Emergence and Localisation of Semantic Role Circuits in LLMs

Anonymous ACL submission

Abstract

Despite displaying semantic competence, large language models' internal mechanisms that ground abstract semantic structure remain insufficiently characterised. To investigate whether and how LLMs develop causally functional representations of semantic roles, we introduce a causal-temporal methodology combining contrastive minimal pairs, edge-attribution circuit discovery, and training-time tracking. Our analysis reveals that LLMs encode semantic roles through highly localised circuits (89–92% attribution within $\leq$28 nodes) that emerge gradually via structural refinement rather than phase transitions. These circuits exhibit moderate cross-scale conservation (24–51% component overlap) alongside high spectral similarity, with larger models reusing similar components while rewiring connections. These findings suggest that LLMs form compact, causally isolated mechanisms for abstract semantic structure that exhibit partial transfer across scales and architectures.

1 Introduction

Do LLMs develop abstract, causally functional representations of semantic structure? Large language models (LLMs) exhibit localised circuits for factual recall (Goldowsky-Dill et al., 2023a; Meng et al., 2022), arithmetic (Conmy et al., 2023; Stolfo et al., 2023), and logical reasoning (Kim et al., 2025). However, it remains unclear whether such mechanisms extend to the *abstract relational semantic structure* that underlies natural language understanding. Current mechanistic studies often focus on specialised algorithmic behaviours in trained models (e.g., induction heads, copying, factual associations (Meng et al., 2022; Kissane et al., 2024)), leaving two central gaps in our understanding of semantic representations in LLMs.

First, abstract semantic structure. Semantic roles (e.g., AGENT, THEME, INSTRUMENT) constitute a prevalent descriptive component of predicate–argument structure that generalises across surface forms and syntactic realisations in natural language (Fillmore, 1976). Formally, a semantic role r associates a predicate p and argument position i with a thematic relation: for instance, in *"The children played in the garden"*, the LOCATION role links *garden* to the playing event whether expressed as *"in the garden"*, *"at the garden"*, or *"outside in the garden"* (see App. A for a complete formalisation). Mechanistically, such predicate–argument binding requires integrating information across tokens and abstracting from surface cues, unlike behaviours explained by positional heuristics or lexical templates. Whether LLMs implement this binding through *causally functional circuits* however remains an open question.

Second, temporal emergence. Most circuit analyses examine only final checkpoints, obscuring *when* semantic mechanisms arise, stabilise, and become computationally indispensable. Given that several behaviours emerge through sharp transitions (e.g., in-context learning, algorithmic generalisation (Wei et al., 2022; Nanda et al., 2023; He et al., 2024)), it is unclear whether semantic-role circuits likewise appear abruptly or gradually consolidate. Understanding this timeline is important both for training-time interventions (Aljaafari et al., 2025b; Cheng et al., 2024) and for explaining how syntactic and semantic abstractions co-develop. To address these questions, we introduce **COMPASS** (Compositional Predicate–Argument Semantic Structure; Fig.1), a causal–temporal methodology that combines edge-attribution patching (Hanna et al., 2024) with training-time circuit tracking[1]. It investigates predicate–argument binding by isolating role-specific computation through contrastive prompts that differ only in their role-indicating scaffold (e.g., "delivered the package *to*

[1]Supporting code and datasets are available at a public repository < anonymised url >

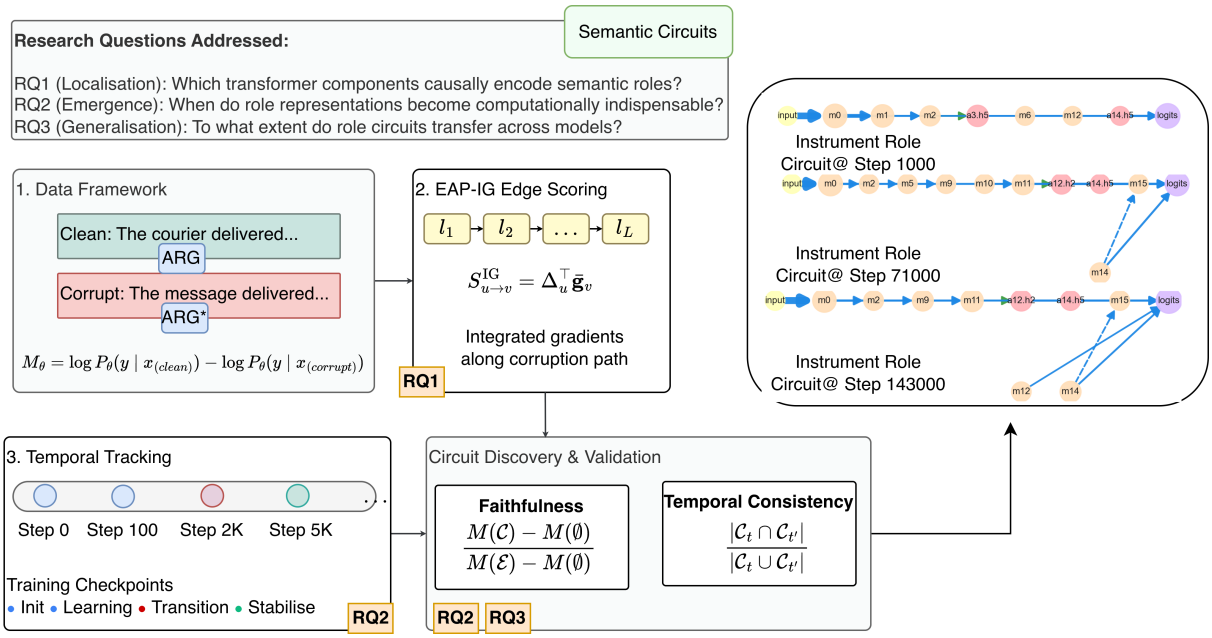

Figure 1: **COMPASS methodology.** It extracts and tracks the circuits that mediate semantic-role behaviour in LLMs, revealing where role-specific computation occurs and how it develops over training. (1) Role-cross minimal pairs isolate predicate–argument binding. (2) EAP-IG identifies edges whose interventions affect role predictions, producing sparse, causally functional subgraphs. (3) Temporal analysis follows these subgraphs across checkpoints to determine when their structure stabilises and when they become computationally indispensable.

the office" vs. "*with the* truck"), what we term *role-cross minimal pairs*, then tracks the causal pathways responsible for correct predictions. This approach disentangles the formation of circuit structure from its functional engagement and enables the evaluation of its transferability. We apply COMPASS across model scales and families to answer:

- **RQ1 (Localisation):** Which model components causally encode semantic roles?
- **RQ2 (Emergence):** When do these circuits become computationally indispensable?
- **RQ3 (Generalisation):** To what extent do role circuits transfer across model scales and architectures?

We find that role-binding circuits localise to compact sets of attention heads and MLPs, with the results indicating that predicate–argument binding emerges through *gradual refinement* rather than sudden reorganisation. Circuit presence does not guarantee immediate functional use: models appear to allocate semantic capacity early and exploit it only later. Larger models reuse similar components while wiring them differently. By demonstrating causally determined, functional circuits for shallow semantic structure, our method provides mechanistic evidence that LLMs acquire structured semantic representations and offers new avenues for targeted circuit editing and training-time intervention.

Contributions. We summarise our contributions as: (i) introduce COMPASS, a causal–temporal method for discovering and tracking semantic-role circuits in LLMs; (ii) show that semantic-role information concentrates in small sets of components whose structural organisation stabilises before becoming functionally indispensable; and (iii) demonstrate partial cross-scale and cross-architecture transfer of these circuits.

2 Related Work

Mechanistic Interpretability (MI) and Circuit Discovery. MI seeks to reverse-engineer neural network computations (Bereska and Gavves, 2024). Foundational work analysed how attention heads, MLPs, and residual streams implement algorithms (Elhage et al., 2021), leading to discoveries such as induction heads (Olsson et al., 2022) and task-specific circuits (Wang et al., 2023; Conmy et al., 2023). Causal intervention methods form the core of modern circuit analysis: activation patching tests causal effects by swapping activations (Meng et al., 2022), while gradient-based variants such as Attribution Patching (AtP) (Nanda, 2023; Syed et al., 2024) improve scalability but suffer from gradient saturation. AtP* (Kramár et al., 2024) intro-

duces architectural fixes; Edge Attribution Patching (EAP) (Hanna et al., 2024) attributes causal influence to individual edges, with EAP-IG mitigating saturation via Integrated Gradients. Parallel work on feature disentanglement uses Sparse Autoencoders (Templeton et al., 2024; Bricken et al., 2023) and transcoders (Dunefsky et al., 2024), but they are surrogate-based approaches that introduce reconstruction errors and do not guarantee causal necessity (Gao et al., 2025; Kantamneni et al., 2025).

Training Dynamics and Compositional Emergence. Transformers often acquire capabilities through sharp transitions, including grokking-style shifts (Power et al., 2021; Nanda et al., 2023; Aljaafari et al., 2025b) and phase changes in in-context learning (Wei et al., 2022). Recent work shows linguistic structure develops progressively: subspaces associated with syntax and semantics become more coherent over training (Müller-Eberstein et al., 2023), and task-specific circuits emerge in coordinated phases (Tigges et al., 2024). However, most analyses rely on probing or scalar metrics, leaving open whether transitions reflect emergence of causally functional circuits. To our knowledge, no prior work combines causal circuit localisation with systematic temporal tracking of compositional semantic representations across training and scale.

Semantic Understanding in Neural Language Models. While transformers achieve strong Semantic Role Labelling (Chen et al., 2025), most work examines accuracy rather than internal mechanisms. Probing studies suggest hierarchical linguistic knowledge organisation (Tenney et al., 2019; Hewitt and Manning, 2019), but probing reveals only linear separability, not causal involvement (Caucheteux et al., 2021; Conia and Navigli, 2022). Circuit-level interpretability has mapped concrete behaviours such as IOI (Wang et al., 2023) but has not addressed whether transformers implement causally functional circuits for abstract predicate–argument relations, motivating our focus on semantic-role mechanisms.

