How Do Large Language Models Learn Concepts During Continual Pre-Training?

Barry Menglong Yao[♠] **Sha Li**[♡] **Yunzhi Yao**[♣] **Minqian Liu**[♡]
Zaishuo Xia[♠] **Qifan Wang**[♢] **Lifu Huang**[♠]
[♠]UC Davis [♡]Virginia Tech [♣]UCLA [♢]Meta AI
{bmyao,zsxia,lfuhuang}@ucdavis.edu
{shal,minqianliu}@vt.edu
{wqfcr}@meta.com

Abstract

Human beings primarily understand the world through concepts (e.g., *dog*), abstract mental representations that structure perception, reasoning, and learning. However, how large language models (LLMs) acquire, retain, and forget such concepts during continual pretraining remains poorly understood. In this work, we study how individual concepts are acquired and forgotten, as well as how multiple concepts interact through interference and synergy. We link these behavioral dynamics to LLMs' internal **concept circuits**, computational subgraphs associated with specific concepts, and incorporate graph metrics to characterize **circuit topology**. Our analysis reveals: (1) LLMs' concept circuits provide a non-trivial, consistent signal of concept learning and forgetting; (2) concept circuit topology evolves stage-wise, rising during concept acquisition, and transiently rising before declining during concept forgetting; (3) semantically similar concepts induce stronger interference than weakly related ones; (4) conceptual knowledge differs in their transferability, with some significantly facilitating the learning of others. Together, our findings provide a circuit-level view of concept learning dynamics and motivate concept-aware training strategies, such as Circuit-aware Experience Replay, which uses circuit topology to prioritize concepts vulnerable to forgetting[1].

1 Introduction

Humans organize perception and reasoning around **concepts**, abstract mental categories (e.g., *dog*) that enable generalization from individual observations to shared properties, relations, and actions (Harpaintner et al., 2018). Similarly, a defining characteristic of large language models (LLMs) is their ability to abstract conceptual knowledge from vast corpora, encoding it within internal *concept-level* representations that facilitate downstream reasoning and generation(Aljaafari et al., 2024; Wang et al., 2024b; Park et al., 2023). As new concepts continually emerge and models are updated through continual pre-training, it becomes essential to understand how reliably new concepts are acquired and prior concepts are retained, and how to make concept learning more efficient in the presence of interactions among the many concepts and types of conceptual knowledge.

Prior studies have examined concept learning in LLMs from two angles. One line of work uses prompt-based **knowledge probing** to test whether specific conceptual properties or relations (e.g., *commonsense facts*) can be elicited from a pretrained model (Gu et al., 2023; Liao et al., 2023; Shani et al., 2023; Zheng et al., 2024; Peng et al., 2022; Xu et al., 2024a), while another leverages **mechanistic interpretability** tools to localize internal structures associated with conceptual information (Aljaafari et al., 2024; Wang et al., 2024b). However, these efforts provide only a partial view of concept learning: they typically probe isolated pieces of conceptual knowledge that models may already possess, rather than systematically characterizing how concepts are acquired, consolidated, and forgotten during continual pre-training.

In this work, we take a step toward filling this gap by asking two more research questions: (1) *how internal concept representations correlate with concept acquisition and forgetting*, and (2) *how they relate to interference and synergy among multiple concepts and types of knowledge during joint training*. Answering these questions matters not only for interpretability, but also for practice: they can inform concept-aware continual pre-training decisions, such as how much training to allocate to previously learned concepts to mitigate forgetting and to newly introduced concepts to facilitate efficient acquisition, how to schedule and reorder

[1]The code is publicly available at https://github.com/PLUM-Lab/llm_circuit_for_learning_dynamics.

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing, pages 55302–55325
October 24-29, 2026 ©2026 Association for Computational Linguistics

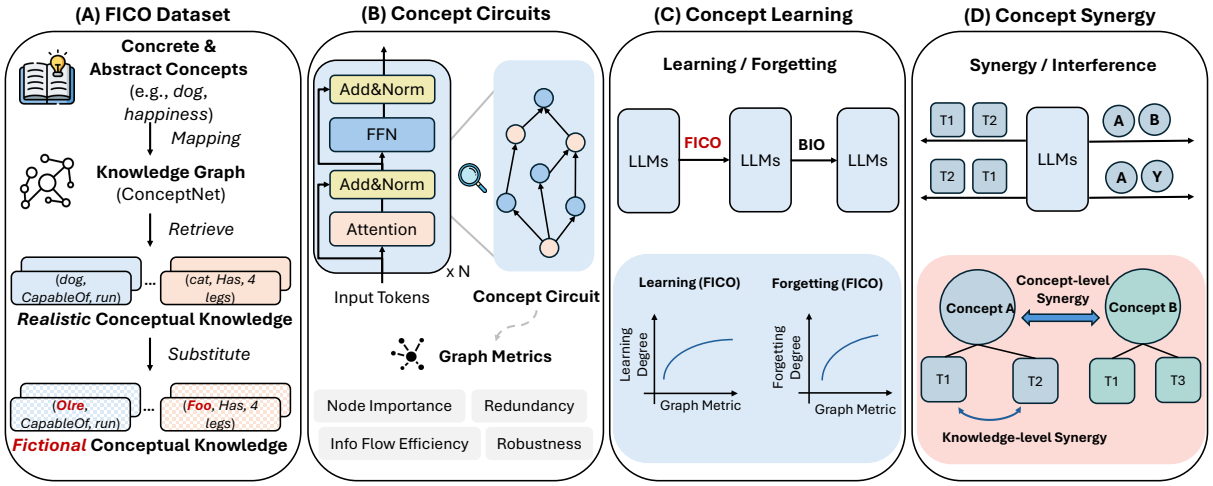


Figure 1: (A) Construct the FICO dataset based on ConceptNet. (B) Extract **Concept Circuits**, LLM computational subgraph associated with individual concepts, and characterize their structure using graph metrics. (C) Analyze **concept learning and forgetting dynamics** in two-stage continual pretraining. (D) Study **synergy and interference** across concepts (e.g., A, B, Y) and knowledge type (e.g., $T1$ and $T2$).

training data, and how to mitigate destructive interference among semantically related concepts.

To enable a controlled study, we introduce the FICO (**FI**ctional **CO**ncept) dataset, built from ConceptNet-derived conceptual relations but mapped to synthetic, non-existent concept names, preserving realistic knowledge structure while making the concepts novel to the model. We then conduct controlled continual pre-training experiments and link behavioral changes to internal mechanisms by extracting **Concept Circuits**, computational subgraphs associated with each concept, using circuit identification methods and characterizing their topology over time with graph metrics. This joint analysis allows us to trace how concept-level behavior (e.g., learning and forgetting measures) co-evolves with circuit topology, and how cross-concept relatedness and knowledge-type ordering shape interference and transfer (Figure 1).

Our results reveal several meaningful and systematic patterns in concept learning dynamics in LLMs: **(1)** circuit topology is non-trivially correlated with both concept learning and forgetting, suggesting a trade-off between acquisition strength and subsequent stability: more centralized and redundant circuit structures are associated with stronger acquisition, but also with greater forgetting under continual training. This trade-off is also reflected behaviorally, as concepts with larger learning gains tend to exhibit greater forgetting during subsequent training. **(2)** Concept circuits follow stable stage-wise temporal trajectories across multiple graph metrics in both training stages: their topology rises and then stabilizes as concepts are acquired, whereas during continued training on new domains it transiently rises before gradually declining and stabilizing, suggesting distinct phases of concept consolidation and subsequent reorganization. **(3)** At the level of multi-concept learning, semantically similar concepts interfere more strongly than weakly related concepts, and different types of conceptual knowledge differ substantially in their transferability; e.g., pretraining on *Hyponym & Hypernym* knowledge improves subsequent learning performance on *Synonym & Antonym* knowledge by 63.74%. Together, these findings offer a circuit-level view of concept learning dynamics and show that the resulting circuit-based signals can be translated into actionable training strategies, such as circuit-aware replay, to improve continual pre-training stability.

Our contributions are summarized as follows:

- We present the first systematic study of concept acquisition and forgetting in LLMs during continual pretraining, revealing consistent correlations between concept circuits, learning dynamics, and forgetting behavior, suggesting these signals as indicators of future learning and forgetting.

- We present a relational perspective on knowledge learning in LLMs, revealing interference and synergy patterns across concepts and knowledge types during joint training, motivating interference-aware data scheduling.

- Our analysis informs actionable concept-aware training strategies for continual pre-training. As a proof of value, we instantiate a circuit-aware replay strategy to improve knowledge retention.

2 Dataset Construction

We follow previous studies (Wang et al., 2024b) and define a **concept** (e.g., *dog*) as an abstraction over the world that captures the shared features and essential characteristics of a class of entities. **Conceptual knowledge** (e.g., *a dog has four legs*) refers to factual or relational information associated with a concept, reflecting familiarity with and understanding of its properties and relations (Wang et al., 2024a). Each concept is therefore linked to a set of conceptual knowledge that collectively describe its semantic content. To study concept learning across different levels of concreteness, we consider both **concrete concepts** that are grounded in direct sensory experience (e.g., *dog*), and **abstract concepts** which are not directly perceptible (e.g., *love*). We sample 500 concrete concepts from the THINGS dataset (Hebart et al., 2023) and 500 abstract concepts from the Concreteness Ratings dataset (Brysbaert et al., 2014). For each concept, we retrieve associated conceptual knowledge from ConceptNet (Speer et al., 2017), a large-scale knowledge graph of millions of concepts and 34 typed relations.

To mitigate influence of pre-existing knowledge encoded in LLMs, we replace each real concept name with a fictional name generated by GPT-5 (Hurst et al., 2024) and perform relation filtering and canonicalization, as detailed in Appendix A.2. This process forms our FICO, **FI**ctional **CO**ncept dataset, illustrated in Figure 1(A), with statistics in Appendix B. Following prior work (Zucchet et al., 2025), we use GPT-5 to generate multiple natural-language templates for each relation type (e.g., "*{concept} has the ability to*" for *CapableOf*), converting each knowledge triple (e.g., (dog, CapableOf, run)) into prefix-target examples. For evaluation, we sample 500 concepts and use a *disjoint* template pool to construct test instances with unseen surface forms, enabling assessment of whether LLMs learn underlying relations rather than memorizing specific templates.

3 How Internal Concept Representations Correlate with LLM Learning and Forgetting Dynamics?

3.1 Concept Circuit as Internal Concept Representation

Given a knowledge triple $k_{ij} = (c_i, r_{ij}, o_{ij})$, where c_i is a subject concept (e.g., *dog*), r_{ij} is a relation type (e.g., *HasProperty*), and o_{ij} is an object which can be another concept or a descriptive phrase (e.g., *fleas* or *four legs*), a **Knowledge Circuit** is defined as the minimal computational subgraph in a pretrained LLM that can faithfully predict the target object o_{ij} conditioned on a textual prefix converted from the subject–relation pair (c_i, r_{ij}) (Yao et al., 2025b). We adapt this notion to concept-level analysis and define a **Concept Circuit** for concept c_i as the computational subgraph that jointly and faithfully predict all conceptual knowledge $\mathcal{K}_i = \{k_{i0}, k_{i1}, \ldots, k_{ij}, \ldots\}$ associated with c_i. Unlike a union of independently extracted circuits for individual facts, a Concept Circuit is identified at the level of $\mathcal{K}_i$ and captures the shared computation underlying the conceptual knowledge.

To extract concept circuits at different checkpoints during continual training, we use EAP-IG (Hanna et al., 2024) as our circuit identification method. EAP-IG assigns an importance score to each edge while balancing computational efficiency with attribution faithfulness[2]. Given the edge importance scores, we construct a circuit by selecting the top-scoring edges such that the resulting subgraph preserves at least 70%[3] of the full model's performance on the corresponding concept.

Topology of Circuit Graph Previous work has shown that the geometric and topological properties of LLM circuits and features can provide insight into model behaviors such as grokking, reasoning, and learning. (Humayun et al., 2024; Zhou et al., 2025; Raj; Li et al., 2025; Ou et al., 2025). Motivated by these findings, we compute four families of standard graph metrics to characterize the topology of concept circuits and their relationship to concept learning dynamics in LLMs: **(1) Density**, defined as the ratio of existing edges to the

[2]More details for the implementation of EAP-IG can be found in Appendix C.

