accumulated answer-relevant knowledge. This perspective offers a more complete account of prior observations: hop-aligned circuits can be interpreted as dominant recall patterns, whereas shortcut behaviors and chain-of-thought prompting reflect the role of horizontal recall and intuition-like signals in shaping answer selection. It also explains multi-hop failures as insufficient recall or extraction under the multi-hop context, even when all relevant single-hop facts are answered correctly.

Our contributions are summarized as follows:

- For the first time, we uncover *layer-order inversion*, where later-hop entities become decodable earlier than bridge entities in LLMs, demonstrating that hop-aligned circuits do not generalize to higher-hop queries.
- We introduce a *probabilistic recall-and-extract* framework that models multi-hop reasoning as probabilistic recall and selective extraction, highlighting shallow recall of deeper-hop knowledge as an intuition-like mechanism.
- Through extensive experiments, we show that this framework effectively explains prior circuit-like observations, shortcuts, and systematic failure modes in multi-hop reasoning.

2 Preliminaries

2.1 Problem Definition

We consider multi-hop reasoning over a set of entities $\mathcal{E}$ and relations $\mathcal{R}$, where each single-hop fact is represented as a triplet $(e, r, e') \in \mathcal{E} \times \mathcal{R} \times \mathcal{E}$. By treating each relation r as a mapping function $r(e) = e'$, a k-hop fact can be defined as:

$$e_{i+1} = r_i(e_i), \quad i = 0, \ldots, k-1, \tag{1}$$

which yields the chained expression:

$$r_{k-1} \circ r_{k-2} \circ \cdots \circ r_0(e_0) = e_k, \tag{2}$$

where e_0 is the subject entity, e_k is the answer entity, and $\{e_1, \ldots, e_{k-1}\}$ are the bridge entities.

While multi-hop reasoning can be formalized as relation composition, it remains unclear whether LLMs internally decompose a k-hop query into single-hop steps or implicitly encode the composed relation $r_{k-1} \circ \cdots \circ r_0$ (Dai et al., 2022). We therefore study latent multi-hop reasoning by tracing whether bridge entities $e_1, \ldots, e_{k-1}$ emerge in hidden states, and where they appear (Zhu et al., 2025).

2.2 Hop-aligned Circuits Hypothesis

Knowledge circuits refer to computation subgraphs in LLMs whose activations are sufficient to reproduce specific factual knowledge (Yao et al., 2024). The *hop-aligned circuits hypothesis* extends this view to multi-hop inference, positing that each hop is resolved sequentially across layers (Wang et al., 2024; Yao et al., 2025b). Formally, a k-hop query is resolved recursively:

$$e_{i+1} = F_i(e_i, r_i), \quad i = 0, \ldots, k-1, \tag{3}$$

where F_i denotes the mapping function from (e_i, r_i) to e_{i+1}. The final answer is obtained as:

$$F_{k-1}\big(F_{k-2}(\cdots F_0(e_0, r_0) \cdots, r_{k-1})\big) = e_k. \tag{4}$$

Each F_i is hypothesized to be implemented by a subset of layers, producing a circuit-like computation in which bridge entities are recalled and propagated in hop order, as illustrated in Figure 2(b).

Empirical support for hop-aligned circuits has been reported from multiple perspectives. Early studies (Biran et al., 2024; Ghandeharioun et al., 2024) show that patching hidden states from single-hop queries or later layers into earlier layers improves multi-hop performance, suggesting that bridge entities are computed in shallow layers.

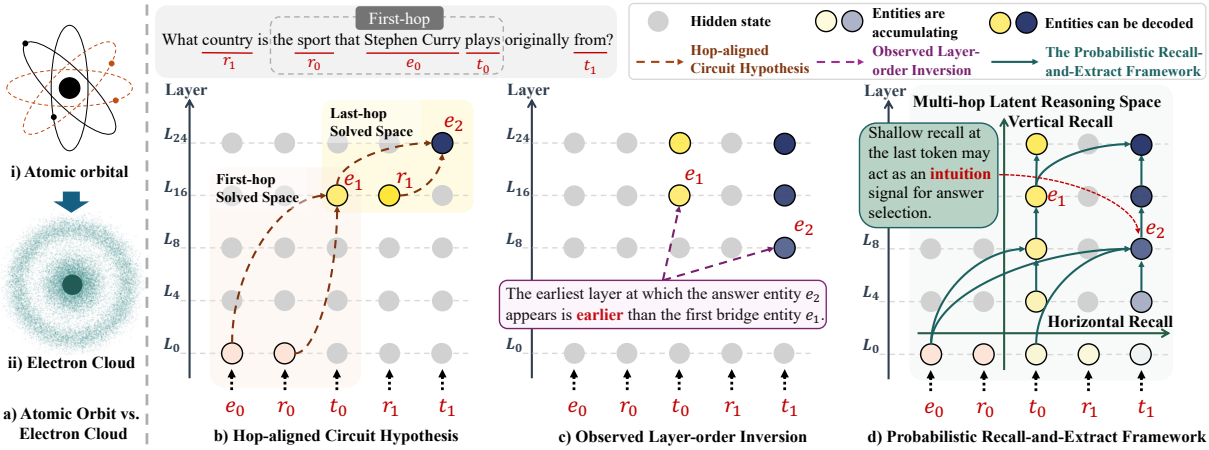


Figure 2: An illustration of latent multi-hop reasoning in LLMs from a probabilistic perspective. (a) Electron behavior follows a probabilistic distribution rather than fixed orbits. (b) **Hop-aligned circuit hypothesis** assumes layer-by-layer recall of bridge entities. (c) Observed **layer-order inversion**, where later-hop entities may emerge earlier than bridge entities. (d) **Probabilistic recall-and-extract framework** combining vertical and horizontal recall, with last token shallow recall serving as an intuition-like signal for answer selection.

Training-based studies (Wang et al., 2024; Yao et al., 2025a) fine-tune models on synthetic multi-hop data and use causal tracing (Meng et al., 2022) to show that causally important positions exhibit hop-aligned distributions. In parallel, non-tuning approaches (Yao et al., 2025b; Zhang et al., 2025) use logit-lens and patching techniques (Geva et al., 2022; Wang, 2025) to show that bridge entities emerge earlier than answer entities, which is interpreted as evidence for hop-aligned circuits.

Despite these findings, whether LLMs truly rely on hop-aligned circuits remains unclear. First, training-based evidence depends on explicit fine-tuning with synthetic multi-hop data and may not reflect the intrinsic behavior of pretrained models. Second, non-tuning analyses are largely restricted to two-hop settings and fixed templates, which may bias conclusions. Finally, prior work (Yao et al., 2024) shows that models can still answer certain multi-hop questions correctly even after first-hop knowledge is removed, challenging strictly sequential circuit-based explanations. Related phenomena, such as shortcuts (Ju et al., 2024; Zhang et al., 2024b) and chain-of-thought (Wei et al., 2022; Renze and Guven, 2024), further complicate circuit-based interpretations. These limitations motivate us to re-examine hop-aligned circuits, particularly for higher-hop queries, to better understand latent multi-hop reasoning in LLMs.

2.3 Rethinking Latent Multi-hop Reasoning

To this end, we investigate how LLMs internally perform multi-hop reasoning through the following research questions:

RQ1: Does latent multi-hop reasoning in LLMs follow hop-aligned circuits?

This question examines whether LLMs exhibit latent multi-hop reasoning when answering multi-hop queries, and whether such reasoning conforms to the layer-wise structure assumed by hop-aligned circuit hypothesis.

RQ2: If hop-aligned circuits do not fully explain LLM behavior, what internal mechanisms support multi-hop reasoning?

We aim to characterize the internal mechanisms by which LLMs acquire and utilize the knowledge required for multi-hop reasoning, and assess how these mechanisms can account for prior empirical observations.

RQ3: Why do LLMs fail on multi-hop questions?

We further investigate the causes of failure in multi-hop queries, particularly in cases where LLMs correctly answer all relevant single-hop facts but still produce incorrect multi-hop predictions.

Overview. To address these research questions, Section 3 introduces our experimental setup for analyzing latent multi-hop reasoning. Section 4 examines whether LLMs follow hop-aligned circuits (RQ1). Section 5 investigates the internal mechanisms of multi-hop reasoning from a probabilistic perspective, and identifies an intuition-like phenomenon in which deeper-hop knowledge can be directly recalled in LLMs (RQ2). Section 6 analyzes failure cases on multi-hop questions (RQ3). Finally, Section 7 concludes this paper.