Method Positioning. Probing scales efficiently but lacks causal grounding; intervention methods such as path patching (Goldowsky-Dill et al., 2023b) and causal scrubbing (Chan et al., 2022) provide strong guarantees but are computationally prohibitive across many checkpoints. Surrogate-based approaches introduce reconstruction artefacts hindering temporal comparisons. We employ EAP-IG (Hanna et al., 2024), which provides path-specific causal attribution directly on the original model, extending it to a temporal setting to analyse when semantic-role circuits form, stabilise, and become functionally engaged across training and scale (detailed comparison of methods in App. G).

3 Methodology

3.1 Semantic Role Circuits

Computational graph representation. Following Hanna et al. (2024), we view transformer computation as a directed acyclic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where nodes $u \in \mathcal{V}$ correspond to module outputs $u = (\text{type}, \ell, h, i)$ with type $\in \{\text{AttnHead}, \text{MLP}\}$, layer ℓ, head h (or $h = \varnothing$ for MLPs), and position i. Each u outputs an activation $\mathbf{z}_u \in \mathbb{R}^{d_{\text{model}}}$ to the residual stream, and edges $(u \to v) \in \mathcal{E}$ denote residual connections.

Task: Role-conditioned continuation. To investigate predicate–argument binding, we adopt a role-conditioned continuation task grounded in frame semantics (Fillmore, 1976) and PropBank (Palmer et al., 2005a). We vary the *role-indicating scaffold* while keeping predicate and argument fillers fixed. If models represent abstract semantic roles, this behaviour should be mediated by **localised role circuits** $\mathcal{C}^{(r)}$. We construct **role-cross minimal pairs** that differ only in the scaffold:

"The courier delivered the package <u>to the</u>"
→ GOAL: "office"
"The courier delivered the package <u>with the</u>"
→ INSTRUMENT: "truck"

Pairs use single-token fillers and enforce **token parity** ($|\text{toks}(x^{(r)})| = |\text{toks}(x^{(s)})|$). We retain only items where the model predicts correct continuations in both variants, ensuring circuits reflect functionally active behaviour (detailed in App. B).

Evaluation. Task performance uses next-token accuracy: $\text{CNP}_{Acc} = \mathbb{E}_{(x^{(r)}, y^{(r)})}\left[\mathbb{1}\left[y^{(r)} = \arg\max_v P_\theta(v \mid x^{(r)})\right]\right]$. Circuit attribution uses the negative log-probability of the role-appropriate target: $\mathcal{L}_{\text{CNP}}(x^{(r)}, y^{(r)}) = -\log P_\theta(y^{(r)} \mid x^{(r)})$, and EAP-IG attributes this loss to edges to identify components supporting role–appropriate predictions.

3.2 COMPASS: Causal-Temporal Circuit Discovery

COMPASS integrates causal localisation with temporal tracking to recover role circuits $\mathcal{C}^{(r)}$ that exhibit (i) causal necessity, (ii) structural sparsity ($|\mathcal{C}^{(r)}| \ll |\mathcal{E}|$), and (iii) temporal stability. It identifies emergence times $(\hat{t}_c, t_{\text{cons}})$ characterising when circuits become functional. The procedure has three phases (full details of method and metrics are in App. E and D).

Phase 1: Causal localisation via EAP-IG. For each role pair $(x^{(r)}, x^{(s)})$, EAP-IG (Hanna et al., 2024) computes edge scores approximating their causal contribution to $\mathcal{L}_{\text{CNP}}$: $S^{\text{IG}}_{u\to v} = \Delta_u^\top \bar{\mathbf{g}}_v$, where $\Delta_u = \mathbf{z}_u^{(r)} - \mathbf{z}_u^{(s)}$, and $\bar{\mathbf{g}}_v$ averages gradients along an IG interpolation path with $\alpha_k = k/m$ (we use m=5). We normalise by total mass: $\tilde{S}^{\text{IG}}_{u\to v} = S^{\text{IG}}_{u\to v} / \sum_{e\in\mathcal{E}} |S^{\text{IG}}_e|$ and extract the top-K=200 edges by $|\tilde{S}^{\text{IG}}_{u\to v}|$ at each checkpoint t to define $\mathcal{C}_t^{(r)}$. Node importance is induced from incident edge mass: $\text{Importance}_t^{(r)}(\ell, h) = \sum_{\substack{(u\to v)\in\mathcal{C}_t^{(r)} \\ v\in(\ell,h)}} |\tilde{S}^{\text{IG}}_{u\to v}|$.

Phase 2: Temporal monitoring. At each checkpoint t, we compute causal and structural signals, including metrics from Mueller et al. (2025). *Faithfulness* measures circuit indispensability: $\text{Faithfulness}_t(\mathcal{C}) = (M_t(\mathcal{C}) - M_t(\emptyset))/(M_t(\mathcal{E}) - M_t(\emptyset))$, where $M_t(\mathcal{C})$ is CNP_{acc} when only edges in $\mathcal{C}$ are active, $M_t(\mathcal{E})$ is full-model performance, and $M_t(\emptyset)$ is null baseline. *Circuit persistence* is measured via Jaccard similarity: $\text{Stability}_t(\mathcal{C}) = |\mathcal{C}_t \cap \mathcal{C}_{t+\delta}|/|\mathcal{C}_t \cup \mathcal{C}_{t+\delta}|$, along with Top-$K$ node mass and Gini coefficient.

Phase 3: Emergence detection. We define two temporal markers. *Consolidation time* t_{cons} marks when the top-K node set stabilises: $t_{\text{cons}} = \min\{t : \text{Stability}_t(\mathcal{C}) \geq 0.6 \text{ for } \geq 2 \text{ steps}\}$.

Functional transition $\hat{t}_c$ is the change-point in faithfulness, estimated by the best-fitting two-segment linear model via least-squares regression: $\hat{t}_c = \arg\max_t R^2\left(\text{PiecewiseLinear}(t)\right)$,

with 95% confidence intervals via bootstrap resampling (n=1,000; details in App. D.4).

3.3 Cross-Model Similarity

We assess circuit transferability across models by comparing structure and function via graph overlap and spectral geometry.

Structural overlap. Using the top-K=30 nodes and edges ranked by $|\tilde{S}|$, we compute Jaccard overlap between models i, j for each role: $J(V_i, V_j) = |V_i \cap V_j|/|V_i \cup V_j|$, $J(E_i, E_j) = |E_i \cap E_j|/|E_i \cup E_j|$.

Spectral similarity. To compare higher-order circuit geometry, we compute root-mean-square deviation between the smallest k=16 eigenvalues of the edge-weighted Laplacian using top-K=50 edges: $d_{\text{spec}}(i,j) = \sqrt{\frac{1}{k}\sum_{m=1}^{k}(\lambda_m^{(i)} - \lambda_m^{(j)})^2}$.

Smaller d_{spec} indicates similar information-flow geometry despite differing edge sets (full details in App. D.4).

4 Experimental Setup

Models. We select models to satisfy three criteria: (i) dense training checkpoints to trace circuit formation, (ii) variation in scale to assess transferability, and (iii) architectural diversity to distinguish general strategies from implementation-specific artefacts. PYTHIA (14M, 410M, 1B) (Biderman et al., 2023) provides comprehensive checkpoints, enabling fine-grained analysis of circuit emergence. To test whether these circuits persist beyond a single family, we include LLAMA-1B (Touvron et al., 2023), which differs in tokenisation, training data, and architectural choices. This pairing separates training-time dynamics (within family) from architectural robustness (across families).

Datasets and Experimental Software. We instantiate the role-conditioned continuation task (Sec. 3.1) across eight semantic roles spanning participant (BENEFICIARY, INSTRUMENT), directional (GOAL, SOURCE, PATH), locative and temporal (LOCATION, TIME), and propositional (TOPIC) categories, providing broad coverage of predicate–argument structure types (Fillmore, 1976; Palmer et al., 2005b; Carnie, 2021). After filtering to retain only examples where models correctly predict role-appropriate targets in both contexts, we obtain approximately 6,000–8,000 pairs per model across roles (exact counts appear in Table 5). Full dataset construction details, role categorisations, filtering procedures, and statistics are provided in App. B. Software and computation specifications appear in App. C.2.

5 Results

We address three research questions on semantic-role circuits across eight roles using PYTHIA (14M,

410M, 1B) and LLAMA-1B. For presentation clarity, we report main text results for four representative roles, BENEFICIARY, INSTRUMENT, LOCATION, and TIME; spanning participant, locative, and temporal categories. Results for the remaining roles (GOAL, SOURCE, PATH, TOPIC) follow similar qualitative trends and are provided in App. F.3. Unless otherwise specified, results refer to PYTHIA-1B. Comparisons across model scales and architectures are explicitly noted. Full results and metric definitions are given in App. F and E.

5.1 Localisation of Role Circuits (RQ1)

We find highly localised circuits for all tested roles, with the top 20 nodes capturing 89–92% of attribution mass at convergence (Tab. 1). Despite this shared sparsity, roles exhibit qualitatively distinct architectures and developmental trajectories.

Circuit architectures vary by role. Causal-flow analysis (Fig. 2) reveals systematic cross-role differences. BENEFICIARY develops the most complex architecture, combining early attention, extensive MLP branching, and rich late-layer connectivity with multiple value-composition operations. In contrast, TIME shifts from early-layer attention to convergent mid-to-late processing; INSTRUMENT stabilises to a balanced hybrid circuit; and LOCATION transitions from mid-training expansion to more distributed integration. Low to mid Jaccard overlap across roles (Fig. 4) confirms these correspond to distinct, role-specific subgraphs. Full circuit evolution for additional roles in App. F.4.