[3]Following prior circuit-level interpretability work (Ou et al., 2025; Yao et al., 2025b), we use 70% as a heuristic preservation threshold and provide an 80% ablation in Section 3.3. Thresholds $\geq$ 90% often yield overly large, less interpretable circuits.

maximum possible number of edges. Higher density reflects more redundant connections (Humayun et al., 2024; Newman, 2010). **(2) Eigenvector Centrality**, whose standard deviation quantifies how unevenly structural influence is distributed across nodes. Higher variance indicates a more concentrated hub structure (Newman, 2010). **(3) Global Efficiency**, i.e., the average inverse shortest-path distance between node pairs. Higher global efficiency indicates that signals can propagate more efficiently across the graph (Latora and Marchiori, 2001). **(4) k-core**, whose average number captures the depth of a graph's densely connected core and is used as a proxy for resilience to disruption in graph (Laishram et al., 2018; Seidman, 1983). We select these four metrics because, in the graph analysis literature, they are standard measures of graph redundancy, node importance, information flow efficiency, and graph robustness, respectively.

3.2 Experiment Design

To quantify how internal concept representations (i.e., concept circuits) relate to concept acquisition and forgetting, given a pre-trained LLM π_0, we conduct a two-stage [4] continual pretraining setup. In **Stage 1 (Concept Acquisition)**, we continually train on the FICO training set to learn novel concepts, yielding π_1, which is used to analyze concept learning dynamics. In **Stage 2 (Forgetting Induction)**, we further train π_1 on the BIO dataset (Allen-Zhu and Li, 2023), a standard pretraining dataset used in previous LLMs knowledge acquisition studies (Allen-Zhu and Li, 2023; Zucchet et al., 2025), yielding a new model π_2, which is used to analyze concept forgetting dynamics. We experiment with GPT-2 Large (Radford et al., 2019), LLaMA-3.2-1B, LLaMA-3.1-8B (Dubey et al., 2024), and Qwen3-4B-Base (Yang et al., 2025).[5] In both stages, we concatenate the prefix and target phrase, as described in Section 2, and optimize the model using next-token prediction.

For evaluation, we provide only the prefix on the FICO test set and compute concept learning/forgetting degrees from the model's gold-target logit, as defined below. We use **Spearman's rank correlation coefficient** (ρ) (Spearman, 1961) to analyze the correlation between concept learning/forgetting dynamics and concept circuit topology.[6]

Definition 1 (Knowledge Learning/Forgetting Degree)**.** *For a knowledge triple $k = (c, r, o)$ with prefix $x = T(c, r)$, let $s_\pi(k) = \text{logit}_\pi(o \mid x)$ denote the gold-target logit.*[7] *The* **knowledge learning degree** *and* **knowledge forgetting degree** *of k are defined as $\Delta_{\text{learn}}(k) = s_{\pi_1}(k) - s_{\pi_0}(k)$ and $\Delta_{\text{forget}}(k) = s_{\pi_1}(k) - s_{\pi_2}(k)$, respectively.*

Definition 2 (Concept Learning/Forgetting Degree)**.** *For a concept c with associated triples $\mathcal{K}_c$, its* **concept learning degree** *and* **concept forgetting degree** *are defined as $\Delta_{\text{learn}}(c) = \frac{1}{|\mathcal{K}_c|}\sum_{k\in\mathcal{K}_c}\Delta_{\text{learn}}(k)$ and $\Delta_{\text{forget}}(c) = \frac{1}{|\mathcal{K}_c|}\sum_{k\in\mathcal{K}_c}\Delta_{\text{forget}}(k)$, respectively.*

3.3 Experiment Finding

Finding 1. *Concepts exhibit substantial cross-concept variation in both learning degree and forgetting degree, indicating that LLMs acquire and forget different concepts to markedly different extents under the same training regime.*

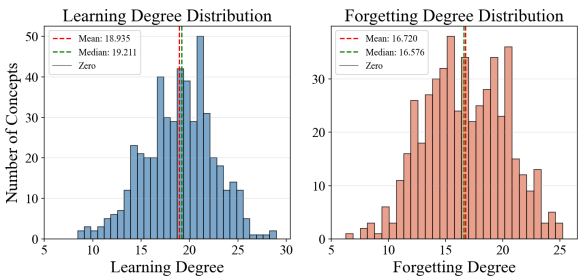


Figure 2: Distribution of learning and forgetting degree across concepts.

Concept Learning/Forgetting Degree Distribution. Figure 2 shows the distribution of concept learning (logit increase) and forgetting (logit decrease) degrees across 500 concepts on the FICO test set. The distribution is unimodal but widely spread, indicating substantial variability in how concepts are learned and forgotten. Practically, some concepts are learned more readily and robustly, while others require greater effort to learn

[4] We further evaluate longer continual updates and observe consistent trends; see Appendix D.

[5] We observe similar trends across models. Due to space constraints, we present GPT-2 results in the main paper and defer LLaMA and Qwen results to Appendix E.

[6] See Appendix F for additional experiment details.

[7] Following prior work (Heimersheim and Nanda, 2024; Hanna et al., 2024), we use logits as a fine-grained indicator of LLM learning and report log-probability results in Appendix G.

Type	Metric	Learning	Forgetting
Circuit Topology (70%)	Density	0.4474	0.4585
Circuit Topology (70%)	k-core	0.4468	0.4587
Circuit Topology (70%)	Eigenvector Centrality	0.4403	0.4459
Circuit Topology (70%)	Global Efficiency	0.4085	0.4322
Circuit Topology (80%)	Density	0.4723	0.4785
Circuit Topology (80%)	k-core	0.4475	0.4610
Circuit Topology (80%)	Eigenvector Centrality	0.3235	0.3314
Circuit Topology (80%)	Global Efficiency	0.4662	0.4747
Activation	Mean EAP-IG	0.2680	0.2784
Activation	STD EAP-IG	0.2398	0.2494
Activation	Max EAP-IG	0.1575	0.1342
Random Topology	Density	-0.0078	-0.0255
Random Topology	k-core	-0.0458	-0.0437
Random Topology	Eigenvector Centrality	0.0473	0.0737
Random Topology	Global Efficiency	-0.0559	-0.0769

Table 1: Correlation between learning/forgetting degree and circuit metrics (Spearman ρ)

and retain. This observation motivates our subsequent analysis, which seeks to identify indicators that can predict the differing learning and forgetting behaviors across concepts.

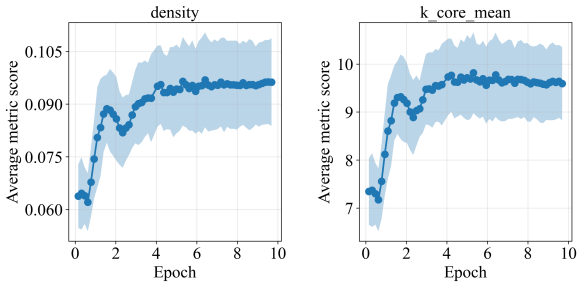
Finding 2. *Concept learning degree and forgetting degree shows non-trivial, consistent correlations with multiple circuit graph metrics, suggesting that circuit topology can serve as an informative indicator of concept learning and forgetting dynamics.*

Correlation between Concept Learning/Forgetting and Concept Circuits. As shown in Table 1, our circuit topology metrics exhibit stronger correlations [8] with concept learning degree ($\rho \approx 0.41$–0.45) than either edge-level EAP-IG statistics (Hanna et al., 2024) or random topology metrics ($|\rho| < 0.08$) [9], indicating that circuit topology may be systematically associated with how concepts are acquired. For example, positive correlations with eigenvector centrality and k-core suggest that circuits with centralized bottlenecks and stable structural cores tend to achieve larger logit gains during learning. Notably, these same topological properties are also associated with greater vulnerability under continual training. In particular, concepts whose circuits are dominated by a small set of influential hub nodes, reflected by high eigenvector centrality, tend to exhibit larger forgetting degrees, potentially because these centralized components are more susceptible to perturbation during subsequent training.

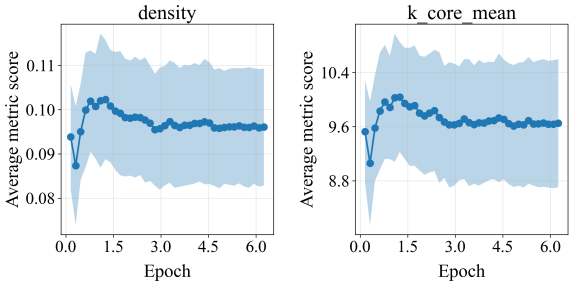
This structural trade-off is also reflected behaviorally: concepts that are learned more strongly also tend to be forgotten more strongly under subsequent training, revealing a positive correlation between learning degree and forgetting degree, as detailed in Appendix J. Furthermore, these correlations remain comparable when increasing the circuit performance-preservation threshold from 70% to 80%, suggesting that the observed circuit-behavior associations are robust to the threshold choice. Taken together, these results indicate that circuit topology captures informative signals of both concept learning and forgetting dynamics in LLMs. Such signals may provide practical guidance for concept-aware continual pretraining decisions, such as how to allocate training effort between newly introduced concepts and previously learned concepts, as further explored in Section 5.

3.3.1 Experiment Finding

Finding 3. *LLMs exhibit consistent stage-wise temporal patterns across multiple circuit graph metrics: the metrics increase and then stabilize during concept acquisition, and show a transient increase followed by a gradual decrease and stabilization during concept forgetting.*



(a) Concept Acquisition.



(b) Concept Forgetting.

Figure 3: Evolution of circuit graph metrics during the learning and forgetting stages, shown for density and mean k-core; all four metrics are in Appendix K.

Evolution of Circuit Topology. We track how circuit topology evolves during both training stages. Figure 3a shows the learning stage: all graph

[8] p-values are provided in Appendix H

[9] See baseline metrics in Appendix I.

metrics rise sharply during early learning (within roughly the first 1.5 epochs), continue to increase gradually with minor fluctuations, and stabilize after epoch $\sim$5, suggesting that concept acquisition is accompanied by the progressive formation and consolidation of circuit structure. Figure 3b shows a consistent temporal pattern of circuit graph metrics during the forgetting stage: an initial increase followed by a gradual decrease and eventual stabilization. The early increase may indicate transient entanglement of previously learned concept circuits with newly introduced knowledge during continued training, while the subsequent decrease and stabilization may reflect structural relaxation and convergence to a weaker representation under interference.

4 How Do Interference and Synergy Arise Across Concepts and Conceptual Knowledge During Joint Training?

The analyses above suggest a trade-off between acquisition strength and vulnerability under continual training. This raises a natural question: is such vulnerability intrinsic to individual concepts, or is it also shaped by interactions among concurrently learned concepts?

4.1 Interference and Synergy across Concepts

4.1.1 Experiment Design

To examine cross-concept interactions during joint training, we construct relatedness-based groups for each target concept. Relatedness is defined as the cosine similarity between embeddings of the concatenated conceptual knowledge associated with each concept, computed using Qwen3-Embedding-4B (Yang et al., 2025). For each concept c, we rank all other concepts by similarity and select the top-K, middle-K, and bottom-K concepts as the highly, moderately, and weakly related groups, respectively, with $K = 100$. We then perform joint training for each concept c under three settings, where c is trained together with concepts from one of the three relatedness groups[10]. Performance is evaluated using (1) the average logit assigned to target objects and (2) the corresponding average probability. This procedure is repeated across all concepts in the FICO test set to characterize cross-concept interactions.

[10] A controlled experiment disentangling semantic relatedness from surface-level overlap shows that the interference persists at zero lexical overlap; see Appendix L.

	Avg. probability			Avg. logit		
Model	Most	Med.	Least	Most	Med.	Least
GPT-2 Large	0.575	0.672	**0.759**	14.37	14.90	**16.24**
LLaMA-3.2-1B	0.754	0.811	**0.877**	18.23	18.55	**20.28**
LLaMA-3.1-8B	0.781	0.846	**0.889**	23.65	24.18	**25.76**
Qwen3-4B-Base	0.807	0.828	**0.881**	26.74	27.10	**28.73**

Table 2: Concept interference across models: average target probability and logit after co-training targets with its most, moderately, or least related concepts. (Appendix E).

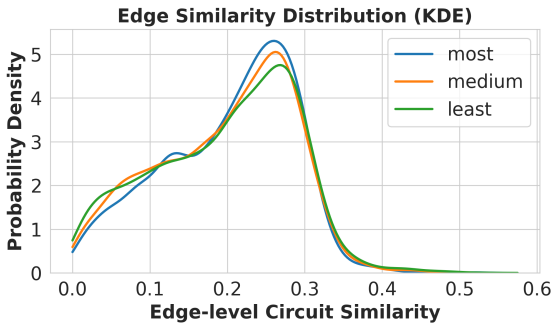


Figure 4: Distribution of Jaccard similarity between circuits of co-trained concept pairs, by relatedness group.

4.1.2 Experiment Finding

Finding 4. *Training with highly related concepts yields lower performance than training with weakly related concepts, indicating stronger interference among semantically similar concepts during joint learning.*

Table 2 shows a clear and consistent dependence on semantic relatedness. Across both evaluation metrics (average logit and average probability), training with weakly related concepts achieves the highest performance (75.9%), substantially outperforming training the highly related (57.5%) and moderately related (67.2%) concepts. These results demonstrate that cross-concept interactions meaningfully affect how effectively a target concept can be acquired in a multi-concept setting.