Model	Category	2-hop	3-hop	4-hop	Total
GPT-J-6B	Correct	233	133	42	408
	Incorrect	298	366	178	842
	Missing	562	605	460	1,627
Llama3-8B	Correct	609	330	275	1,214
	Incorrect	420	477	397	1,294
	Missing	301	451	168	920

Table 1: Dataset statistics of MQuAKE by hop count and outcome category (**Correct**, **Incorrect**, **Missing**).

3 Experimental Setup

3.1 Datasets and Models

Our experiments are based on MQuAKE (Zhong et al., 2023), which contains multi-hop questions with up to four hops, multiple natural-language verbalizations per query, and explicit annotations of intermediate facts. These properties allow us to probe the internal emergence of bridge entities beyond two-hop, template-based synthetic settings.

To analyze outcome-specific behavior, we partition MQuAKE by model predictions into three subsets: **Correct** includes cases where both multi-hop and all single-hop questions are answered correctly; **Incorrect** includes cases where only the multi-hop question fails; and **Missing** includes cases where the multi-hop question and at least one associated single-hop question fail. Dataset statistics are summarized in Table 1. We evaluate two representative decoder-only LLMs: GPT-J-6B (Wang and Komatsuzaki, 2021) and Llama 3-8B (Touvron et al., 2023). We use greedy decoding (temperature $= 0$) to ensure deterministic and reproducible inference.

3.2 Patchscopes Analysis

We adopt Patchscopes (Ghandeharioun et al., 2024) as our primary tool for probing latent multi-hop reasoning in LLMs. Given a selected token and layer, Patchscopes extracts the corresponding hidden state from the source query, patches it into a placeholder token in a target prompt, and uses the generation from the target prompt to probe the information encoded in the source hidden state. Compared to supervised probing methods (Belinkov, 2022; Belrose et al., 2023), Patchscopes is fully unsupervised and enables direct visualization of layer-wise representations. Following prior work (Biran et al., 2024), we repeat the patching procedure three times per query and report aggregated statistics.

Post-processing. Patchscopes generations from shallow layers can be noisy due to insufficient contextual priors, which may spuriously overlap with entities of interest. We therefore apply global and local filtering based on sentence similarity, and report both raw and filtered outputs as robustness checks. Full details are provided in Appendix A.1.

3.3 Causal Intervention

To examine the causal role of knowledge recall at different token positions, we design a causal intervention method. The intervention consists of three runs: **Clean Run**, where the model processes the original multi-hop query and produces the probability of the correct answer $P_{\text{clean}}[o]$. **Corrupted Run**, where Gaussian noise is added to the subject embedding, yielding $P_{\text{corr}}[o]$. **Patched Run**, where the activation at token position i and layer l in the clean run is replaced with its corrupted counterpart, resulting in $P^{(i,l)}_{\text{patch}}[o]$. By corrupting a specified activation in the clean run and propagating its effect forward, we estimate how much this activation contributes to the original prediction while preserving the original reasoning trajectory. We measure the intervention effect as:

$$\text{IE}(i, l) = P_{\text{clean}}[o] - P^{(i,l)}_{\text{patch}}[o],$$

where a larger value indicates a stronger causal contribution to the correct prediction. Appendix A.2 further explains this design through a probabilistic-path derivation.

4 Do LLMs Rely on Explicit Hop-aligned Circuits?

We decompose **RQ1** into two sub-questions: *(RQ1.1) Do LLMs exhibit latent reasoning when answering multi-hop queries? (RQ1.2) Is such reasoning implemented through hop-aligned circuits?*

4.1 Probing Setup

Following prior work (Biran et al., 2024; Yao et al., 2025b), we apply Patchscopes to the **Correct** subset by probing hidden states at the *last subject token* and the *last token*, and report (i) the decoding frequency of each entity and (ii) the earliest layer at which each entity becomes decodable.

4.2 Results

Figure 3 reports entity decoding proportions (bars) and the average earliest decodable layer (lines) for a given token. Bars of the same color indicate queries with the same total hops, and entities are ordered from left to right by hop index. Based on these results, we draw the following conclusions.

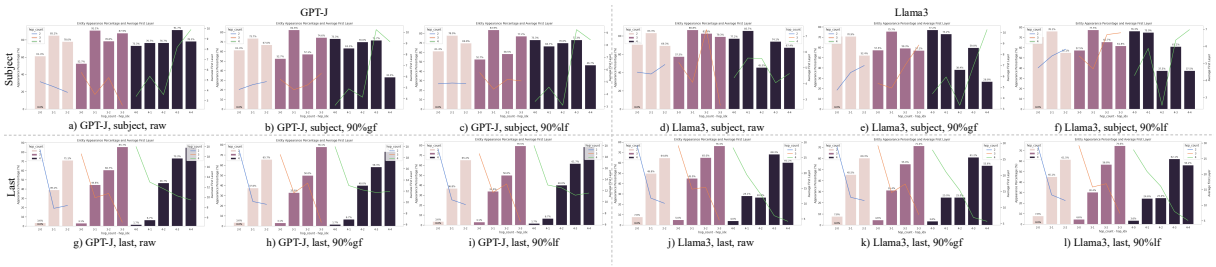


Figure 3: Patchscopes results for GPT-J (left) and Llama3 (right) on the Correct subset. We probe hidden states at the last subject token (top) and the last token of the query (bottom) under three settings: raw, 90% gf, and 90% lf.

Model	Token	2-hop			3-hop				4-hop				
		e_0	e_1	e_2	e_0	e_1	e_2	e_3	e_0	e_1	e_2	e_3	e_4
GPT-J	Subject	4.83	4.35	3.81	5.74	3.60	5.24	2.55	3.41	5.37	3.59	8.18	9.89
	Last	19.36	7.81	**8.38**	18.57	9.91	10.76	**5.05**	30.00	13.00	11.68	10.27	**9.47**
Llama3	Subject	4.72	4.61	5.20	5.34	4.08	7.41	2.63	4.37	5.61	5.59	4.06	4.64
	Last	28.92	11.81	**10.11**	28.93	17.75	15.34	**4.64**	28.22	19.43	13.18	6.08	**4.24**

Table 2: Average Earliest Layer of Entity Emergence under the Raw Setting

RQ1.1: LLMs generally exhibit latent reasoning in multi-hop question answering. When probing the *subject* token, all entities are frequently decodable, with the first bridge entity e_1 often achieving the highest decoding rate (Figure 3 (a,d)). Under filtered settings (Figure 3 (b-c,e-f)), the earliest decodable layer of entities tends to shift deeper as the hop index increases, consistent with progressive information accumulation across layers. When probing the last token (Figure 3 (g-l)), bridge entities remain decodable, whereas the subject entity e_0 is rarely decoded. These observations suggest that LLMs retain intermediate factual information in their hidden states during multi-hop reasoning.

RQ1.2: However, Latent multi-hop reasoning does not consistently follow explicit hop-aligned circuits. At the last token, the final-hop entity is decoded with higher probability than earlier-hop entities, and its earliest decodable layer often shifts earlier as the hop index increases. This contrasts with hop-aligned circuit hypotheses, which predict similar decoding probabilities for bridge entities and monotonically increasing earliest decodable layers with hop index.

Crucially, Table 2 (see also Appendix A.4) shows that the last-hop entity at the last token (**bold**) often becomes decodable as early as, or earlier than, bridge entities at the subject token. We term this phenomenon *layer-order inversion*. For example, on 4-hop queries in Llama 3, the final entity e_4 at the last token emerges earlier than the first bridge entity e_1 at the subject across all settings. Such behavior is incompatible with explicit hop-aligned circuits, which assume that bridge entities emerge sequentially across layers.

4.3 Discussion

A key reason prior work attributes multi-hop reasoning to hop-aligned circuits is the limited experimental scope. As discussed in Section 2.2, most analyses focus on two-hop reasoning over synthetic, template-based datasets, which restricts observable latent behaviors. Consistently, we observe no layer-order inversion in two-hop queries, while the effect becomes more pronounced as hop count increases.

At a deeper level, this discrepancy reflects a mismatch between the probabilistic nature of LLMs and symbolic interpretations of their internal computations. Prior analyses often interpret the most prominent internal activation patterns as evidence of deterministic, step-by-step reasoning. A similar misinterpretation also appeared in early atomic models, which assumed fixed electron orbits rather than probability clouds (Figure 2(a)). By analogy, rather than a strictly symbolic computation, latent multi-hop reasoning in LLMs should be understood as a probabilistic recall process, which we investigate systematically in the next section.