Role	T-5	T-10	T-20	k for 80/90/95%	Gini (mass)
BENEFICIARY	0.395	0.611	**0.906**	16 / 20 / 25	**0.439**
INSTRUMENT	0.492	0.688	**0.917**	14 / 19 / 23	**0.489**
LOCATION	0.483	0.673	**0.897**	15 / 21 / 25	**0.519**
TIME	0.466	0.664	**0.910**	15 / 20 / 24	**0.475**

Table 1: **Final-step concentration profiles of role circuits (step 143K).** For each role, we report cumulative attribution mass captured by the top (T-k) nodes (T-5, T-10, T-20), the smallest k achieving 80/90/95% total mass, and the Gini coefficient over attribution mass. Together, these metrics show that role circuits are highly sparse, with a small set of nodes dominating computation, while differences in Gini reflect variation in how evenly mass is distributed beyond the top contributors.

Roles exhibit distinct developmental trajectories. Temporal analysis (Fig. 3) shows heterogeneous emergence dynamics. INSTRUMENT starts with high faithfulness, undergoes a sharp decline, then stabilises with edge density dipping before

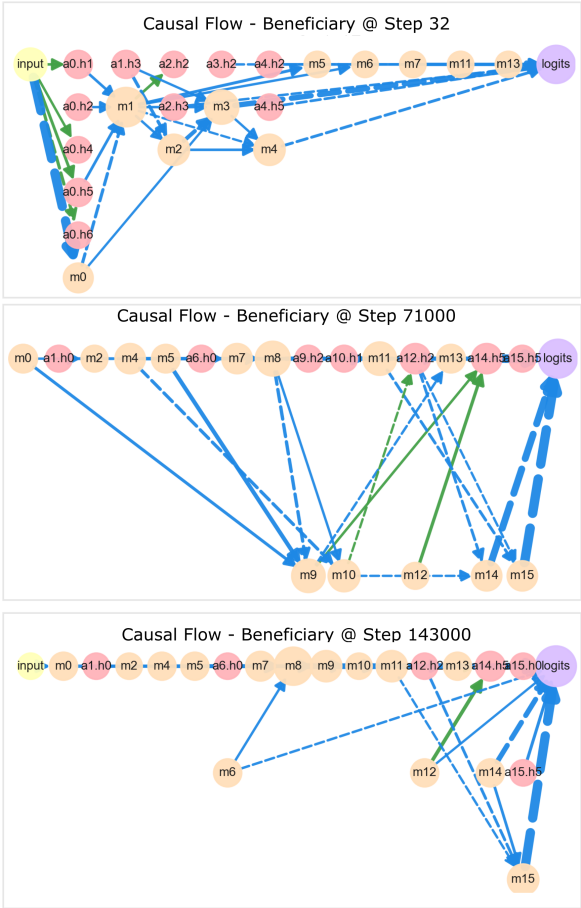

Figure 2: **Evolution of the BENEFICIARY circuit across training.** The circuit reorganises from early exploration (step 32), through mid-training feature extraction (step 71K), to its final architecture (step 143K). Edge colour encodes operation type (blue: residual; green: value composition; dashed: negative), and edge width indicates attribution magnitude.

rising, suggesting late consolidation into denser pathways. BENEFICIARY exhibits the most pronounced volatility: faithfulness spikes sharply at step 50k, drops sharply, partially recovers; circuit size decreases early but re-expands in the final 70K steps, yielding a distinctive "expand–contract" pattern. LOCATION and TIME maintain relatively stable faithfulness after initial drops, with edge density showing moderate fluctuations, consistent with gradual strengthening of connectivity. Circuit sizes contract from initialisation (29–54 nodes) to convergence (31–36 nodes) across all roles, with BENEFICIARY showing the largest initial size and most rapid contraction.

Key findings. (i) Highly concentrated circuits with role-specific structure. All roles converge to compact circuits (89–92% mass in $\leq$20 nodes) with role-specific architectures confirmed by low cross-

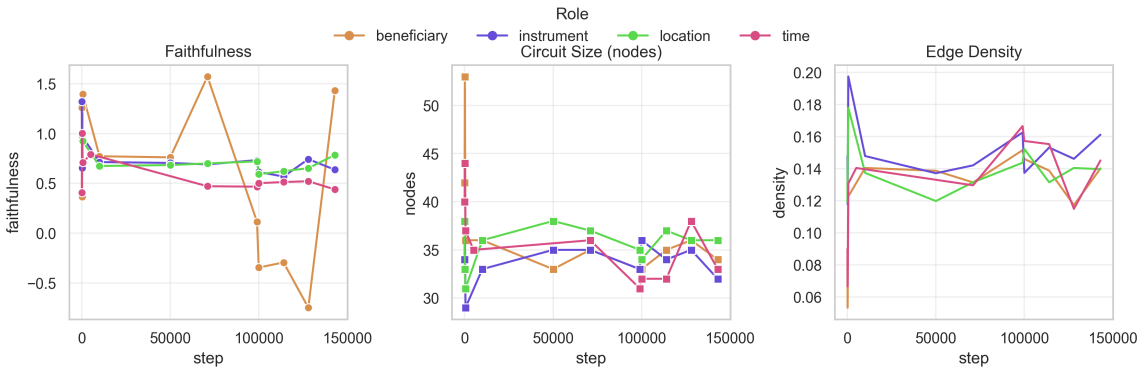

Figure 3: **Structural and functional dynamics of role circuits across training.** Faithfulness (left) shows pronounced role-dependent volatility. In contrast, structural metrics evolve smoothly: circuit size (middle) contracts gradually, and edge density (right) rises or stabilises, showing structure consolidates early and steadily while functional engagement remains variable.

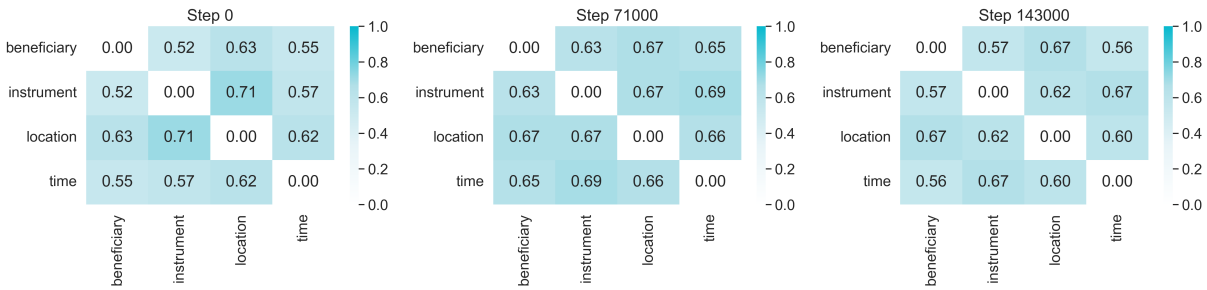

Figure 4: **Cross-role overlap of high-importance components over training (PYTHIA–1B).** Overlap remains consistently low across training stages, indicating roles recruit largely distinct component sets, supporting circuit differentiation rather than shared mechanisms.

role overlap. **(ii) Continuous refinement, not discrete transitions.** Structural metrics (circuit size, edge density) evolve smoothly throughout training, indicating continuous refinement rather than discrete transitions. **(iii) Structure–function dissociation.** Faithfulness exhibits pronounced non-monotonicity even as circuits structurally consolidate. This demonstrates that circuits can be structurally well-formed yet functionally but temporarily underutilised. **(iv) Role-specific stabilisation timelines.** Roles show distinct functional trajectories: BENEFICIARY exhibits extreme volatility, INSTRUMENT sharp early decline then stability, LOCATION and TIME moderate stability after initial adjustment.

5.2 Emergence Dynamics (RQ2)

We find that role-binding circuits emerge via *continuous refinement* rather than discrete phase transitions. Roles become causally indispensable early (0–512 steps), but structural consolidation unfolds over tens of thousands of steps, producing pronounced *structure–function dissociation*.

Role	t_{ind} (steps)	t_{cons} (steps)
BENEFICIARY	–	50,000
INSTRUMENT	32	128
LOCATION	128	128
TIME	512	5,000

Table 2: **Emergence timings for role circuits.** t_{ind} marks earliest step where ablation consistently harms performance; t_{cons} marks when structure stabilises. All roles consolidate within ∼2k steps, but indispensability varies widely.

Early indispensability with heterogeneous emergence. Roles become indispensable at different stages (Tab. 2): INSTRUMENT by step 32, LOCATION by step 128, and TIME by step 512. BENEFICIARY exhibits fluctuating improvement in faithfulness and never exceeds our conservative indispensability threshold ($\mu + \sigma$), leaving t_{ind} undefined and indicating usefulness without provable necessity under this criterion. This dissociation likely reflects differences in cue salience, frequency, and computational demands. Architectural heterogeneity across roles is detailed in App. F.4.

Role	$\hat{t}_c$ (steps)	95% CI
BENEFICIARY	10,000	[128, 10,000]
INSTRUMENT	50,000	[128, 10,000]
LOCATION	10,000	[128, 10,000]
TIME	5,000	[128, 114,000]

Table 3: **Change-point estimates for Top-K node mass** via piecewise linear regression. Extremely wide confidence intervals indicate gradual, continuous evolution without discrete phase transitions.

Model pair	Node J.	Edge J.	d_{spec}
PYTHIA–14M ↔ PYTHIA–410M	0.24	0.11	0.12
PYTHIA–14M ↔ PYTHIA–1B	0.29	0.12	0.10
PYTHIA–410M ↔ PYTHIA–1B	0.42	0.17	0.01
PYTHIA–1B ↔ LLAMA–1B	0.51	0.14	0.02

Table 4: **Cross-scale and -family similarity of role circuits.** Node-level overlap increases with model scale while edge-level overlap remains low. Cross-family comparison shows the highest node overlap. Values report median Top-K Jaccard (K=30) and spectral distance. Lower d_{spec} indicates higher functional similarity.