To explore a potential mechanism associated with this effect, we analyze the degree of overlap between the internal representations of co-trained concepts. For each target concept c, we pair it with each of its related concepts drawn from three semantic similarity groups: highly related, moderately related, and weakly related. For each pair (c, c'), we obtain their concept circuits and compute the Jaccard similarity between their edge sets. Figure 4 visualizes the resulting similarity distributions using KDE plots. The highly related setting

exhibits a sharper, higher-density peak at moderate circuit similarity, indicating consistent reuse of overlapping computational pathways between target and auxiliary concepts, which may induce sustained representational competition during joint training, consistent with the stronger interference we observe. Instead, the moderately and weakly related settings show lower and more dispersed similarity distributions, reflecting reduced circuit overlap, consistent with the weaker interference observed in these settings. This observation motivates interference-aware data scheduling that reduces circuit similarity among co-trained concepts within the same batch.

4.2 Interference and Synergy across Knowledge

4.2.1 Experiment Design

Beyond concepts, we ask whether *different types of conceptual knowledge* can also exhibit interference or synergy, even when they describe the *same* concept. We consider five high-level semantic knowledge categories: (1) Hyponym & Hypernym (HAH), (2) Synonym & Antonym (SAA), (3) Meronym & Holonym (MAH), (4) Property & Affordance (PAA), and (5) Spatial Relation (SR), as detailed in Appendix A. To study interactions across knowledge types, we adopt a pairwise continual-training setup. In Stage 1, the model is trained on either a source knowledge type S or the BIO dataset (Allen-Zhu and Li, 2023), which does not contain conceptual knowledge, for the same training steps. In Stage 2, it is continually trained on a target knowledge type T, yielding the checkpoints $\pi_{S\to T}$ and $\pi_{\mathrm{BIO}\to T}$, respectively. We define **paired transferability** from S to T as

$$T(S \to T) = \frac{\mathrm{logit}(T \mid \pi_{S\to T}) - \mathrm{logit}(T \mid \pi_{\mathrm{BIO}\to T})}{|\mathrm{logit}(T \mid \pi_{\mathrm{BIO}\to T})|}. \quad (1)$$

Positive $T(S \to T)$ indicates *synergy*, where learning S facilitates subsequent learning of T, while negative values indicate *interference*.

4.2.2 Experiment Finding

Finding 5. *Substantial and asymmetric transfer effects emerge across knowledge types, where pretraining on one type can facilitate learning of another, with highly directional and uneven benefits across ordered pairs.*

Figure 5 provides a fine-grained view of pairwise synergy among the different knowledge types,

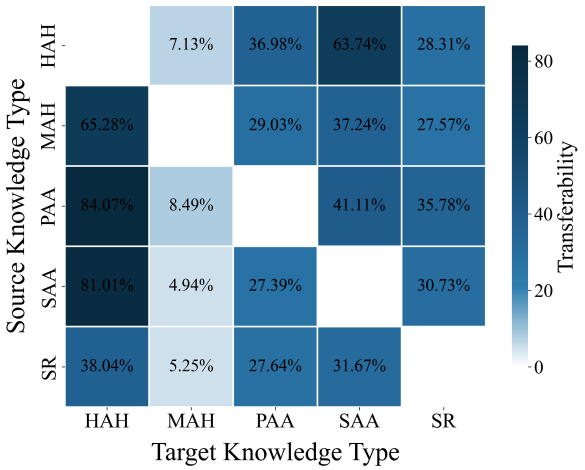


Figure 5: Paired transferability across knowledge

where each cell shows the paired transferability $T(S \to T)$ when a *source* type S is trained before a *target* type T. While all five categories encode concept-related information, they capture distinct semantic facets, leading to non-redundant and uneven transfer behaviors. The heatmap reveals heterogeneous and directional effects: for example, pretraining on Property & Affordance (PAA) yields substantial gains when transferring to Hyponym & Hypernym (HAH) and Synonym & Antonym (SAA), suggesting that learning functional and attribute-level regularities can effectively scaffold the acquisition of more abstract relational structures. In contrast, the reverse directions exhibit markedly weaker transfer, highlighting the asymmetry of these interactions. Moreover, some target types, such as Meronym & Holonym (MAH), show relatively small improvements across most sources, indicating higher intrinsic learnability and reduced sensitivity to prior knowledge-type pretraining. Overall, these findings suggest practical guidance for future training curricula, such as reordering training data to place knowledge types with strong positive transfer earlier, thereby encouraging synergy and efficiency of downstream concept learning.

To explore a potential mechanism behind this asymmetry, we extract one EAP-IG circuit per knowledge type from both Stage-2 checkpoints. For each direction $S \to T$, we compute **convergence**, the Jaccard similarity between the circuit extracted from $\pi_{S\to T}$ and the one extracted from $\pi_{\mathrm{BIO}\to T}$, i.e., how closely the transferred circuit resembles the one formed without a related source type. As shown in Figure 6, convergence

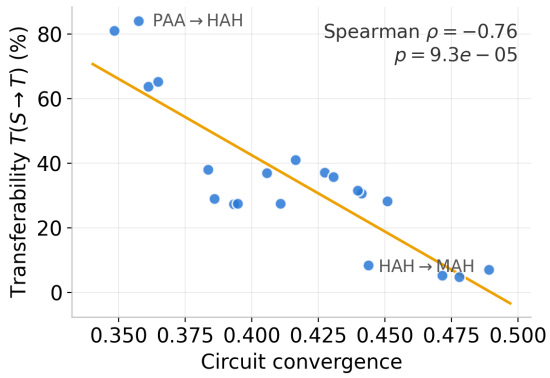


Figure 6: Transferability versus circuit convergence across the 20 ordered knowledge-type directions (GPT-2 Large). Directions whose $\pi_{S\to T}$ circuit remains least similar to the $\pi_{\text{BIO}\to T}$ circuit transfer best.

is negatively correlated with transferability across all 20 ordered directions (Spearman $\rho = -0.76$, $p = 9.3 \times 10^{-5}$): the best-transferring directions, such as PAA→HAH, are those whose $\pi_{S\to T}$ circuits deviate most from the $\pi_{\text{BIO}\to T}$ circuit. Directions with stronger transfer thus arrive at target circuits that depart most from the one formed without a relevant source, indicating that the source type reshapes the target computation rather than leaving it to re-form as it would from BIO alone. This connects Findings 4 and 5 at the circuit level: cross-knowledge interactions leave structural traces whether the related knowledge is trained jointly (Finding 4) or beforehand (Finding 5).

Method	Replay Ratio	Accuracy(%)	Hit@5(%)	Logit
w/o Replay	0	1.93	7.23	3.93
Random Replay	0.01	11.76	19.96	6.77
Gradient Replay	0.01	15.17	28.35	7.96
Circuit Replay	0.01	22.49	38.78	8.85
Random Replay	0.05	40.39	61.11	10.88
Gradient Replay	0.05	41.96	59.76	11.17
Circuit Replay	0.05	51.58	71.06	12.72
Random Replay	0.10	60.94	77.36	12.98
Gradient Replay	0.10	59.57	74.46	12.95
Circuit Replay	0.10	69.15	82.74	14.26
Random Replay	0.50	82.48	91.42	16.34
Gradient Replay	0.50	81.90	90.44	16.81
Circuit Replay	0.50	86.48	93.04	17.42

Table 3: Retention performance on the AntiLeak dataset under continual pretraining.

5 How Circuit Analysis Can Inform Better Continual Pretraining?

As shown in Finding 2, circuit topology metrics provide informative signals of concept forgetting during continual pre-training of LLMs. Leveraging this insight, we derive a simple yet actionable training strategy, **Circuit-aware Experience Replay**, that prioritizes concepts predicted to be more susceptible to forgetting and allocates additional training effort to them (e.g., via experience replay (Isele and Cosgun, 2018)).

	GPT-2 Large		LLaMA-3.2-1B		LLaMA-3.1-8B		Qwen3-4B-Base	
Metric	Learn	Forget	Learn	Forget	Learn	Forget	Learn	Forget
Density	0.447	0.459	0.399	0.411	0.349	0.386	0.333	0.416
k-core	0.447	0.459	0.375	0.381	0.349	0.389	0.319	0.409
Eigenvector Centrality	0.440	0.446	0.163	0.175	0.148	0.182	0.328	0.394
Global Efficiency	0.409	0.432	0.332	0.372	0.201	0.207	0.285	0.304

Table 4: Correlation between learning (Learn) and forgetting (Forget) degree and circuit topology metrics across model families (Spearman ρ). All p-values are below 10^{-3}; full p-values are reported in Table 7.

Method Given a model π_{learn} that has learned a sequence of concepts $\{c_0, c_1, \dots\}$, our goal is to further train the model on a new dataset while preserving performance on previously learned concepts. To this end, we extract concept circuits for all learned concepts and compute their corresponding circuit topology metrics. Based on topology metric[11], we select the top $p\%$ of concepts with the largest topology metric values and replay their associated training samples during continual pretraining on the new dataset. [12].

Experiment To evaluate this strategy in a realistic setting, we use the AntiLeak dataset (Wu et al., 2025), which contains newly introduced real-world knowledge not previously learned by existing LLMs. We train GPT-2 Large on AntiLeak to obtain π_{learn}, continue training on the BIO dataset (Allen-Zhu and Li, 2023), and measure retention on AntiLeak with **Logit**, **Accuracy**, and **Hit@5** (Appendix N). We compare **Circuit-aware Experience Replay** against **Random Replay** and **Gradient Replay**, which ranks concepts by per-sample gradient norm at the final stage-1 checkpoint. As shown in Table 3, circuit-aware replay outperforms both baselines at every replay ratio and metric (e.g., 22.49% vs. 15.17% accuracy at ratio 0.01). The gap reflects what the two signals measure: gradient norm indicates difficulty rather than forgetting risk, so gradient replay spends its budget on weakly learned facts with little to protect.

[11] As shown in Table 1, the Density metric exhibits the strongest correlation with forgetting, and is therefore used to rank concepts.

[12] Circuit extraction computational overhead is in Appendix M.

6 Robustness and Ablations

We assess the stability of our findings along three additional axes: model family and scale, and the continual-training horizon.

Model family and scale We repeat RQ1 and RQ2 on LLaMA-3.2-1B, LLaMA-3.1-8B, and Qwen3-4B-Base. As shown in Table 4, density and k-core remain at $\rho \approx 0.3$ or above for both learning and forgetting degree on all four models, while eigenvector centrality is as predictive as density on Qwen3-4B-Base but weaker on LLaMA, whose circuits exhibit uniformly small eigenvector centrality across concepts ($\approx$0.027 vs. $\approx$0.040 in GPT-2), compressing the metric's range. The interference and transfer patterns also replicate across families. As summarized in Table 2, co-training a target with its least related concepts yields the highest target probability and logit on all four models. Knowledge-type transferability likewise remains heterogeneous and directional (Appendix E).

Longer training horizon Finally, we extend the two-stage FICO→BIO setup to a four-stage FICO→BIO→FICO→BIO sequence, testing whether our conclusions persist under repeated acquisition and interference phases. Graph metrics correlate even more strongly with learning degree in this setting, reaching $\rho = 0.512$ (Appendix D), which may indicate that repeated training encourages the reuse or consolidation of previously learned computational structures.

7 Related Work

LLM Knowledge Acquisition. A growing body of work studies learning mechanism behinds LLM knowledge acquisition. One line uses synthetic or fictional corpora, e.g., biographies of fabricated individuals (Allen-Zhu and Li, 2023; Zucchet et al., 2025; Ou et al., 2025; Zhu et al., 2025; Feng et al., 2024), Wikipedia-style entries for fictional entities (Chang et al., 2024), or **post-cutoff data** (Huang et al., 2024), to examine how data properties, training choices, and curricula (e.g., ordering dependencies between facts and implications) affect learning. Another line (Chang et al., 2024; Xu et al., 2024b; Leybzon and Kervadec, 2024; Im and Li, 2024; Qian et al., 2024; Tigges et al., 2024) leverages mechanistic interpretability to analyze LLM learning by tracing how internal representations evolve across training stages. The third line (Ren and Sutherland, 2024; Chen et al., 2023; Jain et al., 2023) identify phase transitions that reveal discrete, objective-dependent shifts in model behavior during training. In contrast, we target **concept-level** knowledge rather than isolated factual triples, enabling the study of higher-order phenomena such as concept learning, forgetting, interference, and synergy, while also opening the door to future work on concept generalization, compositional generalization, and hierarchical interactions (Zaeem and Komeili, 2021).