5 A Probabilistic Perspective on the Recall-and-Extract Framework

To answer **RQ2**, we extend the Recall-and-Extract framework (Geva et al., 2023) from a probabilistic perspective by introducing horizontal recall and intuition-like recall. We focus on two sub-questions: *(RQ2.1) How LLMs probabilistically*

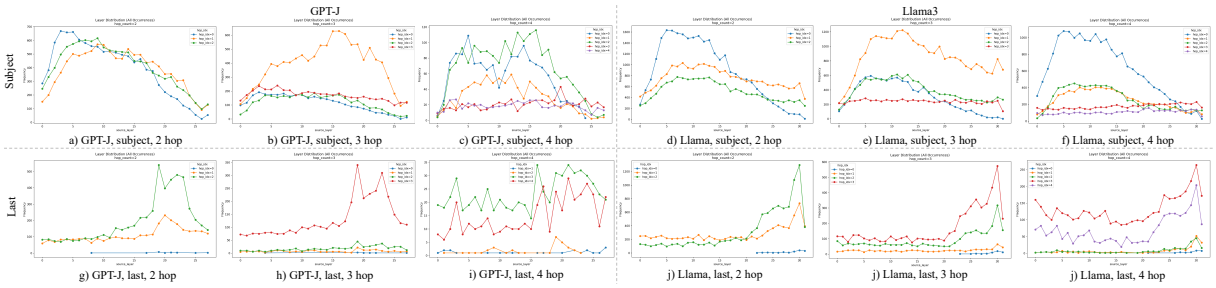


Figure 4: Layer-wise generation distributions produced by Patchscopes on the Correct subset.

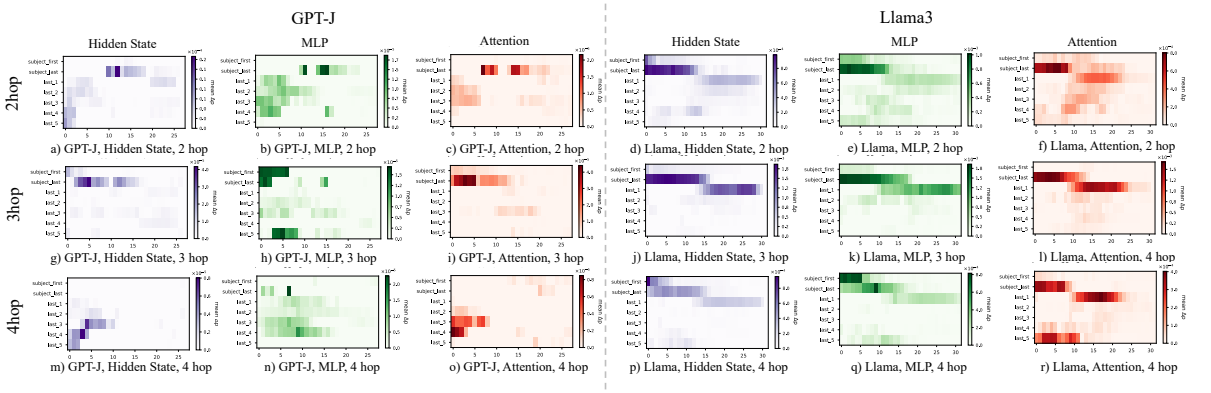


Figure 5: Causal intervention effects across layers and token positions on the Correct subset.

acquire the knowledge for multi-hop reasoning? and *(RQ2.2) How can such a probabilistic mechanism account for prior empirical observations?*

5.1 Probing Setup

To address **RQ2.1**, we conduct three complementary analyses on the **Correct** subset. Specifically, (1) we analyze Patchscopes across layers to characterize the probabilistic distribution of entity-level knowledge in intermediate hidden states; (2) causal intervention analysis is conducted to assess causal effects at the subject token and the last k tokens ($k = 5$) across hidden states, MLP outputs, and attention outputs; (3) we further perform hidden-state similarity analysis as supplementary evidence to track how internal representations evolve across layers, with details provided in Appendix A.1.2.

5.2 Results

Figure 4 illustrates the distribution of entities decoded by Patchscopes across layers, where curves of different colors correspond to entities at different hop indexes. Figure 5 reports causal intervention effects across layers and token positions, with darker colors indicating larger decreases in the correct-answer probability after intervention. Figure 8 in Appendix A.1.2 provides hidden-state similarity results between multi-hop queries and their corresponding single-hop queries, grouped by *(hop index – total hops)*. Together, these results suggest a probabilistic extension of the Recall-and-Extract mechanism, which we summarize below.

RQ2.1: Latent multi-hop reasoning in LLMs is supported by a probabilistic recall-and-extract mechanism, which can be summarized by three key points as shown in Figure 2(d).

Probabilistic recall operates both vertically and horizontally. Our results suggest that knowledge recall in LLMs is jointly shaped by vertical recall across layers and horizontal recall across tokens. Vertical recall has been suggested in prior studies (Meng et al., 2022; Geva et al., 2023), where MLP layers progressively recall factual knowledge conditioned on the current representation. Consistent with this view, Figure 4 (a–f) shows that entities from different hop indexes can be decoded across layers at the subject position, while Figure 8 provides supplementary evidence that representations gradually shift from earlier-hop to later-hop knowledge. However, vertical recall alone cannot explain *layer-order inversion*. We therefore propose horizontal recall, where accumulated context guides knowledge recall. As shown in Figure 4(g–j), later-hop entities are often more decodable than earlier-hop entities at the last token even in shal-

low layers, indicating that deeper-hop knowledge can become available without explicitly decoding all preceding bridge entities, a behavior consistent with the autoregressive next-token prediction objective of LLMs (Brown et al., 2020).

Notably, vertical and horizontal recall can interact: early-hop knowledge obtained through vertical recall can increase the probability of horizontal deeper-hop recall, and vice versa. Such coupling jointly shapes how knowledge distributions evolve during latent multi-hop reasoning (Zhang et al., 2024a; Wu et al., 2025b).

Last-token shallow recall acts as an intuition-like signal. Horizontal recall explains how deeper-hop knowledge becomes available at the last token, but its causal effect on prediction remains unclear. To examine its causal role, we conduct the causal intervention analysis. As shown in Figure 5, for the weaker GPT-J model, intervening on shallow activations at the last tokens produces stronger effects than interventions at the subject token, with the effect increasing in higher-hop queries. In contrast, for stronger Llama3-8B model, last-token interventions are much weaker than subject.

These results suggest that shallow horizontal recall of deeper-hop knowledge at the last token behaves as an intuition-like signal for answer selection. Analogous to human intuition, it may not alter the reasoning process itself, but can bias the final judgment. In LLMs, attention masking prevents this signal from influencing earlier tokens, so it does not directly participate in preceding hop-aligned recall. Nevertheless, it can still affect the final prediction, with its influence increasing for weaker models and higher-hop queries.

Attention extracts answer-relevant knowledge. Figure 4(g–j) shows a sharp increase in answer-entity decodability at the last token in deeper layers, aligning with the Recall-and-Extract view that deeper-layer attention extracts answer-relevant information (Geva et al., 2023; Li et al., 2024a; Tamayo et al., 2024). Causal intervention further supports this view, as perturbing deep-layer attention at the last token has clear effects that grow with hop count (Figure 5(f,l,r)).

5.3 Discussion

RQ2.2: The probabilistic recall-and-extract framework provides a more complete and coherent account of prior multi-hop reasoning phenomena. Its advantages are most evident in explaining hop-aligned circuits and shortcut.

Hop-aligned Circuits Hypothesis. Prior work (Yao et al., 2025b) interprets layer-wise decoding patterns as evidence that LLMs resolve each hop sequentially. Under our probabilistic recall-and-extract framework, these observations are more naturally explained as reflecting the most probable recall trajectory across token positions. For example, at the subject position, the probability of decoding the first bridge entity e_1 is typically higher than that of decoding later-hop entities (e.g., e_2) at the last token, making the "$e_1 \rightarrow e_2$" trajectory more likely to be observed in layer-wise analyses. Crucially, this probabilistic interpretation not only accounts for such circuit-like patterns in restricted settings (e.g., two-hop), but also accommodates higher-hop behaviors such as *layer-order inversion*, which are difficult to reconcile with strictly hop-aligned circuit explanations.

Shortcut. Shortcuts (Ju et al., 2024; Zhang et al., 2024b) occur when an LLM answers a multi-hop question correctly despite failing on some corresponding single-hop questions. Under our framework, shortcuts can be viewed as cases where the intuition-like signal is directly used for answer selection. Our causal intervention results suggest that such direct recall paths become more influential for weaker models or harder queries, where reliable hop-by-hop reasoning is less likely to dominate. This aligns with broader shortcut-learning studies showing that models may exploit shallow associative features instead of robust reasoning in complex scenarios (Geirhos et al., 2020; Yuan et al., 2024).