Continuous sparsification without phase transitions. Piecewise linear regression on Top-K mass yields extremely wide change-point confidence intervals (CIs; Tab. 3), ruling out discrete transitions. This observation contrasts with grokking, where change-points are sharp, localised and abrupt (Power et al., 2021; Nanda et al., 2023). Instead, structural metrics mostly evolve smoothly throughout training (Fig. 3), consistent with gradual sparsification, with some per role specification like LOCATION exhibiting mild "expand–contract" patterns. Final concentration is extreme: Top-20 nodes capture 89.7–91.7% of mass, with 95% coverage in just 23–25 nodes (Tab. 1).

Structure–function dissociation. Although structural metrics evolve smoothly, faithfulness trajectories exhibit pronounced non-monotonicity (Fig. 3, left). Structural consolidation precedes functional indispensability by several steps (Tab. 2: t_{cons}=128 vs. t_{ind}=32–512), demonstrating that *circuit presence does not guarantee circuit engagement*.

Robustness to scaffold variation and frequency effects. To test whether circuits reflect abstract role binding rather than memorised lexical scaffolds, we evaluate within-role paraphrases for LOCATION and INSTRUMENT. Faithfulness and sparsity remain largely unchanged under paraphrase (App. F.5). Moreover, circuit properties at convergence show weak correlations with (i) per-role filtered sample size and (ii) scaffold frequency in the pretraining corpus (App. F.6), suggesting the observed architectural differences are not driven by dataset volume or training-signal strength alone.

Key findings. (i) Continuous refinement throughout training. Structural metrics evolve smoothly, with wide change-point CIs indicating no discrete phase shifts. **(ii) Early indispensability, prolonged consolidation.** Roles become indispensable within 0–512 steps, while structural refinement continues for over 143K steps as attribution mass redistributes. **(iii) Structure–function dissociation and delayed engagement.** Circuits stabilise structurally many steps before becoming functionally indispensable; faithfulness shows crashes and recoveries even as sparsity increases, indicating that *circuit presence does not mean engagement*. **(iv) Role-specific developmental trajectories.** Despite shared qualitative patterns, roles exhibit distinct functional dynamics reflecting role-specific computational demands.

5.3 Cross-Scale and Cross-Family Generalisation (RQ3)

We find moderate structural conservation (24–51% node overlap) across models. They converge on shared *functional vocabularies* while implementing divergent *routing patterns*: component reuse exceeds connection reuse by ∼2:1. Spectral analysis reveals functional alignment despite topological divergence, with small eigenvalue distances (< 0.02) coexisting with 76–88% edge mismatch.

Cross-scale correspondence. Node-level overlap increases with scale proximity (Tab. 4): 14M↔410M yields J_V=0.24, 14M↔1B reaches J_V=0.29, and 410M↔1B achieves J_V=0.42, the strongest within-family match. Edge-level overlap remains low (J_E≈0.11–0.17), indicating models reuse similar high-importance nodes but wire them differently. Spectral distances decrease with scale (d_{spec}=0.12→0.10→0.01), suggesting progressive geometric refinement: larger models realise similar information-flow patterns with increasingly aligned connectivity, reflecting shared training conditions (e.g., corpus, optimiser) and architectural continuity within families.

Cross-family correspondence. The PYTHIA-1B↔LLAMA-1B comparison exhibits the *highest node overlap* (J_V=0.51), exceeding even the

closest within-Pythia pair (410M↔1B: J_V=0.42). This suggests architectural constraints at the 1B scale bias both models toward similar component sets despite different pretraining corpora and architectural choices. However, edge-level overlap remains modest (J_E=0.14) and spectral distance (d_{spec}=0.02) slightly lower than the best within-family match. This "shared components, divergent wiring" pattern indicates models converge on a common functional vocabulary while implementing distinct routing schemas shaped by architectural and training differences.

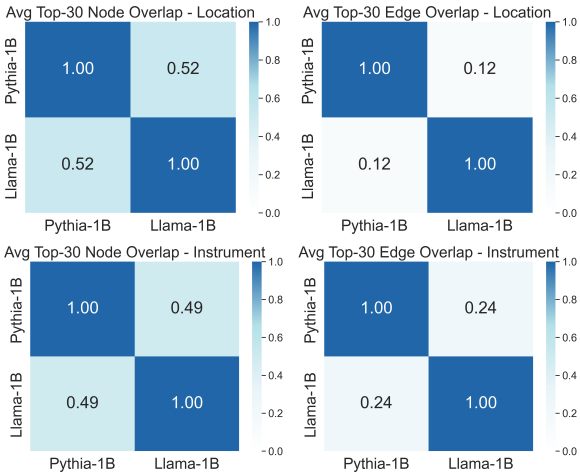

Figure 5: **Cross-family correspondence for Location (top) and Instrument (bottom).** Node sets align substantially more than edges between PYTHIA–1B and LLAMA–1B, suggesting shared component selection but model-specific routing.

Role-specific patterns. Per-role analysis (Fig. 5) reveals heterogeneous cross-family transferability. LOCATION shows higher node overlap (52%) than INSTRUMENT (49%) between Pythia-1B and LLaMA-1B, but lower edge overlap (12% vs 24%), suggesting LOCATION's distributed architecture reuses components while reorganising connections more substantially. INSTRUMENT maintains stronger edge preservation (24%), consistent with more stable routing patterns. Across both roles, node-level overlap consistently exceeds edge-level overlap by ∼2–4×, reinforcing that component reuse dominates connection reuse even across architectural families. Cross-family overlap often exceeds these cross-scale values within the Pythia family for nearby scale pairs, with full per-role results and scale-specific patterns in App. F.

Key findings. (i) Moderate structural conservation across scales and families. High-importance nodes show 24–51% overlap; cross-family correspondence (51%) exceeds strongest within-family pairing (42%), indicating circuits reflect both task demands and model-specific factors. **(ii) Component reuse exceeds connection reuse.** Models converge on shared functional vocabularies (which components matter) while diverging in routing structure. **(iii) Spectral alignment despite topological divergence.** Small spectral distances coexist with large edge differences, showing models realise similar information–flow geometry via distinct connectivity. **(iv) Scale-dependent refinement within families.** Spectral distance decreases monotonically across PYTHIA scales; cross-family increase reflects architectural differences.

6 Conclusion

Our results characterise semantic-role circuits along three dimensions. *RQ1* shows that role information localises into compact, role-specific subgraphs. *RQ2* indicates gradual emergence: functional importance can appear early while structural properties evolve, with role-dependent faithfulness. *RQ3* reveals partial cross-scale conservation, with components reused across models but connected through different routing schemes. Mechanistically, the absence of sharp transitions suggests that circuits are shaped by sustained optimisation rather than being "switched on" once competence appears. From a linguistic standpoint, these circuits approximate role–filler bindings, central to predicate–argument semantics (*who did what to whom, when, with what*), implying partially modular causal mechanisms rather than purely diffuse heuristics. COMPASS extends MI temporally (when circuits appear and engage), cross-scale (how they persist across sizes/families), and semantically (predicate–argument relations beyond lexical cues). The presence of compact, causally functional circuits and their partial transfer across models suggests these mechanisms reflect underlying task structure rather than mere memorising surface co-occurrences. For safety and alignment, localising such mechanisms may enable targeted interventions on specific subgraphs, complementing training-time approaches that promote compositionality (Aljaafari et al., 2025a). Heterogeneous emergence timelines further motivate investigating curricula or early-stopping strategies that prioritise stabilising particular semantic capabilities.

Limitations

This study focuses on English and a subset of roles; whether similar circuits arise in typologically diverse languages or richer role inventories remains unknown. Wide change-point confidence intervals reflect smooth structural trajectories, but more sensitive emergence metrics may yield finer resolution. We analyse decoder-only architectures; encoder–decoder models may exhibit different patterns. Future work should extend to multilingual and multimodal models, study interactions between semantic-role circuits and syntax or coreference mechanisms, and test whether analogous patterns appear for other semantic abstractions such as quantification, modality, or negation.

Ethical considerations

This work aims to improve the interpretability of semantic processing in LLMs through circuit discovery methods. While understanding internal mechanisms benefits safety research and model development, we acknowledge potential risks in localising computational pathways that could be exploited for targeted interventions. Our analysis focuses on controlled semantic tasks with comprehensive validation across training and scales to ensure responsible development and transparent reporting of findings.

References

Nura Aljaafari, Danilo Carvalho, and Andre Freitas. 2025a. CARMA: Enhanced compositionality in LLMs via advanced regularisation and mutual information alignment. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 16250–16270, Suzhou, China. Association for Computational Linguistics.

Nura Aljaafari, Danilo S Carvalho, and André Freitas. 2025b. Trace for tracking the emergence of semantic representations in transformers. *arXiv preprint arXiv:2505.17998*.

Emmanuel Ameisen, Jack Lindsey, Adam Pearce, Wes Gurnee, Nicholas L. Turner, Brian Chen, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, Jonathan Marcus, Michael Sklar, Adly Templeton, Trenton Bricken, Callum McDougall, Hoagy Cunningham, Thomas Henighan, Adam Jermyn, Andy Jones, and 8 others. 2025. Circuit tracing: Revealing computational graphs in language models. *Transformer Circuits Thread*.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 86–90, Montreal, Quebec, Canada. Association for Computational Linguistics.

Leonard Bereska and Stratis Gavves. 2024. Mechanistic interpretability for AI safety - a review. *Transactions on Machine Learning Research*.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, and 1 others. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, and 6 others. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*.

Andrew Carnie. 2021. *Syntax: A generative introduction*. John Wiley & Sons.

Charlotte Caucheteux, Alexandre Gramfort, and Jean-Remi King. 2021. Disentangling syntax and semantics in the brain with deep networks. In *International conference on machine learning*, pages 1336–1348. PMLR.

Lawrence Chan, Adrià Garriga-Alonso, Nicholas Goldwosky-Dill, Ryan Greenblatt, Jenny Nitishinskaya, Ansh Radhakrishnan, Buck Shlegeris, and Nate Thomas. 2022. Causal scrubbing, a method for rigorously testing interpretability hypotheses. *AI Alignment Forum*.