LLM Concept Probing and Editing. Prior work studies conceptual knowledge in LLMs mainly through: (i) **prompt-based probing** of concept properties and relations (Gu et al., 2023; Liao et al., 2023; Shani et al., 2023; Zheng et al., 2024; Peng et al., 2022); (ii) **definition-name alignment** tests (e.g., reverse-dictionary probes) that assess mapping between descriptions and names (Xu et al., 2024a); and (iii) **compositional binding and consistency** evaluations that test correct attribution of concept knowledge to instances and consistency across hierarchies (He et al., 2023; Sosa et al., 2024; Sahu et al., 2022). Complementing these behavioral probes, mechanistic interpretability aims to localize internal components responsible for concept-related behavior (Aljaafari et al., 2024; Wang et al., 2024b). Unlike these largely static evaluations, we focus on the **dynamics of concept learning**: we connect internal *concept circuits* to acquisition, forgetting, and cross-concept interactions, providing a mechanistic, time-resolved complement to existing probing approaches.

8 Conclusion

In this work, we study how large language models acquire, retain, and forget concepts during continual pretraining by linking output behavior with circuit-level interpretability. We introduce the FICO dataset, define **Concept Circuits** as internal representations of concepts, and use graph metrics to characterize **Circuit Topology**. Our experiments reveal consistent patterns in concept learning, forgetting, interference, and synergy, showing that circuit topology provides meaningful signals of learning dynamics that can guide experience replay to mitigate forgetting. Our findings offer a step toward more concept-aware, interpretable, and robust continual pretraining of LLMs.

Limitations

Our study has several limitations. (1) **Limited Model Scale**. Due to computational constraints, our experiments focus on GPT-2 Large (0.7B), with LLaMA-3.2-1B, LLaMA-3.1-8B, and Qwen3-4B-Base used for ablation; extending the analysis to larger LLMs remains important future work. (2) **Limited Exploration of Actionable Training Strategies**. Our focus is analytical, and we instantiate **Circuit-aware Experience Replay** only as a proof-of-concept, evaluated on GPT-2 Large and AntiLeak. Future work should validate it more broadly and explore other concept-aware strategies such as adaptive effort allocation, interference-aware data scheduling, or curriculum design. (3) **Controlled Operationalization of Concept Learning**. FICO operationalizes concept learning as templated prefix-target prediction over fictional concept-relation triples, with disjoint test templates that isolate relational generalization from surface-form memorization (Section 2). This controlled scope leaves broader conceptual abstraction, hierarchical interactions, and compositional generalization outside our evaluation. (4) **Pre-training Focus**. We study continual pre-training; since post-training is likewise continued training on new data, topology metrics could predict which knowledge is vulnerable and circuit-aware replay could mitigate the "alignment tax" (Ortu et al., 2024; Prakash et al., 2024; Lee et al., 2024), but we leave empirical validation to future work.

Acknowledgments

This research is partially supported by the award No. #2604309 from the Faculty Early Career Development Program (CAREER) and the award No. #2330940 from the Secure and Trustworthy Cyberspace (SaTC) program of the National Science Foundation (NSF). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

Ethical considerations

We carefully follow the ACM Code of Ethics and have not found potential societal impacts or risks so far. To the best of our knowledge, this work has no notable harmful effects and uses, environmental impact, fairness considerations, privacy considerations, security considerations, or other potential risks. We will release the constructed FICO dataset to support reproducibility and future research. Since FICO is derived from ConceptNet, which is distributed under the Creative Commons Attribution-ShareAlike 4.0 license, our released dataset will follow the same license and include appropriate attribution to ConceptNet. Users of the dataset should comply with the corresponding attribution and share-alike terms. The dataset does not contain information that names or uniquely identifies individual people, and we manually check that it does not include offensive content. We have not identified notable societal, fairness, privacy, security, environmental, or other risks beyond those associated with standard use of commonsense knowledge resources. The dataset is in English and focuses on concept-centric commonsense knowledge. It covers broad everyday domains represented in ConceptNet, including objects, animals, locations, properties, affordances, part-whole relations, spatial relations, taxonomic relations, and synonym/antonym relations.

References

Nura Aljaafari, Danilo S Carvalho, and André Freitas. 2024. The mechanics of conceptual interpretation in GPT models: Interpretative insights. *arXiv preprint arXiv:2408.11827*.

Zeyuan Allen-Zhu and Yuanzhi Li. 2023. Physics of language models: Part 3.1, knowledge storage and extraction. *arXiv preprint arXiv:2309.14316*.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46:904–911.

Hoyeon Chang, Jinho Park, Seonghyeon Ye, Sohee Yang, Youngkyung Seo, Du-Seong Chang, and Minjoon Seo. 2024. How do large language models acquire factual knowledge during pretraining? *Advances in neural information processing systems*, 37:60626–60668.

Angelica Chen, Ravid Shwartz-Ziv, Kyunghyun Cho, Matthew L Leavitt, and Naomi Saphra. 2023. Sudden drops in the loss: Syntax acquisition, phase transitions, and simplicity bias in mlms. *arXiv preprint arXiv:2309.07311*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela

Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

J. Feng, S. Russell, and J. Steinhardt. 2024. Extractive structures learned in pretraining enable generalization on finetuned facts. *arXiv preprint arXiv:2412.04614*.

Yuling Gu, Bhavana Dalvi Mishra, and Peter Clark. 2023. Do language models have coherent mental models of everyday things? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1892–1913, Toronto, Canada. Association for Computational Linguistics.

Michael Hanna, Sandro Pezzelle, and Yonatan Belinkov. 2024. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms. *arXiv preprint arXiv:2403.17806*.

Marcel Harpaintner, Natalie M Trumpp, and Markus Kiefer. 2018. The semantic content of abstract concepts: A property listing study of 296 abstract words. *Frontiers in psychology*, 9:1748.

Yuan He, Jiaoyan Chen, Ernesto Jimenez-Ruiz, Hang Dong, and Ian Horrocks. 2023. Language model analysis for ontology subsumption inference. *arXiv preprint arXiv:2302.06761*.

Martin N Hebart, Oliver Contier, Lina Teichmann, Adam H Rockter, Charles Y Zheng, Alexis Kidder, Anna Corriveau, Maryam Vaziri-Pashkam, and Chris I Baker. 2023. Things-data, a multimodal collection of large-scale datasets for investigating object representations in human brain and behavior. *Elife*, 12:e82580.

Stefan Heimersheim and Neel Nanda. 2024. How to use and interpret activation patching. *arXiv preprint arXiv:2404.15255*.

J. Huang, D. Yang, and C. Potts. 2024. Demystifying verbatim memorization in large language models. *arXiv preprint arXiv:2407.17817*.

Ahmed Imtiaz Humayun, Randall Balestriero, and Richard Baraniuk. 2024. Grokking and the geometry of circuit formation. In *ICML 2024 Workshop on Mechanistic Interpretability*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Shawn Im and Yixuan Li. 2024. Understanding the learning dynamics of alignment with human feedback. *arXiv preprint arXiv:2403.18742*.

David Isele and Akansel Cosgun. 2018. Selective experience replay for lifelong learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Samyak Jain, Robert Kirk, Ekdeep Singh Lubana, Robert P Dick, Hidenori Tanaka, Edward Grefenstette, Tim Rocktäschel, and David Scott Krueger. 2023. Mechanistically analyzing the effects of fine-tuning on procedurally defined tasks. *arXiv preprint arXiv:2311.12786*.

Ricky Laishram, Ahmet Erdem Sariyüce, Tina Eliassi-Rad, Ali Pinar, and Sucheta Soundarajan. 2018. Measuring and improving the core resilience of networks. In *Proceedings of the 2018 World Wide Web Conference*, pages 609–618.

Vito Latora and Massimo Marchiori. 2001. Efficient behavior of small-world networks. *Physical review letters*, 87(19):198701.

Andrew Lee, Xiaoyan Bai, Itamar Pres, Martin Wattenberg, Jonathan K Kummerfeld, and Rada Mihalcea. 2024. A mechanistic understanding of alignment algorithms: A case study on dpo and toxicity. *arXiv preprint arXiv:2401.01967*.

Danny D Leybzon and Corentin Kervadec. 2024. Learning, forgetting, remembering: Insights from tracking LLM memorization during training. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 43–57.

Yuxiao Li, Eric J Michaud, David D Baek, Joshua Engels, Xiaoqing Sun, and Max Tegmark. 2025. The geometry of concepts: Sparse autoencoder feature structure. *Entropy*, 27(4):344.

Jiayi Liao, Xu Chen, and Lun Du. 2023. Concept understanding in large language models: An empirical study.

Mark Newman. 2010. *Networks: An Introduction*. Oxford University Press, Inc., USA.

Francesco Ortu, Zhijing Jin, Diego Doimo, Mrinmaya Sachan, Alberto Cazzaniga, and Bernhard Schölkopf. 2024. Competition of mechanisms: Tracing how language models handle facts and counterfactuals. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8420–8436.

Yixin Ou, Yunzhi Yao, Ningyu Zhang, Hui Jin, Jiacheng Sun, Shumin Deng, Zhenguo Li, and Huajun Chen. 2025. How do LLMs acquire new knowledge? a knowledge circuits perspective on continual pre-training. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 19889–19913, Vienna, Austria. Association for Computational Linguistics.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2023. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*.

Hao Peng, Xiaozhi Wang, Shengding Hu, Hailong Jin, Lei Hou, Juanzi Li, Zhiyuan Liu, and Qun Liu. 2022. COPEN: Probing conceptual knowledge in

pre-trained language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5015–5035, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Nikhil Prakash, Tamar Shaham, Tal Haklay, Yonatan Belinkov, and David Bau. 2024. Fine-tuning enhances existing mechanisms: A case study on entity tracking. In *International Conference on Learning Representations*, volume 2024, pages 8057–8082.

Chen Qian, Jie Zhang, Wei Yao, Dongrui Liu, Zhenfei Yin, Yu Qiao, Yong Liu, and Jing Shao. 2024. Towards tracing trustworthiness dynamics: Revisiting pre-training period of large language models. *arXiv preprint arXiv:2402.19465*.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Arnav Raj. Hyperbolic geometry of reasoning: Probing LLM hidden states. In *ICLR 2026 Workshop on Geometry-grounded Representation Learning and Generative Modeling*.

Yi Ren and Danica J Sutherland. 2024. Learning dynamics of LLM finetuning. *arXiv preprint arXiv:2407.10490*.

Pritish Sahu, Michael Cogswell, Yunye Gong, and Ajay Divakaran. 2022. Unpacking large language models with conceptual consistency. *arXiv preprint arXiv:2209.15093*.

Stephen B Seidman. 1983. Network structure and minimum degree. *Social networks*, 5(3):269–287.

Chen Shani, Jilles Vreeken, and Dafna Shahaf. 2023. Towards concept-aware large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13158–13170, Singapore. Association for Computational Linguistics.

Rosario Uceda Sosa, Karthikeyan Natesan Ramamurthy, Maria Chang, and Moninder Singh. 2024. Reasoning about concepts with LLMs: Inconsistencies abound. In *First Conference on Language Modeling*.

Charles Spearman. 1961. The proof and measurement of association between two things.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge.

Curt Tigges, Michael Hanna, Qinan Yu, and Stella Biderman. 2024. Llm circuit analyses are consistent across training and scale. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, pages 40699–40731.

Mengru Wang, Yunzhi Yao, Ziwen Xu, Shuofei Qiao, Shumin Deng, Peng Wang, Xiang Chen, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. 2024a. Knowledge mechanisms in large language models: A survey and perspective. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7097–7135, Miami, Florida, USA. Association for Computational Linguistics.

Xiaohan Wang, Shengyu Mao, Shumin Deng, Yunzhi Yao, Yue Shen, Lei Liang, Jinjie Gu, Huajun Chen, and Ningyu Zhang. 2024b. Editing conceptual knowledge for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 706–724, Miami, Florida, USA. Association for Computational Linguistics.

Xiaobao Wu, Liangming Pan, Yuxi Xie, Ruiwen Zhou, Shuai Zhao, Yubo Ma, Mingzhe Du, Rui Mao, Luu Anh Tuan, and William Yang Wang. 2025. Antileakbench: Preventing data contamination by automatically constructing benchmarks with updated real-world knowledge. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18403–18419.

Ningyu Xu, Qi Zhang, Menghan Zhang, Peng Qian, and Xuanjing Huang. 2024a. On the tip of the tongue: Analyzing conceptual representation in large language models with reverse-dictionary probe. *arXiv preprint arXiv:2402.14404*.

Yang Xu, Yi Wang, Hengguan Huang, and Hao Wang. 2024b. Tracking the feature dynamics in LLM training: A mechanistic study. *arXiv preprint arXiv:2412.17626*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Barry Menglong Yao, Qifan Wang, and Lifu Huang. 2025a. Error-driven data-efficient large multimodal model tuning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 20289–20306, Vienna, Austria. Association for Computational Linguistics.