Beyond the above phenomena, knockout (Geva et al., 2023), back-patching (Biran et al., 2024), and Chain-of-Thought (Wei et al., 2022) are also consistent with our framework, as they modulate shallow-layer recall of deeper-hop knowledge through suppression, restoration, or contextual enrichment. Detailed analyses are provided in Appendix B.

6 Reinterpreting Errors in Multi-hop Reasoning

In this section, we address **RQ3** by reinterpreting errors in multi-hop reasoning under our probabilistic recall-and-extract framework, providing new insights for model intervention and editing.

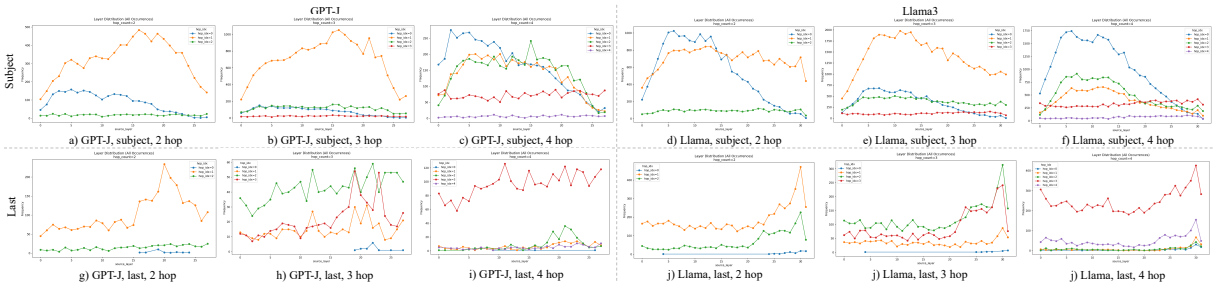


Figure 6: Layer-wise generation distributions produced by Patchscopes on the Incorrect subset.

6.1 Probing Setup

To investigate **RQ3**, we apply Patchscopes to instances in the **Incorrect** and **Missing** subsets. In addition, we further compare layer-wise hidden-state similarity between **Correct** and **Incorrect** to isolate internal differences when single-hop facts are correct but multi-hop reasoning fails.

6.2 Results

We first analyze **Incorrect** cases and attribute the observed failures to two main factors.

RQ3.1: Insufficient shallow recall of deeper-hop knowledge. Figure 6 shows that deeper-hop knowledge is insufficiently decoded in shallow layers, especially at the last token. This indicates that the model fails to form sufficient shallow recall of deeper-hop entities under the multi-hop context. Figure 9 in Appendix A.5 further show lower overall decoding rates of deeper-hop entities in **Incorrect** cases. Moreover, hidden-state similarity analysis in Figure 8 in Appendix A.1.2 shows that the gap between **Correct** and **Incorrect** cases increases as hop depth grows. These results suggest that even when the model answers all corresponding single-hop questions correctly, it may fail to activate sufficient deeper-hop knowledge under different multi-hop contexts, a limitation also noted in prior work (Yang et al., 2025).

RQ3.2: Insufficient knowledge extraction in deeper attention layers. As shown in Figure 6(g-j), even at the last token in deeper layers, the final-hop entity often fails to dominate the decoding distribution: its probability may remain low or only comparable to that of bridge entities. This suggests that deeper-layer attention may not sufficiently amplify answer-relevant knowledge to support accurate generation, resulting in extraction failures despite partial recall.

We further analyze **Missing** cases in Appendix C. When single-hop facts are answered incorrectly, early-hop knowledge (e.g., the first-hop entity) also exhibits low recall probability under the multi-hop query, preventing correct answer generation.

6.3 Discussion

Under our probabilistic recall-and-extract framework, both hop-aligned reasoning and intuition-like recall rely on context-conditioned knowledge recall. Our results show that such recall differs between multi-hop and single-hop settings, even when they involve the same underlying knowledge. This offers new insights for related research areas. For example, in knowledge editing, key representations should move beyond subject-centric samples and be derived through additional techniques to obtain more universal representations (Zhang et al., 2024c). Moreover, attention-encoded knowledge deserves closer study, as existing methods (Meng et al., 2023; Fang et al., 2025) mainly focus on MLP-stored knowledge.

7 Conclusion

In this work, we investigated latent multi-hop reasoning in LLMs and identified *layer-order inversion*, where later-hop entities become decodable earlier than early-hop bridge entities as hop count increases. This finding challenges the hop-aligned circuit hypothesis and motivates our *probabilistic recall-and-extract* framework, which views latent reasoning as a superposition of probabilistic reasoning paths. Under this framework, we introduce horizontal recall across tokens and intuition-like shallow recall at the last token. Causal intervention further shows that such last-token recall can influence answer selection, with stronger effects in weaker models and higher-hop queries. Our framework further interprets hop-aligned circuits as dominant high-probability recall paths, explains the role of horizontal recall in shortcut behaviors, and diagnoses multi-hop failures as insufficient recall or extraction. In future work, we will further investigate how intuition-like recall can be strengthened to improve LLM reasoning performance.

Limitations

Firstly, constrained by computational resources, our experiments are conducted on relatively small-scale models. While we are unable to evaluate larger models (e.g., 70B parameters), we believe that the observed phenomena are sufficient to reveal limitations of existing hop-aligned circuit hypotheses. Rather than claiming universality across model scales, our goal is to motivate a probabilistic perspective on multi-hop reasoning that better accounts for the observed behaviors.

Second, our interpretation of intuition-like recall signals remains hypothesis-driven. It is motivated by Patchscopes results and causal intervention analysis, which suggest that shallow horizontal recall of deeper-hop knowledge at the last token can influence answer selection. However, we do not further verify how such signals can be guided, strengthened, or incorporated into inference strategies to improve LLM reasoning performance. Since intuition-like recall may interact with step-by-step reasoning under the probabilistic recall framework, designing effective methods to leverage this signal requires further theoretical analysis and controlled experiments. We leave this direction for future work. Nevertheless, intuition-like recall offers a new perspective for understanding and improving latent multi-hop reasoning in LLMs.

References

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.

Eden Biran, Daniela Gottesman, Sohee Yang, Mor Geva, and Amir Globerson. 2024. Hopping too late: Exploring the limitations of large language models on multi-hop queries. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14113–14130.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502.

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. 2025. Alphaedit: Null-space constrained knowledge editing for language models. In *The Thirteenth International Conference on Learning Representations*.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

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*, pages 12216–12235.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 30–45, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495.

Asma Ghandeharioun, Avi Caciularu, Adam Pearce, Lucas Dixon, and Mor Geva. 2024. Patchscopes: a unifying framework for inspecting hidden representations of language models. In *Proceedings of the 41st International Conference on Machine Learning*, pages 15466–15490.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V Chawla, Olaf Wiest, and Xiangliang Zhang. 2024. Large language model based multi-agents: a survey of progress and challenges. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 8048–8057.

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 *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4902–4919.

Jiachen Jiang, Jinxin Zhou, and Zhihui Zhu. 2024. Tracing representation progression: Analyzing and enhancing layer-wise similarity. In *The Thirteenth International Conference on Learning Representations*.

Tianjie Ju, Yijin Chen, Xinwei Yuan, Zhuosheng Zhang, Wei Du, Yubin Zheng, and Gongshen Liu. 2024. Investigating multi-hop factual shortcuts in knowledge editing of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8987–9001.

Xiaopeng Li, Shasha Li, Shezheng Song, Jing Yang, Jun Ma, and Jie Yu. 2024a. Pmet: Precise model editing in a transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18564–18572.

Yanyang Li, Shuo Liang, Michael Lyu, and Liwei Wang. 2024b. Making long-context language models better multi-hop reasoners. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2462–2475, Bangkok, Thailand. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Jason Phang, Haokun Liu, and Samuel Bowman. 2021. Fine-tuned transformers show clusters of similar representations across layers. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 529–538.

Aske Plaat, Annie Wong, Suzan Verberne, Joost Broekens, and Niki Van Stein. 2025. Multi-step reasoning with large language models, a survey. *ACM Computing Surveys*.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2023. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5687–5711.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Matthew Renze and Erhan Guven. 2024. The benefits of a concise chain of thought on problem-solving in large language models. In *2024 2nd International Conference on Foundation and Large Language Models (FLLM)*, pages 476–483. IEEE.