Huiyao Chen, Meishan Zhang, Jing Li, Min Zhang, Lilja Øvrelid, Jan Hajič, and Hao Fei. 2025. Semantic role labeling: A systematical survey. *Preprint*, arXiv:2502.08660.

Ning Cheng, Zhaohui Yan, Ziming Wang, Zhijie Li, Jiaming Yu, Zilong Zheng, Kewei Tu, Jinan Xu, and Wenjuan Han. 2024. Potential and limitations of llms in capturing structured semantics: A case study on srl. In *Advanced Intelligent Computing Technology and Applications*, pages 50–61, Singapore. Springer Nature Singapore.

Simone Conia and Roberto Navigli. 2022. Probing for predicate argument structures in pretrained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4622–4632, Dublin, Ireland. Association for Computational Linguistics.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso.

2023. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352.

Jacob Dunefsky, Philippe Chlenski, and Neel Nanda. 2024. Transcoders find interpretable LLM feature circuits. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, and 6 others. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2021/framework/index.html.

Charles J Fillmore. 1976. Frame semantics and the nature of language. *Annals of the New York Academy of Sciences*, 280(1):20–32.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The Pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2025. Scaling and evaluating sparse autoencoders. In *The Thirteenth International Conference on Learning Representations*.

Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Jun Koba Sato, and Aryaman Arora. 2023a. Localizing model behavior with path patching. *CoRR*, abs/2304.05969.

Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Jun Koba Sato, and Aryaman Arora. 2023b. Localizing model behavior with path patching. *ArXiv*, abs/2304.05969.

Michael Hanna, Sandro Pezzelle, and Yonatan Belinkov. 2024. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms. In *ICML 2024 Workshop on Mechanistic Interpretability*.

Tianyu He, Darshil Doshi, Aritra Das, and Andrey Gromov. 2024. Learning to grok: Emergence of in-context learning and skill composition in modular arithmetic tasks. *Advances in Neural Information Processing Systems*, 37:13244–13273.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *North American Chapter of the Association for Computational Linguistics*.

Daniel Jurafsky and James H. Martin. 2025. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*, 3rd edition. Online manuscript released August 24, 2025.

Subhash Kantamneni, Joshua Engels, Senthooran Rajamanoharan, Max Tegmark, and Neel Nanda. 2025. Are sparse autoencoders useful? a case study in sparse probing. In *Forty-second International Conference on Machine Learning*.

Geonhee Kim, Marco Valentino, and André Freitas. 2025. Reasoning circuits in language models: A mechanistic interpretation of syllogistic inference. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 10074–10095.

Connor Kissane, Robert Krzyzanowski, Arthur Conmy, and Neel Nanda. 2024. Attention saes scale to gpt-2 small. Alignment Forum.

János Kramár, Tom Lieberum, Rohin Shah, and Neel Nanda. 2024. Atp*: An efficient and scalable method for localizing llm behaviour to components. *arXiv preprint arXiv:2403.00745*.

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.

Aaron Mueller, Atticus Geiger, Sarah Wiegreffe, Dana Arad, Iván Arcuschin, Adam Belfki, Yik Siu Chan, Jaden Fried Fiotto-Kaufman, Tal Haklay, Michael Hanna, Jing Huang, Rohan Gupta, Yaniv Nikankin, Hadas Orgad, Nikhil Prakash, Anja Reusch, Aruna Sankaranarayanan, Shun Shao, Alessandro Stolfo, and 4 others. 2025. MIB: A mechanistic interpretability benchmark. In *Forty-second International Conference on Machine Learning*.

Max Müller-Eberstein, Rob van der Goot, Barbara Plank, and Ivan Titov. 2023. Subspace chronicles: How linguistic information emerges, shifts and interacts during language model training. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13190–13208, Singapore. Association for Computational Linguistics.

Neel Nanda. 2023. Attribution patching: Activation patching at industrial scale. `https://www.neelnanda.io/mechanistic-interpretability/attribution-patching`.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. Progress measures for grokking via mechanistic interpretability. In *The Eleventh International Conference on Learning Representations*.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, and 1 others. 2022. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005a. The proposition bank: An annotated corpus of semantic roles. *Comput. Linguist.*, 31(1):71–106.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005b. The proposition bank: An annotated corpus of semantic roles. *Comput. Linguist.*, 31(1):71–106.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. 2021. Grokking: Generalization beyond overfitting on small algorithmic datasets. In *ICLR MATH-AI Workshop*.

Beatrice Santorini and Anthony Kroch. 2007. The syntax of natural language: An online introduction.

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

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR.

Aaquib Syed, Can Rager, and Arthur Conmy. 2024. Attribution patching outperforms automated circuit discovery. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 407–416, Miami, Florida, US. Association for Computational Linguistics.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, and 3 others. 2024. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline. In *Annual Meeting of the Association for Computational Linguistics*.

Curt Tigges, Michael Hanna, Qinan Yu, and Stella Biderman. 2024. Llm circuit analyses are consistent across training and scale. In *Advances in Neural Information Processing Systems*, volume 37, pages 40699–40731. Curran Associates, Inc.

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*.

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

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

A Linguistic Background: Predicate–Argument Structure and Thematic Roles

A.1 Predicate–Argument Structure

Predicates, typically verbs, denote events or states and introduce *argument positions* corresponding to event participants (Santorini and Kroch, 2007). In *"The courier delivered the package to the office with the truck"*, the predicate *deliver* evokes a transfer event with participants such as the actor (*courier*), the transferred entity (*package*), the destination (*office*), and the means (*truck*). These participants may be realised as subjects, objects, or prepositional phrases, but surface form does not uniquely determine semantic function: "with X" may express INSTRUMENT (*with the truck*) or COMITATIVE (*with Mary*), and the same role may occur in different syntactic configurations (e.g., dative alternation: *"give the book to Mary"* vs. *"give Mary the book"*).

A.2 Thematic Roles (θ-Roles)

Thematic roles provide a shallow semantic representation that captures how participants relate to an event (Fillmore, 1976). Roles such as AGENT, THEME, GOAL, INSTRUMENT, LOCATION, TIME, and BENEFICIARY support abstraction across syntactic frames and lexical variation. This level of representation identifies *who did what to whom, where, and when* without committing to deeper logical structure (Jurafsky and Martin, 2025). Shallow semantics refers to the intermediate layer commonly modelled in semantic role labelling (Jurafsky and Martin, 2025).

A.3 Predicate–Argument Binding

Predicate–argument binding is the process of assigning thematic roles to the appropriate argument tokens. It forms the structured substrate on which more complex semantic composition builds: without correct role assignment, higher-level interpretation (e.g., quantification, scope, or discourse reference) cannot proceed. Our study isolates this binding mechanism, focusing on shallow semantics, and does not address deeper semantic phenomena such as quantifier scope or anaphora.

A.4 Computational Instantiation in Transformers

In transformer models, predicate–argument binding might arise through: (i) attention heads routing information between predicates and their arguments, (ii) distributed representations encoding role–filler associations, or (iii) localised circuits whose coordinated activity is *causally necessary* for role prediction. Our study tests for such circuits by combining causal edge-attribution patching, temporal emergence analysis, and cross-model comparison. Our goal is to determine whether transformers encode thematic roles via *computationally indispensable* mechanisms that are structurally localised, temporally trackable, and partially conserved across architectures.

Relevance for interpretability. If transformers encode predicate–argument binding via compact circuits, this suggests that meaningful semantic abstractions emerge naturally during training. Such findings provide mechanistic links between representation learning and the acquisition of linguistic structure, and offer principled targets for editing semantic behaviour.

B Dataset Generation Formalisation and Intervention Specifications

B.1 Task: Role-Cross Next-Token Prediction

We construct **role-cross minimal pairs** to isolate the semantic role processing. Each pair consists of two incomplete prompts that differ only in their role-indicating scaffold:

$$\begin{aligned} x^{(r)} &= \text{“The agent verb theme scaffold}^{(r)}\text{”} \\ x^{(s)} &= \text{“The agent verb theme scaffold}^{(s)}\text{”} \end{aligned} \tag{1}$$

where scaffold$^{(r)}$ (e.g., “to the”, “about the”) indicates the semantic role r of the next token. The two contexts are constructed such that:

- In clean context $x^{(r)}$, target token $y^{(r)}$ is role-appropriate and most probable
- In corrupted context $x^{(s)}$, a **different** token $y^{(s)}$ appropriate for role s should be most probable
- Both $y^{(r)}$ and $y^{(s)}$ are drawn from a cross-role lexicon, ensuring they are valid fillers for their respective roles, but not for each other's roles

Evaluation Task. The model performs role-cross prediction correctly if it predicts the role-appropriate target in each context:

$$\begin{aligned} \text{Accuracy} = \mathbb{1}[y^{(r)} = \arg\max_v P_\theta(v \mid x^{(r)})] \\ \wedge \mathbb{1}[y^{(s)} = \arg\max_v P_\theta(v \mid x^{(s)})] \end{aligned} \tag{2}$$

This measures whether the model correctly binds different role fillers based solely on the role scaffold, holding agent, verb, and theme constant.

Example.

$$\begin{aligned} x^{(\text{GOAL})} &= \text{“The driver sent the wall to the”} \\ &\rightarrow y^{(\text{GOAL})} = \text{“office”} \\ x^{(\text{TOPIC})} &= \text{“The driver sent the wall about the”} \\ &\rightarrow y^{(\text{TOPIC})} = \text{“plan”} \end{aligned}$$

The scaffolds “to the” and “about the” have the same token length (parity), but should activate different role-specific vocabularies.