Yunzhi Yao, Ningyu Zhang, Zekun Xi, Mengru Wang, Ziwen Xu, Shumin Deng, and Huajun Chen. 2025b. Knowledge circuits in pretrained transformers. *Preprint*, arXiv:2405.17969.

Mohammad Nokhbeh Zaeem and Majid Komeili. 2021. Cause and effect: Hierarchical concept-based explanation of neural networks. *arXiv preprint arXiv:2105.07033*.

Junhao Zheng, Shengjie Qiu, and Qianli Ma. 2024. Concept-1k: A novel benchmark for instance incremental learning. *arXiv e-prints*, pages arXiv–2402.

Yufa Zhou, Yixiao Wang, Xunjian Yin, Shuyan Zhou, and Anru R Zhang. 2025. The geometry of reasoning: Flowing logics in representation space. *arXiv preprint arXiv:2510.09782*.

Mingkang Zhu, Xi Chen, Zhongdao Wang, Bei Yu, Hengshuang Zhao, and Jiaya Jia. 2025. Effective LLM knowledge learning via model generalization. *arXiv preprint arXiv:2503.03705*.

Nicolas Zucchet, Jorg Bornschein, Stephanie C.Y. Chan, Andrew Kyle Lampinen, Razvan Pascanu, and Soham De. 2025. How do language models learn facts? dynamics, curricula and hallucinations. In *Second Conference on Language Modeling*.

A Knowledge Type Grouping and Knowledge Filtering

A.1 Knowledge Type Grouping

Our dataset is constructed from the ConceptNet knowledge graph, where each knowledge instance is represented as a triplet $\langle \text{subject}, \text{relation}, \text{object} \rangle$. Since ConceptNet contains a large number of fine-grained relation types, we group them into a smaller set of semantically coherent, high-level knowledge categories to facilitate concept-level analysis. Each relation is assigned to a category according to its primary semantic function.

As shown in Table 5, taxonomic and definitional relations (*IsA*, *DefinedAs*, *FormOf*, *InstanceOf*) are grouped into **Hyponym and Hypernym**. Relations expressing similarity or contrast (*Synonym*, *SimilarTo*, *Antonym*, *DistinctFrom*) are grouped into **Synonym and Antonym**. Part-whole and compositional relations (*PartOf*, *HasA*, *MadeOf*) are grouped into **Meronym and Holonym**. Relations describing properties, affordances, or functional roles (*HasProperty*, *UsedFor*, *CapableOf*, *ReceivesAction*) are grouped into **Property and Affordance**. Spatial relations (*AtLocation*, *LocatedNear*) are grouped into **Spatial Relation**.

A.2 Knowledge Filtering

We retain the above five concept-centric knowledge types in our experiments because they capture relatively stable semantic properties of real-world concepts. In contrast, we filter out the **Causality & Event**, **Lexical/Etymological**, and **Other** categories, since they primarily encode event-level associations, linguistic form, or noisy relations that are less suitable for modeling concept-level knowledge.

Based on these filtered relations, we construct fictional concepts from real concepts. Following previous work (Allen-Zhu and Li, 2023; Zucchet et al., 2025; Ou et al., 2025; Zhu et al., 2025; Feng et al., 2024), for each concept-relation pair, we enforce a single canonical object and remove competing candidates. For example, if *dog CapableOf* is associated with multiple possible objects such as *run* and *guard your home*, we retain only one canonical object. This procedure yields a controlled and constrained concept-relation mapping, ensuring that fictional concepts are not merely renamed versions of existing real-world concepts.

High-level Knowledge Type	Relation Types
Hyponym and Hypernym	*IsA*, *DefinedAs*, *FormOf*, *InstanceOf*
Synonym and Antonym	*Synonym*, *SimilarTo*, *Antonym*, *DistinctFrom*
Meronym and Holonym	*PartOf*, *HasA*, *MadeOf*
Property and Affordance	*HasProperty*, *UsedFor*, *CapableOf*, *ReceivesAction*
Spatial Relation	*AtLocation*, *LocatedNear*
Excluded Relation Categories	
Causality & Event	*Causes*, *MotivatedByGoal*, *HasPrerequisite*, *HasSubevent*, *HasFirstSubevent*, *HasLastSubevent*, *CreatedBy*
Desire	*Desires*, *CausesDesire*
Lexical / Etymological	*DerivedFrom*, *EtymologicallyDerivedFrom*, *EtymologicallyRelatedTo*
Other	*RelatedTo*, *HasContext*, *ExternalURL*, *SymbolOf*

Table 5: Mapping from fine-grained relation types to high-level knowledge categories. We retain five concept-centric categories for experiments and exclude relation types that primarily encode events, lexical form, or noisy contextual associations.

Data	Train	Test
# Concepts	1,000	500
# Knowledges	3,075	1,586
# Samples	92,250	1,586
# Tokens	1,039,784	13,530

Table 6: Dataset statistics of FICO.

B Dataset Statistics

As shown in Table 6, following dataset construction, we leverage 1,000 concepts for training and 500 concepts for testing. The training set contains 3,075 knowledge triples, which are instantiated into 92,250 training samples with different templates, comprising approximately 1.04M tokens in total. We train LLMs on our dataset for 10 epochs, leading to total training tokens of 10.4M, similar as previous continual-pretraining work (Ou et al., 2025). The test set consists of 1,586 knowledge triples and corresponding evaluation samples, totaling 13,530 tokens.

C Circuit Discovery

We instantiate concept circuits using EAP-IG Hanna et al. (2024), an efficient edge-level circuit discovery method that approximates activation patching with integrated gradients. We choose EAP-IG because our analysis requires extracting circuits for many concepts across multiple continual-pretraining checkpoints, making exhaustive causal intervention methods computationally prohibitive. Following prior circuit-discovery work, we view a circuit as a sparse computational subgraph whose standalone computation preserves the model behavior on a target task.

For each concept c_i, we construct a circuit-discovery set from its associated held-out knowledge triples $\{(c_i, r_{ij}, o_{ij})\}_j$. Each clean input is the prefix generated from the subject–relation pair (c_i, r_{ij}), with the gold object o_{ij} as the correct target. To construct a corrupted counterpart, we replace the gold object with an incorrect object sampled from another concept, yielding clean–corrupted pairs for attribution. EAP-IG assigns an importance score to each edge by integrating the gradient of the task loss along the path from corrupted activations to clean activations. The task metric is the logit difference between the correct and corrupted objects:

$$M(x) = \text{logit}(o_{ij} \mid x) - \text{logit}(\tilde{o}_{ij} \mid x),$$

where $\tilde{o}_{ij}$ denotes the corrupted object. Following EAP-IG, we use $L(x) = -M(x)$ as the attribution loss.

After scoring all edges, we rank them by the absolute EAP-IG score and select the smallest set of top-ranked edges whose resulting subgraph preserves at least τ of the full model's performance on the corresponding concept. Performance preservation is measured by the average logit-difference metric $M(x)$ over the concept's triples. In our main experiments, we set $\tau = 70\%$, following prior circuit-level interpretability work (Ou et al., 2025; Yao et al., 2025b) as a practical faithfulness–sparsity trade-off. Lower thresholds may omit important edges, while higher thresholds yield denser circuits that are less suitable for topology analysis. To verify that our conclusions are not tied to this heuristic, we additionally evaluate $\tau = 80\%$ and observe similar circuit–behavior correlations, as shown in Table 1.

D Robustness Under Longer or Repeated Continual Training

To assess whether our findings persist beyond the two-stage continual pretraining setup in the main paper, we conduct an additional **repeated continual training** experiment with GPT-2 Large. Specifically, after obtaining checkpoint π_2 from Stage 2 (Forgetting Induction), as discussed in Section 3.2, we continue training the model for two more stages using the same protocol: we further train π_2 on FICO dataset, followed by BIO dataset, resulting in a **four-stage continual training setup**. Across this extended setting, we observe that the main trends remain qualitatively consistent: concept learning and forgetting continue to show substantial cross-concept variability, maintain non-trivial correlations with circuit graph metrics, and exhibit a positive association between stronger learning and greater forgetting. These results suggest that our findings are robust beyond a single two-stage continual training cycle.

We show results in Figure 7, Figure 8, Figure 9, and Figure 10.

E LLaMA and Qwen Results

Table 7 reports the complete cross-model correlations with p-values corresponding to Table 4 in the main paper.

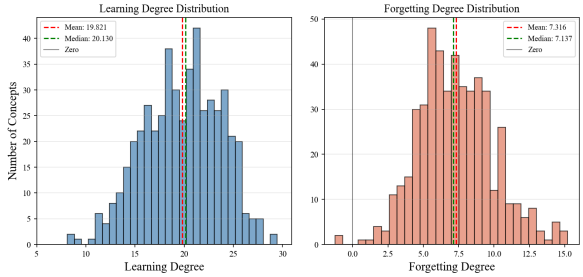


Figure 7: Distribution of learning and forgetting degree across concepts for GPT-2 Large under four-stage continual training.

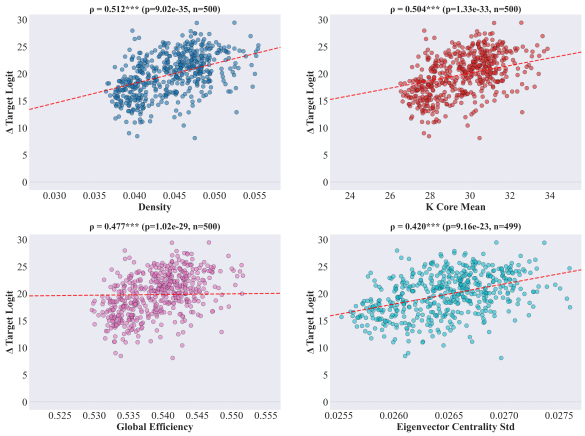


Figure 8: Correlation between learning degree and LLM circuit graph scores.

E.1 LLaMA-3.2-1B Results

E.1.1 RQ1 Results

Figure 11 shows the distribution of concept learning and forgetting degree for LLaMA-3.2-1B, again revealing substantial variability across concepts. Figure 12 and Figure 13 show that circuit topology metrics remain correlated with concept learning and forgetting degree, respectively, reproducing the same qualitative trend observed for GPT-2 Large in the main paper; the strongest metric, density, reaches $\rho \approx 0.40$ for learning and $\rho \approx 0.41$ for forgetting (Table 4). Figure 14 further shows the temporal evolution of circuit topology during continual forgetting, exhibiting a similar pattern of early increase followed by later decrease and stabilization. Finally, Figure 15 shows a positive correlation between learning degree and forgetting degree, confirming that concepts learned more strongly also tend to be forgotten more strongly under subsequent training.

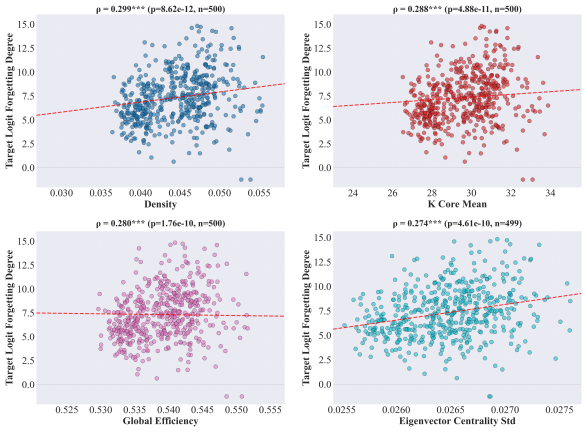


Figure 9: Correlation between forgetting degree and LLM circuit graph scores.

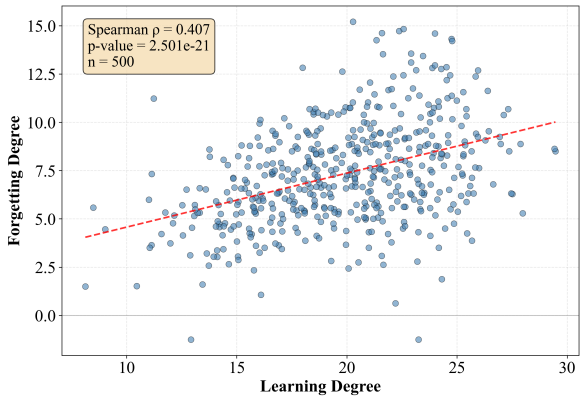


Figure 10: Spearman Correlations between learning and forgetting of concepts

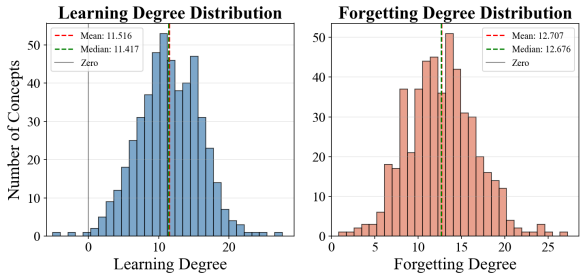


Figure 11: Distribution of learning and forgetting degree across concepts for LLaMA.