Daniel Tamayo, Aitor Gonzalez-Agirre, Javier Hernando, and Marta Villegas. 2024. Mass-editing memory with attention in transformers: A cross-lingual exploration of knowledge. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5831–5847.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax.

Boshi Wang, Xiang Yue, Yu Su, and Huan Sun. 2024. Grokking of implicit reasoning in transformers: A mechanistic journey to the edge of generalization. *Advances in Neural Information Processing Systems*, 37:95238–95265.

Zhenyu Wang. 2025. Logitlens4llms: Extending logit lens analysis to modern large language models. *arXiv preprint arXiv:2503.11667*.

Hui Wei, Zihao Zhang, Shenghua He, Tian Xia, Shijia Pan, and Fei Liu. 2025. PlanGenLLMs: A modern survey of LLM planning capabilities. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 19497–19521, Vienna, Austria. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Jian Wu, Linyi Yang, Dongyuan Li, Yuliang Ji, Manabu Okumura, and Yue Zhang. 2025a. Mmqa: Evaluating llms with multi-table multi-hop complex questions. In *The Thirteenth International Conference on Learning Representations*, page 1.

Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. 2025b. Inference scaling laws: An empirical analysis of compute-optimal inference for llm problem-solving. In *The Thirteenth International Conference on Learning Representations*.

Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. 2024. Do large language models latently perform multi-hop reasoning? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10210–10229.

Sohee Yang, Nora Kassner, Elena Gribovskaya, Sebastian Riedel, and Mor Geva. 2025. Do large language models perform latent multi-hop reasoning without exploiting shortcuts? In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3971–3992, Vienna, Austria. Association for Computational Linguistics.

Yuekun Yao, Yupei Du, Dawei Zhu, Michael Hahn, and Alexander Koller. 2025a. Language models can learn implicit multi-hop reasoning, but only if they have lots of training data. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9695–9713, Suzhou, China. Association for Computational Linguistics.

Yunzhi Yao, Jizhan Fang, Jia-Chen Gu, Ningyu Zhang, Shumin Deng, Huajun Chen, and Nanyun Peng. 2025b. CaKE: Circuit-aware editing enables generalizable knowledge learners. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 11377–11393, Suzhou, China. Association for Computational Linguistics.

Yunzhi Yao, Ningyu Zhang, Zekun Xi, Mengru Wang, Ziwen Xu, Shumin Deng, and Huajun Chen. 2024. Knowledge circuits in pretrained transformers. *Advances in Neural Information Processing Systems*, 37:118571–118602.

Hao Yu, Aoran Gan, Kai Zhang, Shiwei Tong, Qi Liu, and Zhaofeng Liu. 2024. Evaluation of retrieval-augmented generation: A survey. In *CCF Conference on Big Data*, pages 102–120. Springer.

Yu Yuan, Lili Zhao, Kai Zhang, Guangting Zheng, and Qi Liu. 2024. Do LLMs overcome shortcut learning? an evaluation of shortcut challenges in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12188–12200, Miami, Florida, USA. Association for Computational Linguistics.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. 2024a. When scaling meets llm finetuning: The effect of data, model and finetuning method. In *The Twelfth International Conference on Learning Representations*.

Mengqi Zhang, Bowen Fang, Qiang Liu, Pengjie Ren, Shu Wu, Zhumin Chen, and Liang Wang. 2024b. Enhancing multi-hop reasoning through knowledge erasure in large language model editing. *arXiv preprint arXiv:2408.12456*.

Mengqi Zhang, Xiaotian Ye, Qiang Liu, Pengjie Ren, Shu Wu, and Zhumin Chen. 2024c. Knowledge graph enhanced large language model editing. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22647–22662.

Zhuoran Zhang, Yongxiang Li, Zijian Kan, Keyuan Cheng, Lijie Hu, and Di Wang. 2025. Locate-then-edit for multi-hop factual recall under knowledge editing. In *Forty-second International Conference on Machine Learning*.

Zexuan Zhong, Zhengxuan Wu, Christopher Manning, Christopher Potts, and Danqi Chen. 2023. MQuAKE: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15686–15702, Singapore. Association for Computational Linguistics.

Rui-Jie Zhu, Tianhao Peng, Tianhao Cheng, Xingwei Qu, Jinfa Huang, Dawei Zhu, Hao Wang, Kaiwen Xue, Xuanliang Zhang, Yong Shan, and 1 others. 2025. A survey on latent reasoning. *arXiv preprint arXiv:2507.06203*.

A Supplementary Experiment

A.1 Patchscopes Analysis

Due to space constraints, we provide a overview of Patchscopes in Section 3.2. Here, we present the detailed implementation.

A.1.1 Setting

We adopt Patchscopes (Ghandeharioun et al., 2024) as the primary tool to probe latent multi-hop reasoning in LLMs. Patchscopes analyzes internal knowledge by extracting the hidden state from a selected token at a given layer when processing the original query, injecting it into a placeholder token x in an explanatory target prompt like *Syria: Syria is a country in the Middle East, Leonardo DiCaprio: Leonardo DiCaprio is an American actor, Samsung: Samsung is a South Korean multinational corporation, x*, and observing the generation. The generated text provides a natural language interpretation of the knowledge encoded in the source hidden state. Compared to probing methods (Belinkov, 2022; Belrose et al., 2023), Patchscopes is fully unsupervised and enables direct visualization of knowledge encoded in earlier layers.

Following prior work (Biran et al., 2024), we repeat the patching procedure three times per query and report aggregated statistics. Despite using a different dataset, we observe consistent trends on two-hop queries with previous work, supporting the reliability of our results.

A.1.2 Post-processing and Filtering.

When decoding from shallow layers, Patchscopes may produce random outputs due to insufficient contextual priors. Such generations can still include frequent entities (e.g., countries or cities), which may coincidentally overlap with bridge entities and introduce noise when estimating the earliest emergence layer. To mitigate this issue, we try three filters:

- **Global filtering (GF)** computes the similarity (Reimers and Gurevych, 2019) between generated outputs and the original query, and removes the bottom $k\%$ of results with the lowest similarity across the entire dataset.

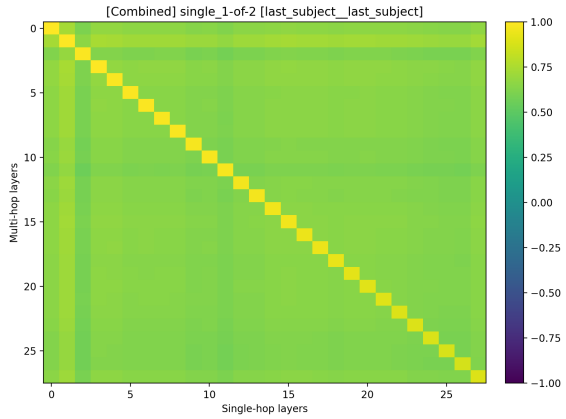


Figure 7: Cross-layer Similarity of Subject *mlp_fc_in* Representations on GPT-J-6B.

- **Local filtering (LF)** applies the same similarity filtering method to each instance, retaining only the top $(100 - k)\%$ generations for each query.
- **Layer filtering** discards generations originating from source layers below a predefined threshold.

As shown in the Figures 11, 10, and 12, the overall decoding distributions remain largely unchanged under most filtering, and noticeable shifts only emerge under strong filtering, specifically with 90% global filtering and 90% local filtering. We therefore focus on these two settings in the main analysis. We therefore report results on the raw Patchscopes outputs as well as those obtained with 90% global and local filtering.

A.2 Causal Intervention

In Section 3.3, we introduce a causal intervention method to analyze the role of internal representations. Different from prior restoration-based approaches (Meng et al., 2022; Yao et al., 2025a), we apply corruption to the clean run rather than restoring clean activations in a corrupted run.

The key motivation comes from our probabilistic recall-and-extract framework in Section 5, where multi-hop reasoning is modeled as a superposition of reasoning paths:

$$P(e_n \mid x) = \sum_{\pi \in \Pi} P(\pi \mid x)\, P(e_n \mid \pi, x). \quad (5)$$

We decompose the prediction into two dominant paths, the hop-aligned reasoning path π_{hop} and the direct recall path π_{direct}:

$$\begin{aligned} P(e_n \mid x) \approx\; & P(\pi_{\text{hop}} \mid x) \prod_{i=1}^{n} P(e_i \mid e_{i-1}, r_i) \\ & + P(\pi_{\text{direct}} \mid x)\, P(e_n \mid \pi_{\text{direct}}, x) \\ & + \sum_{\pi \in \Pi'} P(\pi \mid x)\, P(e_n \mid \pi, x). \end{aligned} \quad (6)$$

This decomposition shows that the final prediction depends not only on recalled information, but also on the underlying path distribution $P(\pi \mid x)$, which is determined by preceding tokens. Therefore, perturbing the input (e.g., subject embedding) can disrupt the entire reasoning path, making it difficult to isolate the contribution of later representations.