B.2 Frame-Based Template Construction

We construct role-cross pairs inspired by PropBank (Palmer et al., 2005a) and FrameNet (Baker et al., 1998) annotations using frame-based templates. Each template consists of:

- **Frame**: Semantic structure (e.g., TRANSFER, COMMUNICATION)
- **Verb**: Single-token predicate (e.g., “sent”, “prepared”)
- **Scaffold**: Role-indicating preposition phrase (e.g., “to the” for GOAL, “about the” for TOPIC)
- **Agent**: Single-token subject (e.g., “driver”, “worker”)
- **Theme**: Single-token object (e.g., “wall”, “package”)
- **Target**: Role-specific single-token filler (e.g., “office” for GOAL, “plan” for TOPIC)

Single-Token Constraint. All lexical items must tokenise to exactly one token when preceded by a space (GPT-NeoX/Llama convention). We validate using the target model's tokeniser and filter out multi-token words. This ensures: (i) precise position alignment for activation patching, and (ii) unambiguous attribution to specific lexical items.

Token Parity Enforcement. For each role-cross pair $(x^{(r)}, x^{(s)})$, we enforce **strict token-level parity**: $|\text{toks}(x^{(r)})| = |\text{toks}(x^{(s)})|$. This is achieved by:

1. Grouping scaffolds by token length (e.g., 2-token: "to the", "in the", "about the"; 1-token: "at", "on")
2. Only pairing roles whose scaffolds have matching token lengths
3. Keeping agent, verb, and theme constant across pairs
4. Rejecting pairs that violate parity after substitution

Token parity is **essential for EAP-IG**, as activation differences $\Delta_u = \mathbf{z}_u^{(r)} - \mathbf{z}_u^{(s)}$ require position-aligned representations.

Role-Specific Lexicons. Each semantic role has a curated lexicon of plausible fillers:

- **Goal**: Places and people ("office", "student", "school")
- **Location**: Places ("kitchen", "office", "park")
- **Instrument**: Tools ("hammer", "knife", "drill")
- **Material**: Substances ("steel", "wood", "stone")
- **Topic**: Abstract concepts ("plan", "idea", "issue")
- **Beneficiary**: People ("student", "client", "friend")

Lexicons are designed such that tokens are **role-discriminative**: strongly preferred in their primary role but less probable in other roles (e.g., "hammer" is a good INSTRUMENT but bad TOPIC).

B.3 Generation Procedure

For each target role r and desired sample size N:

1. Sample target token $y^{(r)}$ from role r's lexicon
2. Identify corrupt role $s \neq r$ such that:
   - Scaffolds for r and s have matching token lengths (parity constraint)
   - Role s has a distinct lexicon (ensures different target $y^{(s)}$)
3. Sample verb-agent-theme triple compatible with both roles
4. Construct clean prefix $x^{(r)}$ with scaffold$^{(r)}$
5. Construct corrupted prefix $x^{(s)}$ by replacing scaffold$^{(r)}$ with scaffold$^{(s)}$
6. Sample foil token $y^{(s)}$ from role s's lexicon
7. Validate:
   - Token parity: $|\text{toks}(x^{(r)})| = |\text{toks}(x^{(s)})|$
   - No leakage: Neither $y^{(r)}$ nor $y^{(s)}$ appears in prefixes
   - All tokens are single-token
8. If validation passes, add $(x^{(r)}, x^{(s)}, y^{(r)}, y^{(s)})$ to dataset
9. Repeat until N valid pairs obtained (patience limit: $30N$ attempts)

B.4 Filtering Procedure

We filter generated pairs to retain only examples where the model predicts the role-appropriate target in **both** contexts. This ensures circuits are functionally active: the model successfully performs role-specific binding in both clean and corrupted contexts. Examples where either prediction is incorrect are discarded, as they would not reflect active role processing.

B.5 Dataset Statistics

All data is in English and after filtering with models, we obtain:

- **Token parity**: 100% (all pairs have matching token counts)
- **Dual prediction accuracy**: 100% (by construction, post-filtering: model predicts the correct target in both clean and corrupted contexts)
- **Cross-role coverage**: Each role has at least $\geq$450 examples

Table 5 provides a per-role and model breakdown.

Examples per Role	*Pythia-14M*	*Pythia-410M*	*Pythia-1B*	*LLaMA-1B*
Goal	845	895	1052	491
Location	904	673	975	815
Source	667	1206	505	612
Path	802	894	707	959
Instrument	459	1460	1212	1098
Beneficiary	773	837	621	1502
Topic	1120	1323	491	1179
Time	557	700	465	712

Table 5: Role-cross dataset statistics after filtering for all models. All examples satisfy: (i) strict token parity between clean and corrupted contexts, (ii) model predicts target correctly in both contexts.

C Computation and parameters specifications

C.1 Hyperparameter Selection

All hyperparameters were selected based on circuit size constraints, computational feasibility, and robustness to measurement noise. We report choices for attribution, sparsity measurement, emergence detection, and cross-scale comparison.

Attribution (EAP-IG).

- **Integrated gradients steps**: 5. We use only 5 steps due to the large number of training checkpoints analysed. Ablation tests (not shown) confirmed 5 steps provide stable attributions whilst maintaining computational tractability for multi-checkpoint analysis.
- **Metric**: Negative log-probability of the role-appropriate target token in the clean context: $\mathcal{L} = -\log P_\theta(y^{(r)} \mid x^{(r)})$. This loss quantifies the model's confidence in correct role binding.

Sparsity and Localization (RQ1).

- **Top-K node mass**: $K \in \{5, 10, 20\}$. We report the fraction of total attribution mass captured by the top-K highest-mass nodes. These values span from highly concentrated cores (K=5) to broader component sets (K=20).
- **Gini coefficient**: Computed over all in-circuit node masses (nodes with non-zero attribution). Higher Gini indicates more unequal mass distribution (tighter concentration).
- **Rationale**: All discovered circuits contain fewer than 40 active nodes at convergence, making K=20 a natural threshold capturing approximately 50% of the component space whilst focusing on high-importance nodes.

Emergence Dynamics (RQ2).

- **Indispensability threshold**: $M(\mathcal{C}_t) - M(\mathcal{E}_t) < 0$ for ≥ 2 consecutive checkpoints. Circuit performance must fall persistently below baseline to avoid transient noise.
- **Change-point detection**: Two-segment piecewise linear regression applied to faithfulness and Top-20 mass trajectories. Bootstrap resampling (n=1,000) estimates 95% confidence intervals. Minimum segment length: 2 checkpoints (prevents overfitting to single-step noise).
- **Consolidation criterion**: Jaccard overlap ≥ 0.6 between top-K=20 node sets over a 2-step sliding window. This threshold balances sensitivity (detects stabilisation) against noise (ignores minor fluctuations).

Cross-Scale Comparison (RQ3).

- **Node/edge overlap**: Top-K=30 components by absolute attribution mass. We increase K from 20 (used in RQ1/RQ2) to 30 for cross-scale comparison because larger models (410M, 1B) have more active nodes; K=30 ensures we compare substantive component sets whilst maintaining focus on high-importance nodes.
- **Spectral distance**: Computed on top-K=50 edges using the lowest k=20 Laplacian eigenvalues. We use more edges (K=50) for spectral analysis than overlap (K=30) because eigenvalue computation requires sufficient connectivity to yield stable spectra. The first 20 eigenvalues capture low-frequency flow structure whilst remaining computationally tractable.
- **Rationale**: Since all circuits contain <40 active nodes at convergence, K=30 captures ∼75% of the component space. This "hard 50% rule" ensures overlap metrics reflect substantive similarity rather than trivial peripheral agreement. For

edges, K=50 balances spectral stability with focus on high-attribution connections.

Data Filtering.

- **Correctness criterion**: Retain only examples where the model predicts the role-appropriate target correctly in **both** clean and corrupted contexts at baseline (full model). Formally: $y^{(r)} = \arg\max_v P_\theta(v \mid x^{(r)})$ AND $y^{(s)} = \arg\max_v P_\theta(v \mid x^{(s)})$.
- **Rationale**: This dual-correctness filter ensures discovered circuits are functionally active—the model successfully performs role binding in both contexts, guaranteeing circuits supporting this capability are present and engaged. Analysing only correctly predicted examples is standard practice in mechanistic interpretability (Wang et al., 2023; Conmy et al., 2023) as it isolates functional mechanisms rather than failure modes.

C.2 Computational Constraints.

All experiments were conducted on an NVIDIA RTX A6000 GPU. Attribution over multiple checkpoints required approximately 5 min per role for *Pythia-1b*. The choice of 5 IG steps was necessary to complete the temporal analysis within feasible compute budgets whilst maintaining attribution stability, as confirmed by spot-checks with higher step counts on selected checkpoints.

C.3 Software Specification.

Experiments were conducted using Python 3.11.13 with NumPy 1.26.4, scikit-learn 1.7.0, scipy 1.15.3, seaborn 0.13.2, tokenizers 0.21.1, torch 2.7.1, transformer-lens 2.16.1, and transformers 4.52.4, and trace 0.2.0.

C.4 Models Specifications.

All models were obtained from Hugging Face (Wolf et al., 2019) and used under their respective intended use, following their respective licenses: Llama 3.2 (Meta Llama 3 Community), Pythia Models (Apache license 2.0), we summarise their key characteristics in Table 6. Pythia models were mainly pre-trained on English data. LLama, on the other hands, has additional multilingual capabilities, including, English, German, French, Italian, Portuguese, Hindi, Spanish, and Thai. The models employ the following tokenisation approaches: Llama 3.2 uses SentencePiece-based BPE, combining 100K tokens from Tiktoken3 with 28K additional tokens to enhance multilingual performance, while Pythia employs GPT-NeoX.

Model	Parameters	Layers	D_{model}	Heads	Activation
Pythia 14M	1.2M	6	128	4	gelu
Pythia 410M	302M	24	2048	16	gelu
Pythia 1B	805M	16	2048	8	gelu
LLaMA3.2 1B	1.1B	16	2048	32	SiLU

Table 6: Summary of model architectures. **Parameters** is the total number of trainable parameters; **Layers** is number of transformer layers; $\mathbf{D_{model}}$: size of word embeddings and hidden states; **Heads**: number of self-attention heads; and **Activation**: activation function used in feedforward layers.