Model	Metric	Learning	Learning p	Forgetting	Forgetting p
Qwen3-4B-Base	Density	0.333	2.1e-14	0.416	2.5e-22
Qwen3-4B-Base	k-core	0.319	2.7e-13	0.409	1.4e-21
Qwen3-4B-Base	Eigenvector Centrality	0.328	5.2e-14	0.394	4.8e-20
Qwen3-4B-Base	Global Efficiency	0.285	8e-11	0.304	3.7e-12
LLaMA-3.2-1B	Density	0.399	2.2e-20	0.411	1.1e-21
LLaMA-3.2-1B	k-core	0.375	5.2e-18	0.381	1.5e-18
LLaMA-3.2-1B	Eigenvector Centrality	0.163	2.6e-4	0.175	9.1e-5
LLaMA-3.2-1B	Global Efficiency	0.332	3.4e-14	0.372	9.2e-18
LLaMA-3.1-8B	Density	0.349	1.1e-15	0.386	4.1e-19
LLaMA-3.1-8B	k-core	0.349	1.1e-15	0.389	2.1e-19
LLaMA-3.1-8B	Eigenvector Centrality	0.148	9.4e-4	0.182	4.4e-5
LLaMA-3.1-8B	Global Efficiency	0.201	6.4e-6	0.207	3.5e-6

Table 7: Complete correlations and p-values for Table 4: Spearman correlation between learning/forgetting degree and circuit topology metrics on LLaMA-3.2-1B, LLaMA-3.1-8B, and Qwen3-4B-Base.

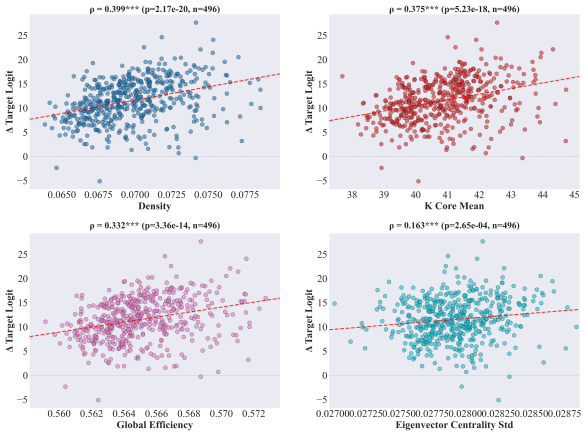


Figure 12: Correlation between learning degree and LLM circuit graph scores.

E.1.2 RQ2 Results

Figure 16 shows concept interference under joint training for LLaMA-3.2-1B, where semantically related concepts induce stronger interference than weakly related concepts. Figure 17 further shows that transferability across knowledge types remains highly heterogeneous and directional, indicating that concept synergy and interference patterns generalize beyond GPT-2 Large.

E.2 LLaMA-3.1-8B Results

E.2.1 RQ1 Results

Figure 18 shows the distribution of concept learning and forgetting degree for LLaMA-3.1-8B, again indicating substantial cross-concept variability. Figure 19 and Figure 20 show that circuit topology metrics remain meaningfully associated with concept learning and forgetting degree, respectively, reproducing the same overall trend observed in smaller models and in the GPT-2 Large results; the strongest metrics, density and k-core, reach $\rho \approx 0.35$ for learning and $\rho \approx 0.39$ for forgetting (Table 4).

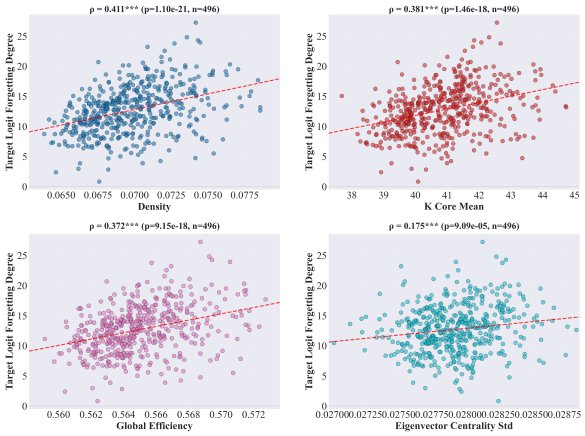


Figure 13: Correlation between forgetting degree and LLM circuit graph scores.

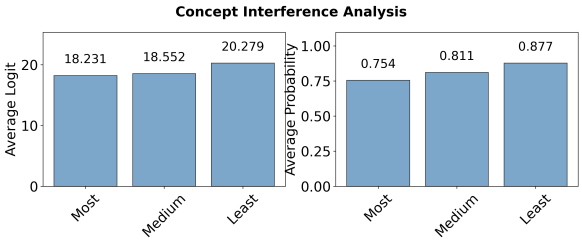


Figure 16: Concept-level interference under joint training.

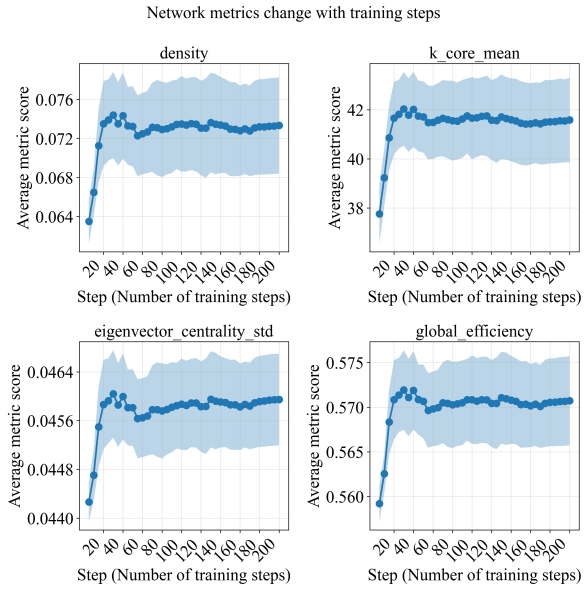


Figure 14: Graph Metric over training steps

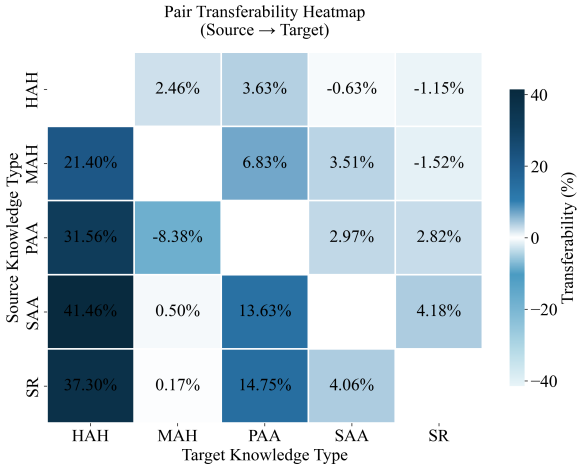


Figure 17: Pairwise dependency between knowledge types

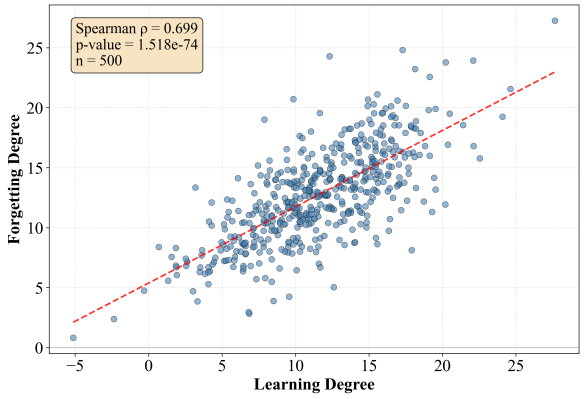


Figure 15: Correlations between learning and forgetting of concepts

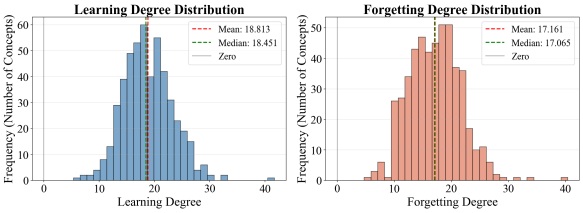


Figure 18: Distribution of learning and forgetting degree across concepts for LLaMA.

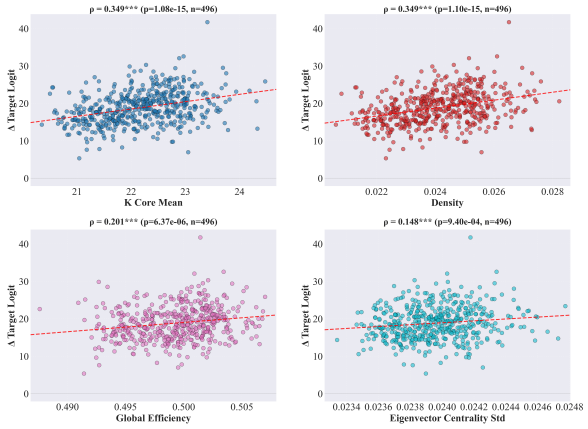


Figure 19: Correlation between learning degree and LLM circuit graph scores.

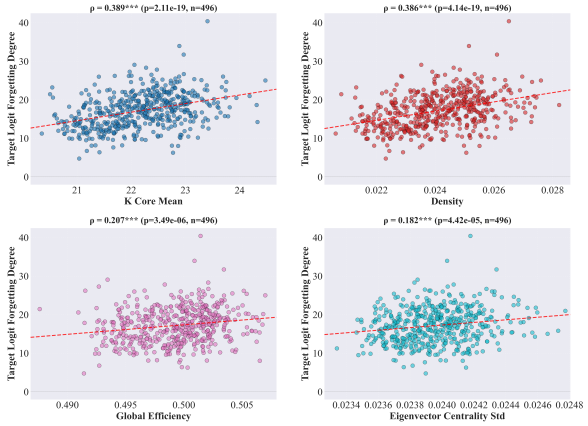


Figure 20: Correlation between forgetting degree and LLM circuit graph scores.

E.2.2 RQ2 Results

Since LLaMA-3.1-8B is a more powerful model, the original interference setting saturates: with $K = 100$ co-training groups, the most, medium, and least similar co-training groups reach 85.1%, 85.8%, and 86.7% average target probability (average target logits of 23.55, 23.94, and 24.59), respectively, a *ceiling effect* in which the task is too easy for a model of this capacity to separate the conditions, not evidence that interference disappears at scale. We therefore leverage a sharpened top-20 setting for LLaMA-3.1-8B: each of the same target concepts is co-trained with its top-20 / middle-20 / bottom-20 most-similar concepts, upsampled so that the training-set size is unchanged (with the same batch size as the original experiment). The final results are shown in Figure 21, recovering the clear interference ordering. To make the mechanism explicit, we further trace full acquisition curves, evaluating each target's held-out samples throughout training (Figure 22): the gap emerges

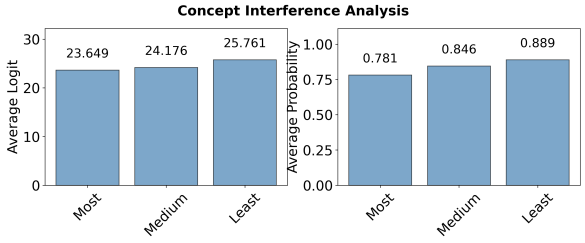


Figure 21: Concept-level interference under joint training on LLaMA-3.1-8B with sharpened top-20 co-training groups (final-step evaluation).

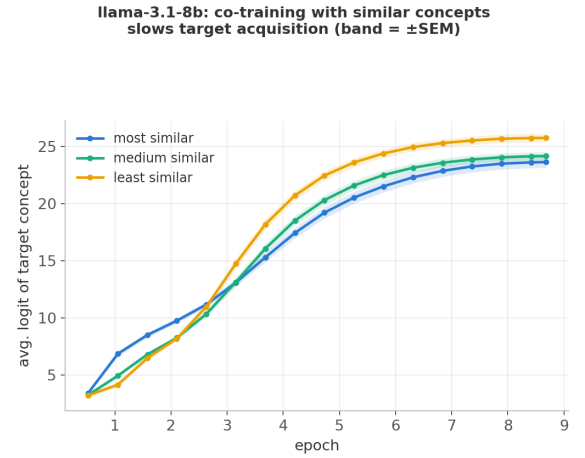


Figure 22: Acquisition curves (average logit of the target concept) for LLaMA-3.1-8B under the three co-training conditions (band = ±SEM). The x-axis shows training epochs over the co-training corpus. The interference gap emerges early in acquisition, peaks mid-training, and narrows gradually as all conditions approach the performance ceiling.