To preserve the reasoning paths while isolating local effects, we apply corrupted activations to the clean run. This allows us to better measure how specific representations, especially at the last token, affect the final prediction without altering the overall reasoning trajectory.

A.3 Hidden State Similarity

Section 5 reports hidden-state similarity experiments as supporting evidence. Due to space constraints, we describe the detailed implementation and additional results in this section.

A.3.1 Setting

We further conduct a hidden state similarity analysis (Phang et al., 2021; Jiang et al., 2024) as a supplementary experiment to examine how knowledge evolves during multi-hop reasoning. This analysis is motivated by prior work viewing MLPs in transformers as key–value memories for factual knowledge (Geva et al., 2021). Formally, the hidden state of token i at layer l is updated as

$$\begin{aligned} h_i^{(l)} &= h_i^{(l-1)} + a_i^{(l)} + m_i^{(l)}, \\ a_i^{(l)} &= \text{attn}^{(l)}\left(h_1^{(l-1)}, h_2^{(l-1)}, \ldots, h_i^{(l-1)}\right), \\ m_i^{(l)} &= W_{\text{proj}}^{(l)}\, \sigma\left(W_{\text{fc}}^{(l)}\, \text{LN}\left(a_i^{(l)} + h_i^{(l-1)}\right)\right). \end{aligned} \quad (7)$$

Based on this formulation, we track three representative hidden states: the attention projection output *attn_proj_out*, corresponding to $a_i^{(l)}$, the MLP input *mlp_fc_in*, serving as a key representation, and the MLP output *mlp_fc_out*, corresponding to retrieved knowledge $m_i^{(l)}$. We compute similarity scores between hidden states from a multi-hop query and those from its corresponding single-hop

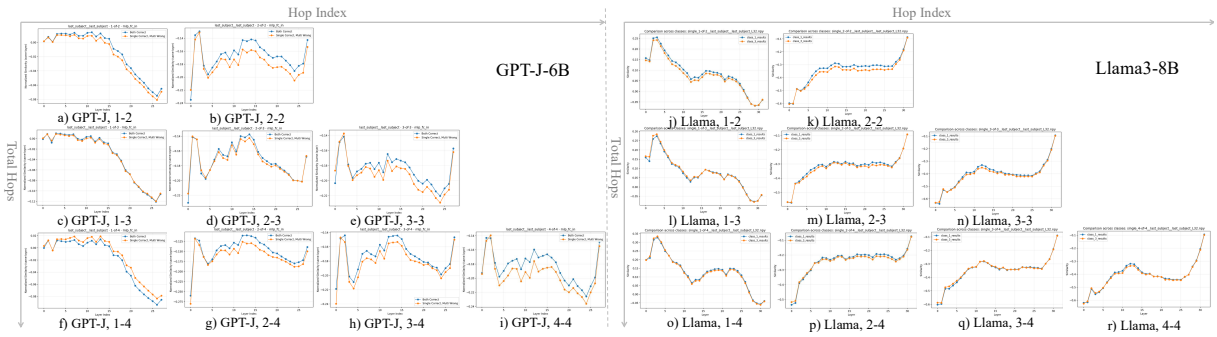


Figure 8: Nomarlized similarity of *mlp_fc_in* at the subject between multi-hop queries and their single-hop queries.

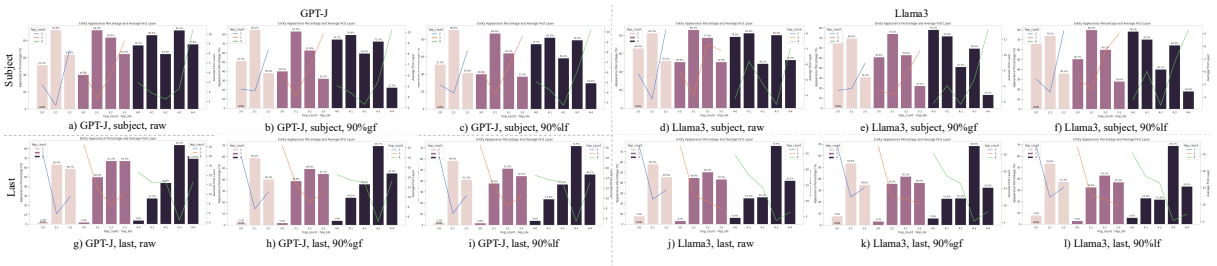


Figure 9: Patchscopes results for GPT-J (left) and Llama3 (right) on the Incorrect subset.

queries. As shown in Figure 7, cross-layer similarities are substantially weaker than same-layer similarities, likely due to mismatched representation spaces. We therefore restrict our analysis to same-layer comparisons between key tokens.

Additional Results Due to space constraints, we report additional hidden state similarity analyses under different settings in this section. At the subject token, we observe consistent patterns across different components in Figure 15-16, aligning with the results obtained from *mlp_fc_in* in Figure 8. At the final token in Figure 17-19, we also observe the same trend discussed in Section 6, the similarity gap between incorrect and correct cases increases as the hop index grows. These results further suggesting that although the model can correctly answer the corresponding single-hop questions, it fails to sufficiently recall the relevant multi-hop knowledge when answering the full multi-hop query.

A.4 Average Earliest Layer

In Section 4, we present results under the raw setting, while the results for the filtered settings are included in Table 3 and Table 4 due to space constraints. Consistent with the main results, *layer-order inversion* is observed across all settings, further suggesting that latent multi-hop reasoning in LLMs may not follow explicit reasoning circuits.

A.5 Patchscopes results on Incorrect cases

In Section 6, we use Patchscopes to analyze failures in **Incorrect** cases. Due to space constraints, we move some of the Patchscopes results to this appendix. As shown in Figure 9, deeper-hop entities are decoded less frequently in **Incorrect** cases at both the subject and last-token positions, compared with the **Correct** subset in Figure 3. This provides complementary evidence that incorrect multi-hop predictions can arise from insufficient activation of deeper-hop knowledge under the multi-hop context.

B Additional Analyses under the Probabilistic Recall-and-Extract Framework

Due to space constraints, this appendix further discusses previously observed experimental phenomena under the probabilistic recall-and-extract framework.

Knockout. Knockout (Geva et al., 2023) refers to the observation that blocking attention from earlier tokens to the last token in shallow layers significantly affects the probability of generating the correct answer, which has been interpreted as evidence that intermediate entities are propagated to the final position. Under our probabilistic recall-and-extract framework, such intervention reduces the probability of recalling deeper-hop entities at

Model	Token	2-hop			3-hop				4-hop				
		e_0	e_1	e_2	e_0	e_1	e_2	e_3	e_0	e_1	e_2	e_3	e_4
GPT-J	Subject	4.84	5.27	5.55	5.75	5.25	5.25	6.28	3.41	4.89	4.14	10.23	9.15
	Last	19.36	10.12	**9.62**	18.57	11.68	13.43	**6.38**	20.00	13.00	12.21	11.74	**11.93**
Llama3	Subject	4.73	6.34	6.98	5.34	4.92	7.01	8.86	4.40	5.93	3.33	6.94	10.26
	Last	28.92	13.21	**12.20**	29.23	14.68	17.32	**7.84**	27.88	21.00	15.39	6.96	**5.76**

Table 3: Average Earliest Layer of Entity Emergence under 90% Global Filter

Model	Token	2-hop			3-hop				4-hop				
		e_0	e_1	e_2	e_0	e_1	e_2	e_3	e_0	e_1	e_2	e_3	e_4
GPT-J	Subject	4.85	4.91	4.86	5.75	4.41	5.21	5.08	3.41	4.58	3.10	9.25	8.43
	Last	19.36	10.37	**9.52**	18.57	11.68	13.19	**6.54**	20.00	13.00	12.50	11.19	**11.57**
Llama3	Subject	4.73	5.43	5.83	5.42	4.67	6.68	6.79	4.28	5.88	2.54	6.36	6.98
	Last	28.92	13.33	**11.53**	29.23	15.87	16.94	**7.44**	27.88	20.85	16.13	8.01	**5.10**