D EAP-IG formalisation

EAP-IG combines the causal faithfulness of activation patching (Meng et al., 2022; Wang et al., 2023) with the path-sensitivity of integrated gradients (Sundararajan et al., 2017), addressing key limitations of alternative approaches. We detailed its steps below.

D.1 Transformers as Acyclic Graphs

We adopt the graph-theoretic representation from Hanna et al. (2024), modelling transformer computation as a directed acyclic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ at module$\times$position granularity. A node $u \in \mathcal{V}$ corresponds to a specific module output:

$$u \in \mathcal{V} \iff u = \big(\text{type},\, \ell,\, h,\, i\big), \quad \text{type} \in \{\text{AttnHead}, \text{MLP}\}, \tag{3}$$

where ℓ is layer index, i is the token position in the sequence, and h is head index for attention heads; $h = \varnothing$ for MLPs. The activation $\mathbf{z}_u \in \mathbb{R}^d$ is the contribution of that module to the residual stream at position i.

An edge $(u \to v) \in \mathcal{E}$ exists if $\mathbf{z}_u$ is linearly mixed into the pre-activation input $\mathbf{s}_v$ of node v via the residual stream and layer normalisation. This yields a fine-grained graph where each edge corresponds to a specific causal pathway between modules.

D.2 EAP-IG Scoring Procedure

For each clean/corrupt pair, let $\mathbf{x}^{(c)}$ and $\mathbf{x}^{(r)}$ be the input embedding sequences for $x^{(c)}1:t$ and $x^{(r)}1:t$, respectively, and $\mathbf{z}_u^{(c)}$ be the activation of node u under $x_{1:t}^{(c)}$, and $\mathbf{z}_u^{(r)}$ be activation of node u under $x_{1:t}^{(r)}$. We compute causal edge attributions through the following protocol:

1. **Cache activations:** Run the model on both versions to obtain $\mathbf{z}_u^{(c)}$ and $\mathbf{z}_u^{(r)}$ for all nodes u.
2. **Compute source deltas:** $\Delta_u = \mathbf{z}_u^{(r)} - \mathbf{z}_u^{(c)}$.
3. **Interpolate inputs:** Define the straight-line interpolation $\mathbf{x}(\alpha) = \mathbf{x}^{(r)} + \alpha(\mathbf{x}^{(c)} - \mathbf{x}^{(r)})$ for $\alpha \in [0, 1]$.
4. **Gradient sampling:** For m evenly spaced α_k and destination node v, at each step k, run a forward pass with input $\mathbf{x}(\alpha_k)$ and compute the gradient of the loss with respect to v's pre-activation input: $\mathbf{g}_v^{(k)} = \partial L(\mathbf{x}(\alpha_k))/\partial \mathbf{s}_v \in \mathbb{R}^d$.
5. **Integrate:** Average over m to obtain the integrated gradient estimate $\bar{\mathbf{g}}_v = \frac{1}{m}\sum_{k=1}^{m} \mathbf{g}_v^{(k)}$.
6. **Edge score:** $S^{\mathrm{IG}}_{u\to v} = \Delta_u^\top \bar{\mathbf{g}}_v \in \mathbb{R}$.

Intuitively, $S^{\mathrm{IG}}_{u\to v}$ approximates the first-order change in the loss if the contribution of u to v's input were replaced by its corrupted counterpart, *averaged* along the naturalistic path from corrupt to clean inputs. By integrating over α, the estimate reduces sensitivity to local saturation at the clean point and captures non-linear response accumulated along the path. We selected $m = 5$, as we saw diminishing returns on higher values, similar to the results in (Hanna et al., 2024).

D.3 Score Normalisation and Aggregation.

To enable fair comparison across layers and modules with different activation scales, we report both raw and normalised attribution scores. The normalised score for edge $(u \to v)$ is:

$$\tilde{S}^{\mathrm{IG}}_{u\to v} = \frac{\Delta_u^\top \bar{\mathbf{g}}_v}{\|\Delta_u\|_2 \, \|\bar{\mathbf{g}}_v\|_2 + \varepsilon}, \quad \varepsilon = 10^{-8}. \tag{4}$$

Unless otherwise stated, aggregated statistics are computed on raw scores; normalization is used for scale-invariant concentration measures.

For role-specific analysis, we aggregate edge scores into role-layer heatmaps by summing absolute scores over edges sharing a destination module:

$$\mathsf{Importance}^{(r)}(\mathrm{layer} = \ell, \mathrm{head} = h) = \sum_{\substack{(u\to v)\in\mathcal{E}: \\ v=(\mathrm{AttnHead},\ell,h,\cdot)}} \left|S^{\mathrm{IG}}_{u\to v}\right|, \tag{5}$$

and analogously for MLPs.

D.4 Circuits Evaluation metrics

Faithfulness: the proportion of the clean–corrupt discrimination preserved by $\mathcal{C}$:

$$\mathrm{Faithfulness}(\mathcal{C}) = \frac{M(\mathcal{C}) - M(\emptyset)}{M(\mathcal{E}) - M(\emptyset)}, \tag{6}$$

where $M(\mathcal{C})$ is the metric in Section 3.2 under $\mathcal{C}$, $M(\mathcal{E})$ under all edges patched, and $M(\emptyset)$ under none patched.

Temporal Consistency. Jaccard stability of top-K edge sets between checkpoints t and t':

$$\mathrm{Stability}(\mathcal{C}_t, \mathcal{C}_{t'}) = \frac{|\mathcal{C}_t \cap \mathcal{C}_{t'}|}{|\mathcal{C}_t \cup \mathcal{C}_{t'}|}. \tag{7}$$

High stability indicates that once a role circuit emerges, it persists across training.

Bootstrap CI for Change-point We estimate a split point t in (x = step, y = faithfulness) by minimising the summed residuals of two OLS lines on $[x_1, \ldots, x_t]$ and $[x_{t+1}, \ldots, x_n]$ with a minimum segment length of 3. For a 95% CI, we resample pairs (x, y) with replacement, sort by x, re-fit the split, map the bootstrap split $x_t^\star$ back to the original grid, and take the 2.5/97.5 percentiles over 1,000 replicates. We report the point estimate $\hat{t}_c$ and the percentile CI over steps.

E Metric Definitions and Interpretation

Setup. All metrics are computed on the *in-circuit* subgraph for each *role* and *training step*. Let $G = (V, E)$ be a directed graph whose vertices $v \in V$ are components (attention heads $a\ell.h$, MLPs $m\ell$, special nodes `input`/`logits`), and whose edges $e = (u \to v) \in E$ carry an attribution score $s(e) \in \mathbb{R}$ and a type $\tau(e) \in \{\mathrm{Q}, \mathrm{K}, \mathrm{V}, \mathrm{Flow}\}$. Unless stated, strength uses absolute attribution $|s(e)|$.

Node mass. Incident absolute attribution:

$$\begin{aligned} \mathrm{mass}(v) &= \sum_{e\in\mathrm{Inc}(v)} |s(e)|, \\ \mathrm{Mass}(G) &= \sum_{v\in V} \mathrm{mass}(v). \end{aligned} \tag{8}$$

Note: $\mathrm{Mass}(G) = 2\sum_{e\in E} |s(e)|$ since each edge contributes to two endpoints. We also report **Total mass** as a proxy for role salience at a step.

Sparsity & Targeting (per role $\times$ step)

Top-K node-mass proportion.

$$\begin{aligned} \mathrm{TopK}(K) &= \frac{\sum_{i=1}^{K} m_{(i)}}{\sum_{i=1}^{|V|} m_{(i)}}, \\ & m_{(1)} \geq \cdots \geq m_{(|V|)}. \end{aligned} \tag{9}$$

Range $[0, 1]$; higher $\Rightarrow$ stronger sparsity. We report $K \in \{5, 10, 20\}$.

Top-P coverage. Minimal K such that $\mathrm{TopK}(K) \geq P$, for $P \in \{0.80, 0.90, 0.95\}$. Lower K indicates higher sparsity.

Gini coefficient (node mass). Standard Gini on nonnegative masses; range $[0, 1]$ ($1 =$ all mass on one node). Primary comparator for **sparse localisation** (RQ1).

comparable across graphs of different sizes.

Structural / Connectivity (per role $\times$ step)

Nodes, Edges. $|V|$ and $|E|$ of the in-circuit graph.

Density. We use Network's directed density:

$$\mathrm{density}(G) = \frac{|E|}{|V|(|V| - 1)} \in [0, 1], \quad (10)$$

assuming no self-loops. (*Implementation:* `nx.density`.)

Reciprocity. Fraction of directed edges participating in reciprocated pairs:

$$\mathrm{recip}(G) = \frac{L_{\leftrightarrow}}{L}. \quad (11)$$

Average out-degree / weighted out-degree.

$$\overline{\deg^+} = \tfrac{1}{|V|} \sum_v \deg^+(v),$$
$$\overline{\deg_w^+} = \tfrac{1}{|V|} \sum_v \sum_{(v \to u) \in E} |s(v \to u)|. \quad (12)$$

Edge-type fractions.

$$\mathrm{frac}_T = \frac{|\{e \in E : \tau(e) = T\}|}{|E|}, \quad T \in \{\mathrm{Q}, \mathrm{K}, \mathrm{V}, \mathrm{Flow}\}, \quad \sum_T \mathrm{frac}_T = 1. \quad (13)$$

Bridges (undirected projection). Count edges whose removal disconnects the *undirected* projection of G (structural bottlenecks).

Layer span.

$$\mathrm{layer_span} = \max_{v \in V} \mathrm{layer}(v) - \min_{v \in V} \mathrm{layer}(v). \quad (14)$$

Average betweenness centrality. $\overline{C_B} = \frac{1}{|V|} \sum_v C_B(v)$ on the directed graph (normalized; may use sampling for efficiency).

Emergence & Stability (per role)

Detectability t_{det} (optional). First step where faithfulness exceeds an early-phase baseline plus 2σ (baseline/variance from the first 2 checkpoints), persisting for ≥ 2 checkpoints.