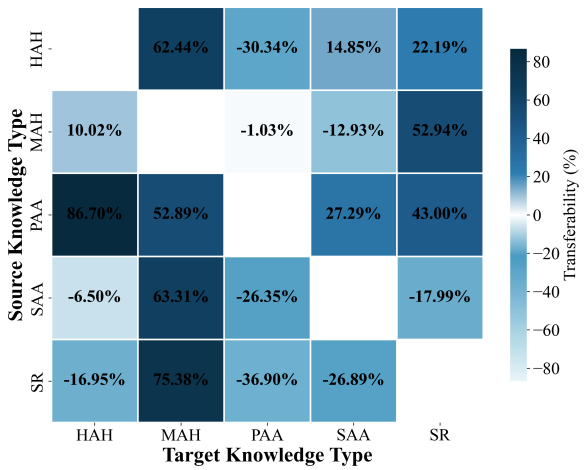


Figure 23: Pairwise dependency between knowledge types

early in acquisition and peaks mid-training, then narrows gradually as all conditions approach the ceiling. At the 8B scale, interference among semantically similar concepts thus manifests as *slower acquisition* rather than significantly lower final attainment. Figure 23 further shows that transferability across knowledge types remains asymmetric and uneven, suggesting that concept-level synergy and interference patterns are robust across model scale.

E.3 Qwen3-4B-Base Results

E.3.1 RQ1 Results

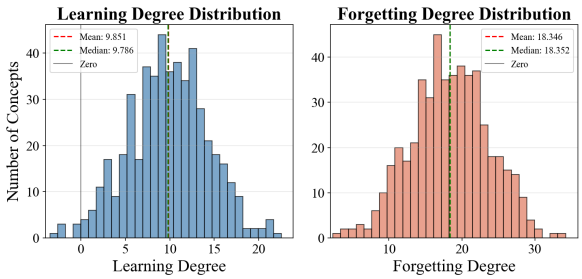


Figure 24: Distribution of learning and forgetting degree across concepts for Qwen3-4B-Base.

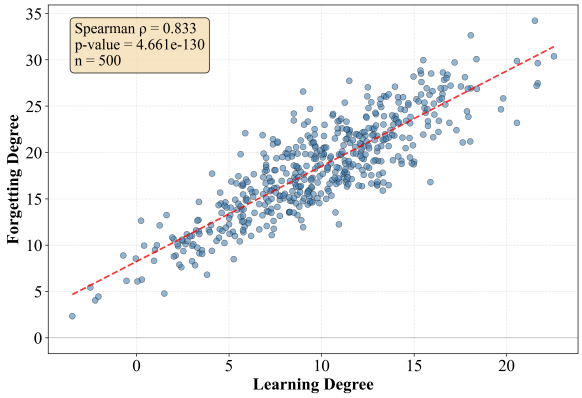


Figure 25: Correlations between learning and forgetting of concepts for Qwen3-4B-Base.

To further validate our findings on modern architectures, we replicate the full RQ1–RQ2 protocol on Qwen3-4B-Base, a recent LLM from a third model family. Figure 24 shows the distribution of concept learning and forgetting degree for Qwen3-4B-Base, again revealing substantial variability across concepts. As shown in Table 4, circuit topology metrics remain correlated with both concept learning and forgetting degree, reproducing the same qualitative trend observed for GPT-2 Large and LLaMA. Finally, Figure 25 shows a positive correlation between learning degree and forgetting degree, confirming that concepts learned more strongly also tend to be forgotten more strongly under subsequent training.

E.3.2 RQ2 Results

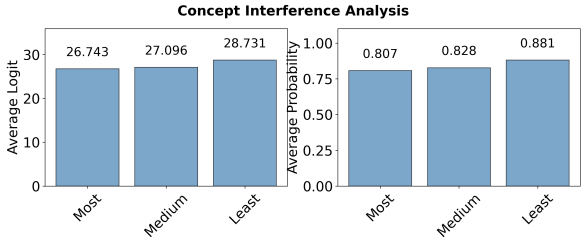


Figure 26: Concept-level interference under joint training on Qwen3-4B-Base.

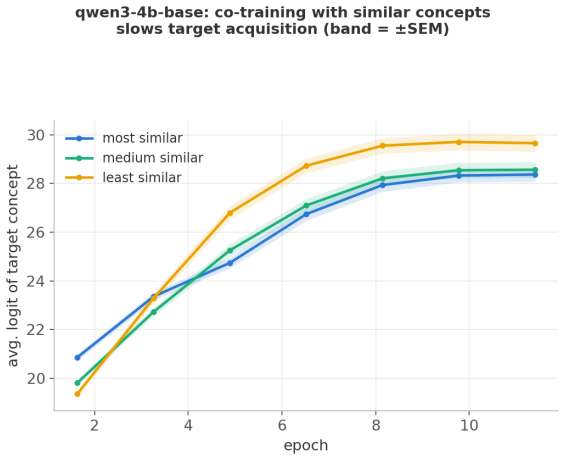


Figure 27: Acquisition curves (average logit of the target concept) for Qwen3-4B-Base under the three co-training conditions (band = ±SEM). The x-axis shows training epochs over the co-training corpus. The interference gap emerges early in acquisition, peaks mid-training, and narrows as all conditions approach the performance ceiling.

We further run the concept-interference experiment on Qwen3-4B-Base with the same grouping as the main experiments and trace full acquisition curves (Figure 27). Figure 26 shows the three co-training conditions: training with weakly related concepts achieves the highest performance (average target probability 0.881, vs. 0.828 for moderately and 0.807 for highly related groups), reproducing the interference ordering observed on the other models. Figure 28 further shows that transferability across knowledge types remains asymmetric and uneven on Qwen3-4B-Base, indicating that concept-level synergy and interference patterns are robust across model families.

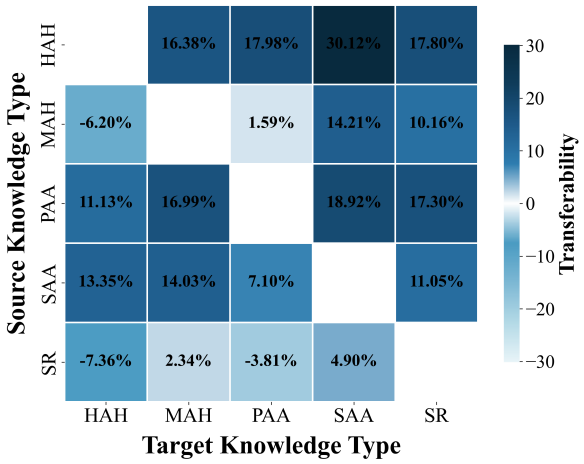


Figure 28: Pairwise dependency between knowledge types on Qwen3-4B-Base.

F Experiment Details

We conduct experiments on 4×40GB A40 GPUs. We use learning rate as 5×10^{-5} and batch size as 128. In our experiments for RQ1, the training data is automatically reshuffled before each training epoch. Since each experiment runs for multiple epochs, this ensures that our analysis is independent of the order in which the concepts are presented to the model, addressing any sensitivity to concept presentation order.

G GPT-2 Large Results for Log Probability

To verify that our findings are not specific to logit-based evaluation, we repeat the RQ1 analyses using **log probability** (Heimersheim and Nanda, 2024; Hanna et al., 2024) as the behavioral metric.

Figure 29 shows the distribution of concept learning and forgetting degree under log probability. Similar to the logit-based results in the main paper, we observe substantial variability across concepts, indicating that both acquisition and forgetting remain highly heterogeneous.

Figure 30a and Figure 30b show that circuit topology metrics remain positively correlated with concept learning and forgetting degree, respectively. In particular, metrics such as eigenvector centrality and k-core continue to exhibit consistent associations, suggesting that the relationship between circuit structure and learning/forgetting dynamics is robust to the choice of behavioral metric.

Finally, Figure 31 shows that concepts with larger learning gains also tend to exhibit greater forgetting under subsequent training, confirming the learn-forget trade-off observed in the main paper. Overall, these results demonstrate that our key findings remain consistent when measured using log probability.

Type	Metric	Learning	Learning p	Forgetting	Forgetting p
Circuit Topology (70%)	Density	0.4474	6.16e-26	0.4585	2.64e-27
Circuit Topology (70%)	k-core	0.4468	7.45e-26	0.4587	2.46e-27
Circuit Topology (70%)	Eigenvector Centrality	0.4403	4.40e-25	0.4459	9.60e-26
Circuit Topology (70%)	Global Efficiency	0.4085	1.73e-21	0.4322	3.95e-24
Circuit Topology (80%)	Density	0.4723	4.34e-29	0.4785	6.54e-30
Circuit Topology (80%)	k-core	0.4475	6.04e-26	0.4610	1.25e-27
Circuit Topology (80%)	Eigenvector Centrality	0.3235	1.27e-13	0.3314	2.95e-14
Circuit Topology (80%)	Global Efficiency	0.4662	2.71e-28	0.4747	2.09e-29
Activation	Mean EAP-IG	0.2680	1.18e-09	0.2784	2.47e-10
Activation	STD EAP-IG	0.2398	5.87e-08	0.2494	1.64e-08
Activation	Max EAP-IG	0.1575	4.13e-04	0.1342	2.67e-03
Random Topology	Density	-0.0078	0.86	-0.0255	0.57
Random Topology	k-core	-0.0458	0.31	-0.0437	0.33
Random Topology	Eigenvector Centrality	0.0473	0.29	0.0737	0.10
Random Topology	Global Efficiency	-0.0559	0.21	-0.0769	0.09

Table 8: Correlation between learning/forgetting degree and circuit metrics (Spearman ρ); p is the corresponding p-value testing the null hypothesis of zero correlation.

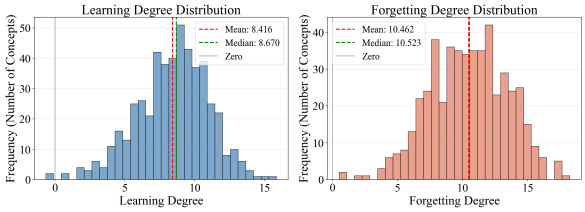


Figure 29: Distribution of learning and forgetting degree across concepts.

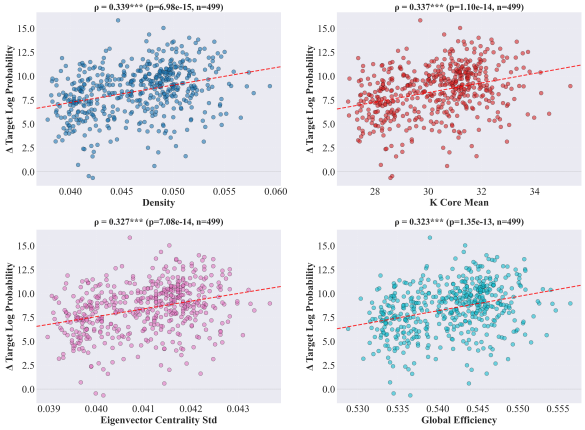
H Complete Results

Table 8 shows complete correlation results with p-values, where p is the corresponding p-value testing the null hypothesis of zero correlation.

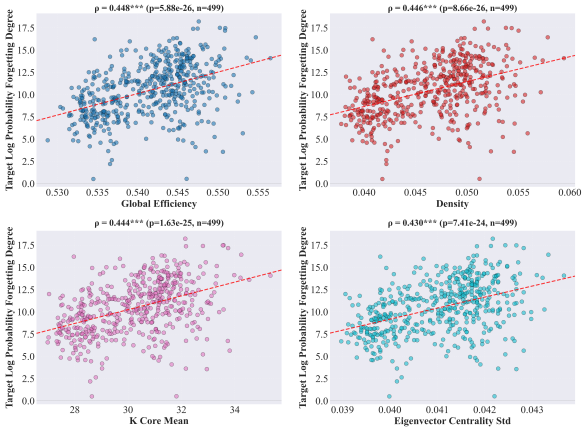
I Baselines Against Circuit Topology

To further evaluate the indicative power of circuit graph topology, we compare our method against two baselines: (1) **Random graph topology**, where we randomly sample the same number of edges to form a graph and compute its topology metrics; and (2) **edge-level EAP-IG statistics** (Hanna et al., 2024), including the mean, maximum, and standard deviation of EAP-IG scores. EAP-IG is a first-order approximation of each edge's indirect effect and serves as a proxy for both gradient norm and activation magnitude at the circuit level.

As shown in Table 1, circuit graph metrics exhibit strong correlations with concept learning and forgetting degree (Spearman $\rho \approx 0.41$–0.47 and 0.43–0.46, respectively, at the 70% threshold).



(a) Correlation between learning degree and LLM circuit pattern.



(b) Correlation between forgetting degree and LLM circuit pattern.

Figure 30: Correlation between learning dynamics and LLM circuit pattern.