Table 4: Average Earliest Layer of Entity Emergence under 90% Local Filter

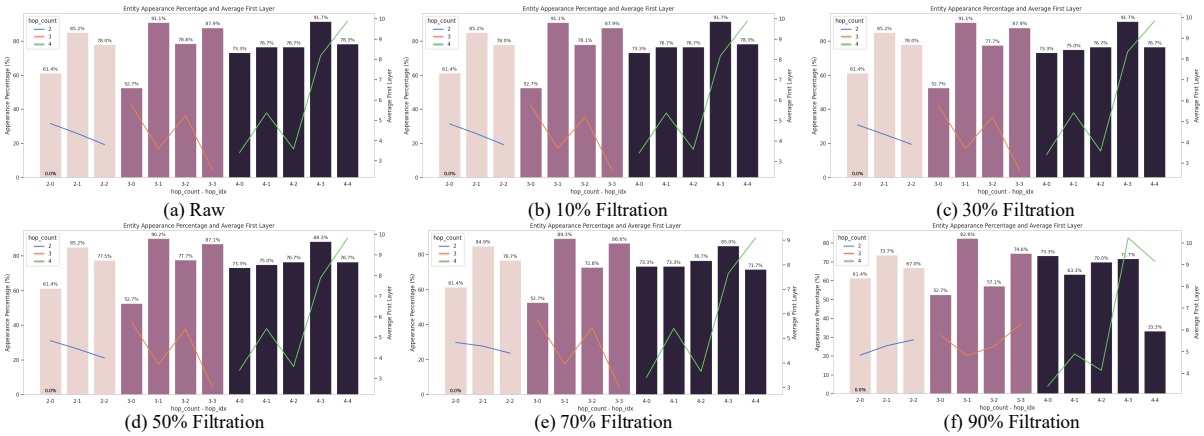
the last token, weakening the intuition-like signal for answer selection and thereby altering the answer distribution.

Back-patching. Back-patching (Biran et al., 2024) refers to the intervention of patching hidden states from deeper layers back into shallow layers, which has been shown to improve performance on multi-hop questions. This effect has previously been explained by the view that intermediate entities are propagated too late, which prevents the last token from handling subsequent hops. Under our probabilistic recall-and-extract framework, back-patching can be understood as compensating for insufficient shallow recall: injecting deeper representations into shallow layers increases the probability of recalling deeper-hop entities, strengthening the intuition-like signal for answer selection and thereby facilitating correct answer generation.

Chain-of-Thought. Chain-of-thought (CoT) prompting (Wei et al., 2022; Renze and Guven, 2024) improves multi-hop reasoning by explicitly generating intermediate steps, which is difficult to explain under circuit-based views that focus mainly on vertical, layer-wise computation. Under our probabilistic recall-and-extract framework, the newly generated tokens (e.g., bridge entities) strengthen the horizontal recall described in Section 5.2 by enriching the context available to subsequent tokens. This makes answer-relevant deeper-hop knowledge more likely to be recalled and selected, thereby shifting the model's output distribution toward the correct answer.

C Analysis of Missing Cases

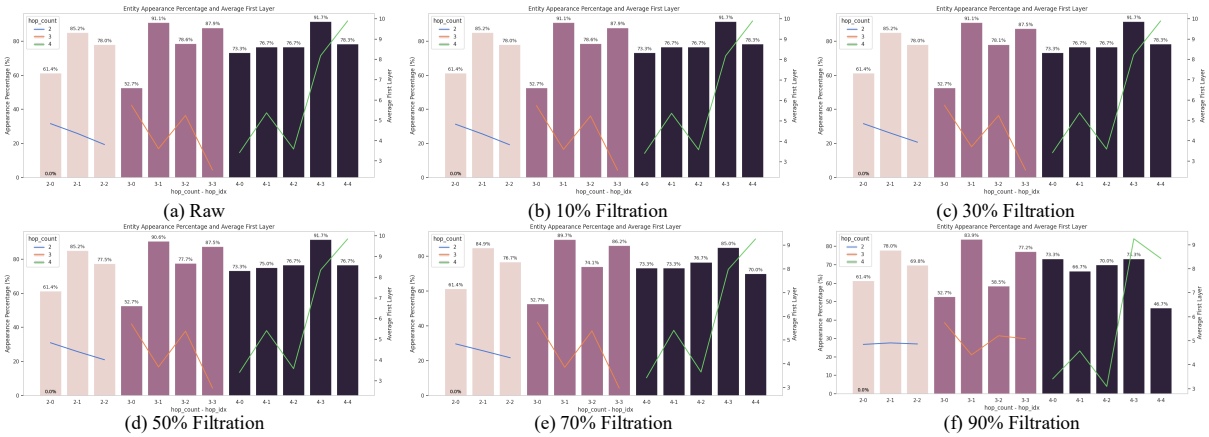
Due to space constraints, we provide the analysis of the **Missing** cases in this section. As shown in Figure 13 and Figure 14, the decoding rates of early-hop knowledge at the subject position are substantially lower in Missing cases than in both **Correct** and **Incorrect** cases. As a result, the model fails to recall the foundational knowledge required for multi-hop reasoning, preventing it from producing the correct answer.



(a) Raw (b) 10% Filtration (c) 30% Filtration

(d) 50% Filtration (e) 70% Filtration (f) 90% Filtration

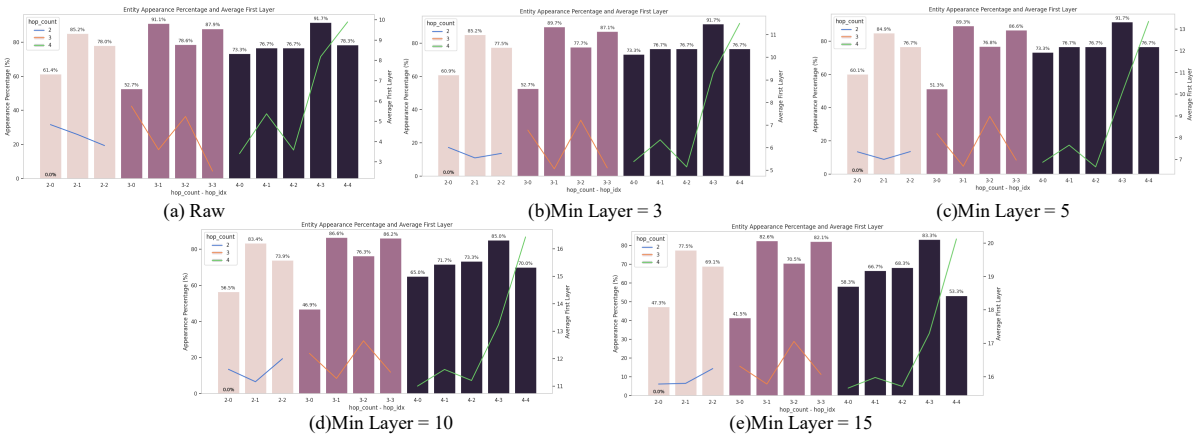
Figure 10: Patchscopes decoding results at subject token on GPT-J-6B under different global filtering ratios.



(a) Raw (b) 10% Filtration (c) 30% Filtration

(d) 50% Filtration (e) 70% Filtration (f) 90% Filtration

Figure 11: Patchscopes decoding results at subject token on GPT-J-6B under different local filtering ratios.



(a) Raw (b)Min Layer = 3 (c)Min Layer = 5

(d)Min Layer = 10 (e)Min Layer = 15

Figure 12: Patchscopes decoding results at subject token on GPT-J-6B with different minimum layer.

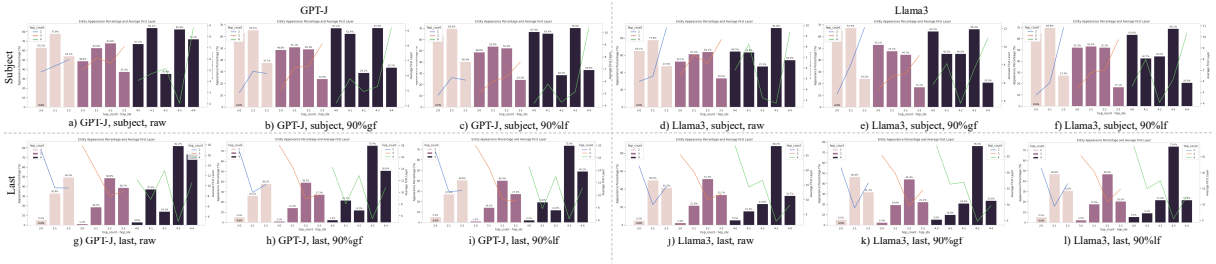


Figure 13: Patchscopes results for GPT-J (left) and Llama3 (right) on the Missing subset.