J Correlation between Learning and Forgetting

Concepts with larger learning degree tend to exhibit larger forgetting degree during subsequent training, suggesting a trade-off between acquisition strength and subsequent stability.

Correlation between Concept Learning and Forgetting. Figure 32 shows a positive correlation between learning degree and forgetting degree under continual training, indicating that concepts learned more strongly also tend to be forgotten more strongly. Together with the circuit correlation results in Section 2, this pattern is consistent with a trade-off between acquisition strength and subsequent stability.

To shed light on the nature of the learning–forgetting relationship, we further analyze the 20% of concepts with the highest learning degrees. Figure 33 and Figure 34 show the distributions of their

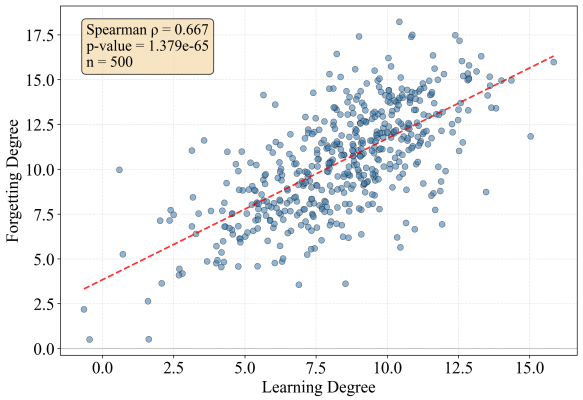


Figure 31: Spearman Correlations between learning and forgetting of concepts

final learning logit and forgetting degree. Two observations are notable. First, while these high-learning-degree concepts do tend to reach elevated final logit values (mean = 21.27), their final logits span a wide range (18–24), indicating that high learning degree does not simply equate to a uniformly high logit ceiling. Second, and more importantly, these concepts also exhibit substantially higher forgetting degrees (mean = 20.95) but span a wide range (14–26). Together, these distributions suggest that the correlation between learning degree and forgetting degree reflects a genuine property of how concepts are encoded during training.

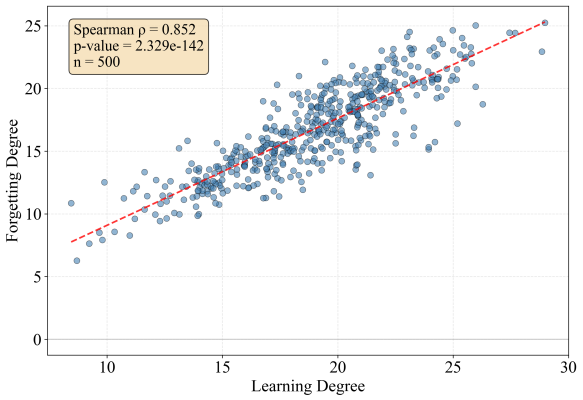


Figure 32: Spearman Correlations between learning and forgetting of concepts

K Evolution of All Circuit Topology Metrics

Figure 35 and Figure 36 show the temporal evolution of all four circuit topology metrics (density, mean k-core, eigenvector centrality, and global efficiency) during the learning and forgetting stages, complementing the two representative metrics in

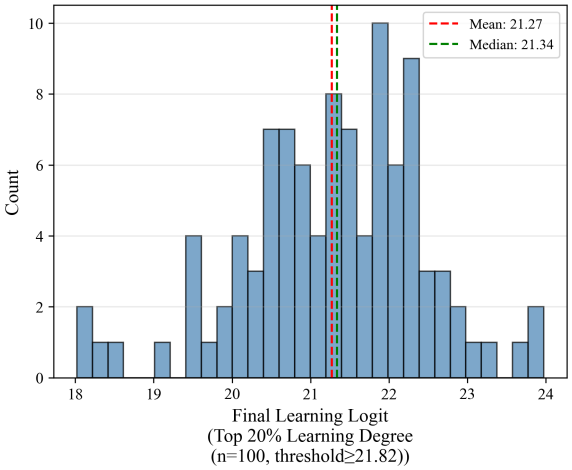


Figure 33: Distribution of final learning logit.

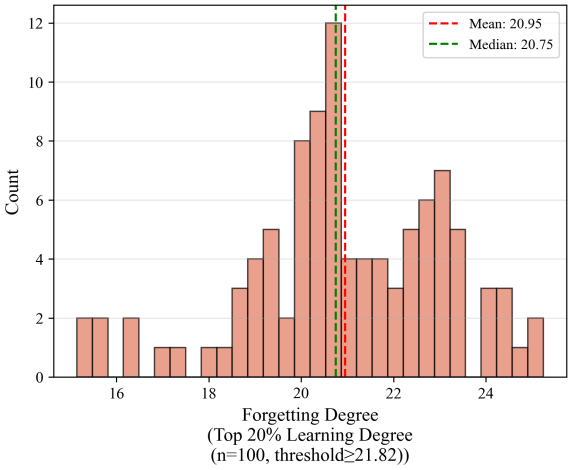


Figure 34: Distribution of forgetting degree

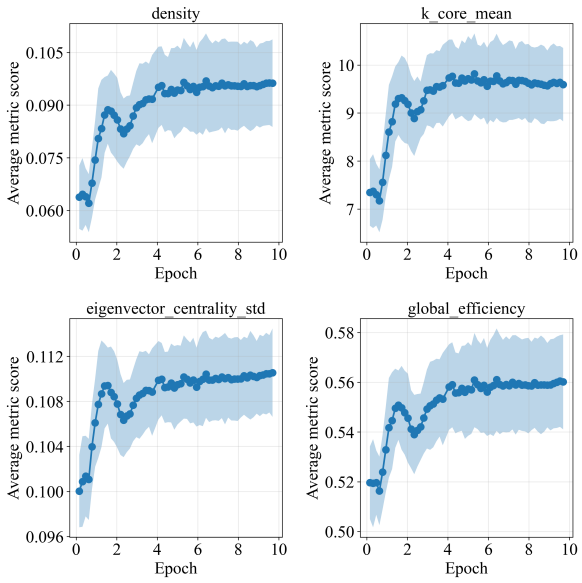


Figure 35: Evolution of all four circuit graph metrics during the learning stage (mean over concepts). The x-axis shows training epochs; the full learning stage spans 10 epochs.

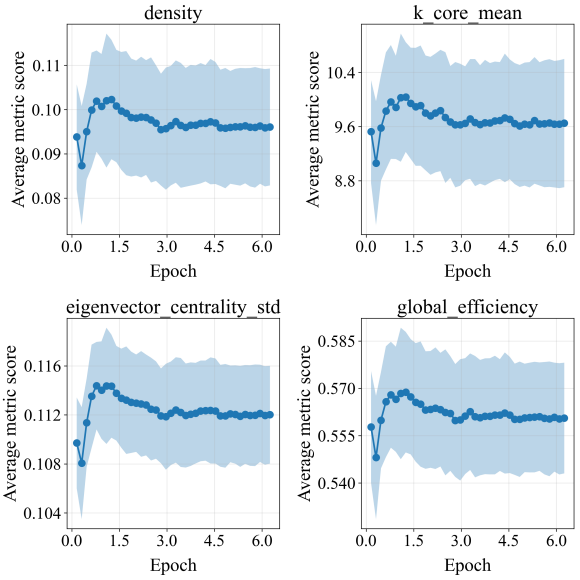


Figure 36: Evolution of all four circuit graph metrics during the forgetting stage (mean over concepts).

the main paper. All four metrics follow the same qualitative pattern within each stage.

L Controlled Interference Experiment: Relatedness vs. Surface Overlap

In the interference experiment of Section 4.1, highly related co-training groups may also share surface forms (object strings, relation types) with the target, so the observed gap could in principle reflect surface-level overlap rather than representational competition. We quantify the covariates of the original design and find that relation-type composition is essentially constant across conditions (mean relation-histogram Jaccard 0.178–0.185, $p = 0.20$), while object-string overlap (19.3% coverage for the highly related group vs. 0.3% for the weakly related group) do differ.

We therefore run a controlled experiment that removes these channels. For each of the same 100 target concepts, we construct three co-training conditions of exactly $K = 75$ auxiliary concepts: high relatedness with natural overlap retained (*hi-ovl*), high relatedness with auxiliary object strings *and* object content-word tokens fully disjoint from the target's (*hi-noovl*), and low relatedness with the same disjointness (*lo-noovl*). Auxiliary texts are subsampled and upsampled so that all three conditions have identical relation histograms, while preserving the relatedness contrast. Training follows the original protocol (GPT-2 Large).

Contrast (avg. prob.)	Δ	p
lo-noovl − hi-noovl	$+11.6$ pp	6.1×10^{-7}
hi-noovl − hi-ovl	-2.2 pp	0.015
lo-noovl − hi-ovl	$+9.4$ pp	6.5×10^{-6}

Table 9: Controlled interference contrasts. The first row isolates semantic relatedness at exactly zero lexical overlap; the second isolates the effect of shared object strings; the third is the protocol-matched total.

Table 9 shows the paired contrasts. The key comparison is the first row, *lo-noovl* vs. *hi-noovl*: both conditions share zero object strings with the target, so they differ only in semantic relatedness. Even under this condition, targets co-trained with highly related but no-object overlap concepts, *hi-noovl*, reach an average target probability 11.6 pp lower than targets co-trained with weakly related concepts, *lo-noovl*, (logit contrast $+0.81$, $p = 7.6 \times 10^{-6}$; 79% of targets show the effect). This pure-relatedness effect accounts for roughly 60% of the $+18.5$ pp least-vs-most gap in the original, uncontrolled design, indicating that the majority of the observed interference is attributable to semantic relatedness itself rather than to surface-level overlap. Removing shared objects slightly *hurts* performance (-2.2 pp), indicating that object overlap by itself slightly helps learning rather than causing the interference. Overall, the interference in Finding 4 is genuinely representational: it survives the removal of all object-string overlap.

M Computational Overhead of Circuit Scoring

Scoring method	Single GPU	10-way parallel
Gradient scoring (per-sample gradient norm)	11.1 min	–
Circuit extraction (EAP-IG)	83.4 min	≈8 min

Table 10: One-time scoring cost for replay-concept selection, measured on GPT-2 Large with all 7,140 AntiLeak concepts.

We measure one-time scoring costs on GPT-2 Large with all 7,140 AntiLeak concepts (Table 10). Gradient scoring (one backward pass per sample on a single GPU) takes 11.1 minutes. EAP-IG circuit extraction requires only a fixed number of forward/backward passes per concept and takes 83.4 minutes on a single GPU; since extraction is fully parallelizable across concepts, this reduces to ≈8.3 minutes wall-clock with 10-way data parallelism in our experiments. Notably, our replay strategy does not require extraction at every checkpoint: circuits are extracted only once, at the final checkpoint of the learning stage. The scoring cost is therefore a fixed one-time overhead, while the retention gains it enables persist for all subsequent deployment, yielding a favorable cost-benefit trade-off.

Quintile (dispersion)	n	Answers lost (%)
Q1	173	67.6
Q2	172	61.9
Q3	172	58.4
Q4	172	53.2
Q5	173	50.9

Table 11: Downstream answer loss by quintile of pre-forgetting eigenvector-centrality dispersion (an answer is lost if it is correct before but incorrect after the forgetting stage).

N Metrics for AntiLeak Experiments

We report three evaluation metrics on the AntiLeak test set: (1) **Logit**, the average logit assigned to the target object; (2) **Accuracy**, the proportion of examples for which the top-1 predicted token matches the ground-truth object; and (3) **Hit@5**, the proportion of examples for which the ground-truth object appears in the top-5 predicted tokens.

O Downstream MCQA Evaluation on AntiLeak

To verify that the forgetting tracked by our logit-based metrics translates into downstream task degradation, we build a closed-book multiple-choice QA evaluation (Yao et al., 2025a) from AntiLeakBench's official multi-choice questions (Wu et al., 2025): 994 single-hop questions whose gold answers match facts in our AntiLeak training set. Since GPT-2 Large is a base model, we use the standard likelihood-based protocol: each of the four options is scored by its log-likelihood as a continuation of the question, and the model's answer is the highest-scoring option.

MCQA accuracy is 92.2% after concept learning and drops to 40.0% after the forgetting stage. Circuit topology computed at the *pre-forgetting* checkpoint predicts this degradation at the level of individual concepts: taking the concepts the model answered correctly before the forgetting stage, ranking them by the eigenvector-centrality score of their pre-forgetting circuits, and splitting them into five equal-sized quintiles, the fraction of answers lost (answered correctly before but incorrectly after the

forgetting stage) decreases monotonically across the quintiles, from 67.6% to 50.9%, as shown in Table 11.

P Use Of AI Assistants

We used AI assistants to help with writing and editing the manuscript. The assistants were used to refine wording, improve clarity, and enhance overall presentation.