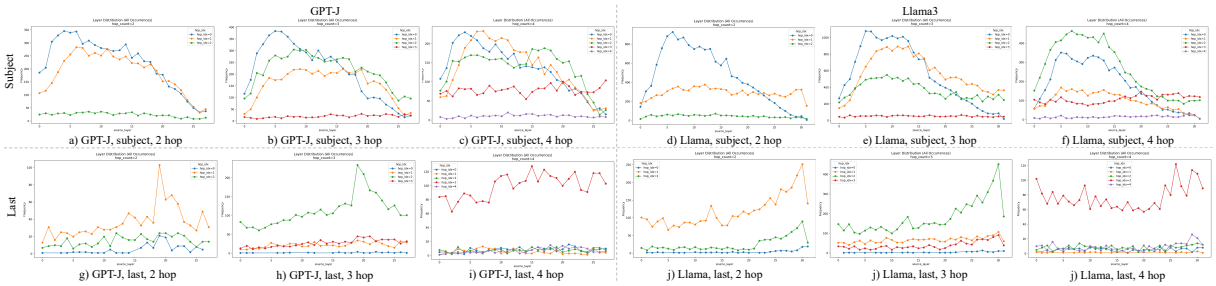


Figure 14: Layer-wise generation distributions produced by Patchscopes on the Missing subset.

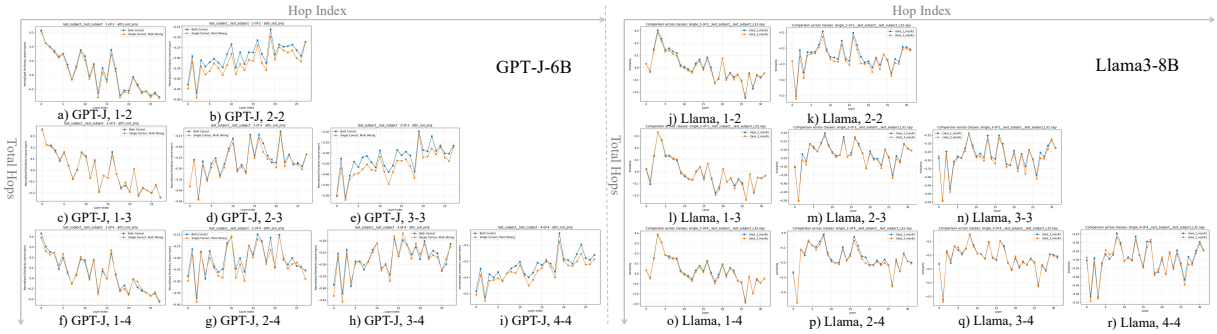


Figure 15: Nomarlized similarity of *attn_proj_out* at the subject between multi-hop queries and their single-hop queries.

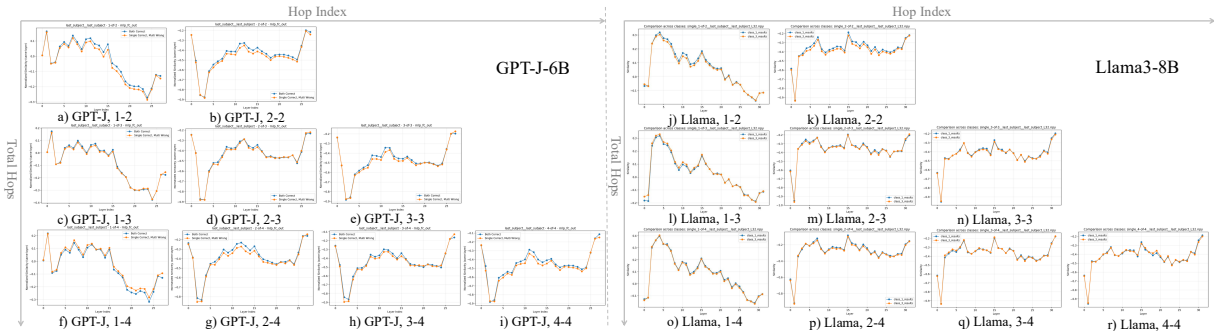


Figure 16: Nomarlized similarity of *mlp_fc_out* at the subject between multi-hop queries and their single-hop queries.

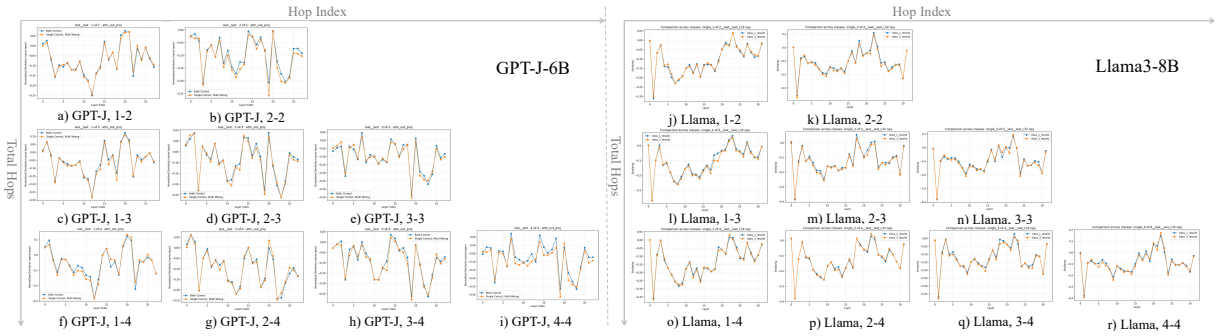


Figure 17: Nomarlized similarity of *attn_proj_out* at the last token between multi-hop queries and their single-hop queries.

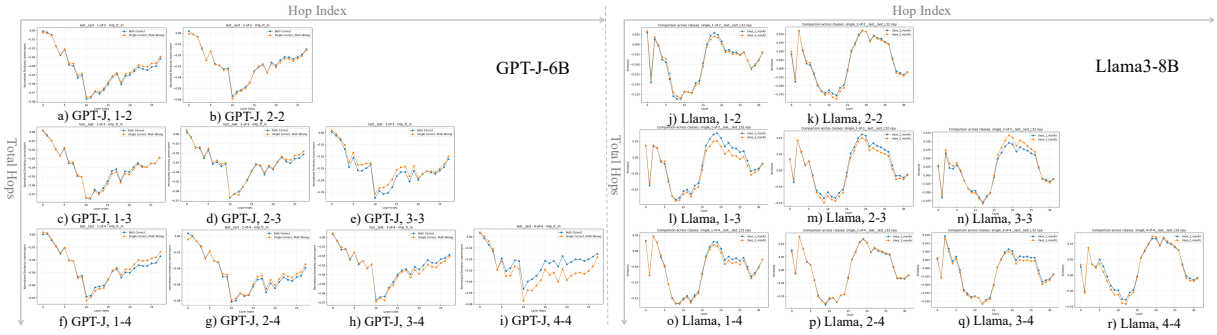


Figure 18: Nomarlized similarity of *mlp_fc_in* at the last token between multi-hop queries and their single-hop queries.

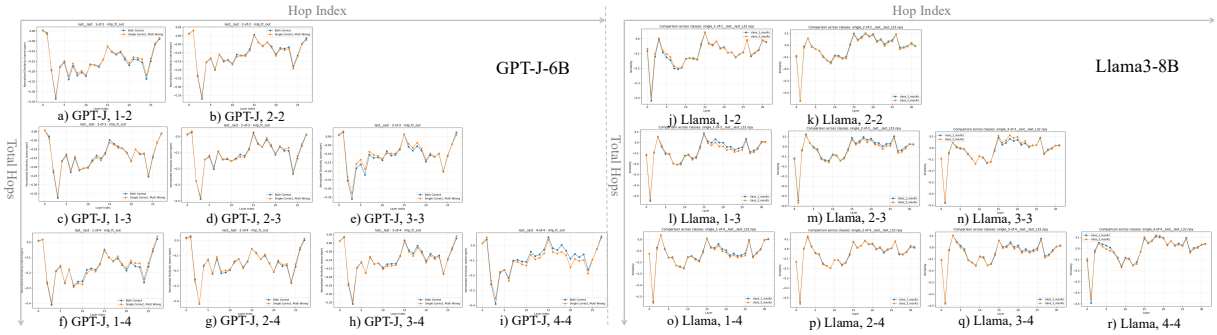


Figure 19: Nomarlized similarity of *mlp_fc_out* at the last token between multi-hop queries and their single-hop queries.