Indispensability t_{ind}. Earliest step where ablating the discovered circuit yields a statistically significant performance drop that *persists* for at least 2 subsequent checkpoints.

Change-point $\hat{t}_c$ (with bootstrap CI). Two-segment least-squares on $y_t \in \{\mathrm{faithfulness}, \mathrm{TopK}(K)\}$ with a minimum segment length of 3. We report $\hat{t}_c$ with a nonparametric bootstrap 95% CI.

Consolidation t_{cons}. Earliest step post-$\hat{t}_c$ where Top-K node sets stabilise: $\mathrm{Jaccard}(V_t^{(K)}, V_{t'}^{(K)}) \geq 0.6$ for a 3-step window (persistence $= 2$), with K=20.

Cross-scale / Cross-role Similarity

Node/edge overlap across models. Mean Jaccard of Top-K node/edge sets between two models, averaged over common checkpoints. (*Defaults:* K=30.)

Spectral similarity. For symmetrised, weighted Laplacians ($w_{uv} = |s(u \leftrightarrow v)|$),

$$d_{\mathrm{spec}}(G_i, G_j) = \mathrm{RMSE}\big(\lambda_{1:k}(G_i), \lambda_{1:k}(G_j)\big),$$

where $\lambda_{1:k}$ are the k smallest eigenvalues. Lower is more similar. (*Defaults:* build undirected graphs from the Top-50 edges by $|s|$, k=20.)

Within-model role overlap. Jaccard of Top-K nodes between roles at fixed steps (specialisation vs. shared scaffolding).

Computation Conventions

- Mass/strength metrics use absolute attributions $|s(e)|$; sign-sensitive analyses are reported separately.
- Type fractions follow RDF edge labels $\tau(e) \in \{\mathrm{Q}, \mathrm{K}, \mathrm{V}, \mathrm{Flow}\}$.
- Density uses `nx.density` on directed graphs (no self-loops). Bridges are computed on the undirected projection.

Metric	Range	High	Low
Top-K mass	$[0, 1]$	concentrated circuit (RQ1)	diffuse attribution
Top-P coverage (K)	$\mathbb{N}$	few nodes capture P (sparse)	many nodes needed
Gini (mass)	$[0, 1]$	strong sparsity (RQ1)	uniform mass
Density	$[0, 1]$	saturated links (post-$\hat{t}_c$)	sparse links
Reciprocity	$[0, 1]$	feedback motifs	feed-forward routing
Avg out-degree	$\mathbb{R}_{\geq 0}$	broad fan-out	narrow fan-out
Avg weighted out-degree	$\mathbb{R}_{\geq 0}$	strong influence spread	weak influence
Edge-type mix	simplex	routing vs. residual balance	—
Bridges	$\mathbb{N}$	bottlenecks (ablation targets)	redundancy
Layer span	$\mathbb{N}$	deeper integration	shallow circuit
Avg betweenness	$[0, 1]$ (norm.)	coordinator hubs	flat routing
Top-K Jaccard (step)	$[0, 1]$	persistent circuit	unstable set
Cross-model Jaccard	$[0, 1]$	architectural consistency	family/scale drift
Spectral distance d_{spec}	$\mathbb{R}_{\geq 0}$	similar flow geometry	divergent geometry

Table 7: Interpretation guide for graph metrics, with expected high and low values meaning.

Causal Flow Visualisation The causal flow diagrams (e.g., Figure 2) visualise the dominant information pathways within discovered circuits. We construct a directed graph $G = (V, E)$ where nodes V represent model components (e.g., attention heads $a_{\ell,h}$, MLP layers m_ℓ, and output logits) and edges E represent causal attribution paths with weights w_{uv} quantifying the contribution of component u to component v. **Edge selection.** Rather than visualising the complete circuit graph, we filter to the top-k edges by attribution magnitude $|w_{uv}|$ among those marked as in-circuit. This selective visualisation serves two purposes: (1) it highlights the *dominant* computational pathways that account for the majority of causal effect, as circuits exhibit high concentration, and (2) it ensures interpretability, as dense graphs obscure rather than illuminate mechanistic structure. We use quantile-based thresholds (95th percentile by default) with a minimum edge count (12) to ensure sufficient context for interpretation. **Graph layout.** Nodes are positioned via multipartite layout by layer depth ℓ, flowing left-to-right from inputs ($\ell = -1$) through transformer layers to logits ($\ell = L + 1$). Edge attributes encode: width $\propto |w_{uv}|$ (attribution strength), colour by edge type (Query/Key/Value composition versus residual flow), and style (solid for positive attribution, dashed for negative suppression). This representation exposes: (i) critical computational pathways for each semantic role, (ii) layer-wise concentration of circuit activity, and (iii) coordination patterns between attention mechanisms versus direct residual connections.

Interpretation Cheat Sheet

See Table 7, with full method algorithm in 1.

Default parameters used in the paper. Unless otherwise noted: consolidation uses K=20, Jaccard ≥ 0.6, persistence $= 2$; cross-model overlap uses K=30; spectral similarity uses Top-50 edges and k=20 eigenvalues.

F Full Results

We add the full results for *Pythia-14m*, *Pythia-410M* and *LLaMA-1B*. We note that all experiments were repeated on five different random seeds, and the reported results are the averaged graphs per model.

F.1 RQ1: Full Localisation Results

This section provides the full localisation analyses for all semantic roles, model scales, and training checkpoints, complementing the representative results in the main text. For each role, we report (i) mass–concentration statistics, (ii) sparsity and coverage metrics, and (iii) stability of component sets over checkpoints.

Mass concentration and sparsity. Table 8 reports the Top–K mass ($K \in \{5, 10, 20\}$) for all roles and models at the final checkpoint (143K steps). Across all settings, Top–20 nodes capture between 83% and 99% of attribution mass, confirming that role circuits remain highly concentrated even in larger scales.

Coverage: minimal node set sizes. Table 9 reports the smallest k achieving 80%, 90%, and 95% mass. Across all roles and models, fewer than 30 nodes suffice for 95% coverage, again confirming that circuits remain compact even in 1B-scale architectures.

Algorithm 1 COMPASS: Causal-Temporal Circuit Discovery

Require: Model checkpoints $\{\theta_t\}_{t=0}^{T}$, role-cross dataset D, role $r \in \mathcal{R}$, top-K threshold

Ensure: Circuit $\mathcal{C}_t^{(r)}$ for each t; emergence times $(t_{\text{ind}}, \hat{t}_c, t_{\text{cons}})$

Phase 1: Causal Localisation (EAP-IG)
for each checkpoint $t = 0, \ldots, T$ **do**
  Compute CNP scores $\{\Delta_{\theta_t}(y; r, s)\}$ for all $(x^{(r)}, x^{(s)}) \in D$
  Run EAP-IG to obtain edge attributions $\{S^{\text{IG}}_{u\to v}\}_{e\in\mathcal{E}}$ (Appendix D)
  Normalise: $\tilde{S}^{\text{IG}}_{u\to v} \leftarrow S^{\text{IG}}_{u\to v} / \sum_e |S^{\text{IG}}_e|$
  Extract circuit: $\mathcal{C}_t^{(r)} \leftarrow \text{TopK}(\{|\tilde{S}^{\text{IG}}_{u\to v}|\}, K)$
end for
Phase 2: Temporal Monitoring
for each checkpoint $t = 0, \ldots, T$ **do**
  Compute faithfulness F_t via ablation (See Sec. 3.2)
  Compute stability S_t via Jaccard (See Sec. 3.2)
  Compute structural metrics: Top-K node mass, Gini coefficient
end for
Phase 3: Emergence Detection
Detect indispensability: $t_{\text{ind}} \leftarrow \min\{t : M_t(\mathcal{E}) - M_t(\mathcal{C}_t) > \epsilon \text{ for } \geq 2 \text{ steps}\}$
Estimate functional transition: $\hat{t}_c \leftarrow \arg\max_t R^2(\text{PiecewiseLinear}(\{F_t\}))$ with bootstrap CIs
Detect consolidation: $t_{\text{cons}} \leftarrow \min\{t : S_t \geq 0.6 \text{ for } \geq 2 \text{ steps}\}$
return $\{\mathcal{C}_t^{(r)}\}_{t=0}^{T}, (t_{\text{ind}}, \hat{t}_c, t_{\text{cons}})$

Cross-scale similarity of component sets. Figures 6 and 7 summarise how role circuits align across PYTHIA–14M, PYTHIA–410M, and PYTHIA–1B. Node-level Top–30 overlaps vary by role, ranging from J_V≈0.24–0.31 between 14M and 1B to J_V≈0.37–0.45 between 410M and 1B. INSTRUMENT shows the strongest cross-scale alignment (up to J_V=0.45 for 410M↔1B), indicating that its circuits are both highly compact and structurally similar at larger scales. Edge-level overlaps are consistently lower (J_E≈0.06–0.18), confirming that connection patterns diverge more than the identity of high-importance nodes. Spectral distances are smallest for the 410M↔1B pairs (d_{spec}≈0.006–0.018) and larger when 14M is involved (d_{spec}≈0.05–0.15), suggesting that while small models already recruit broadly similar components, the overall information-flow geometry only stabilises once scale increases.

	PYTHIA-14M			PYTHIA-410M			LLAMA-1B		
Role	T-5	T-10	T-20	T-5	T-10	T-20	T-5	T-10	T-20
BENEFICIARY	0.465	0.769	**0.988**	0.345	0.58	**0.906**	0.4	0.657	**0.939**
INSTRUMENT	0.445	0.596	**0.833**	0.445	0.596	**0.833**	0.484	0.697	**0.895**
LOCATION	0.368	0.557	**0.841**	0.368	0.557	**0.841**	0.413	0.636	**0.882**
TIME	0.501	0.713	**0.985**	0.445	0.610	**0.852**	0.443	0.677	**0.961**

Table 8: Top-K mass concentration at final checkpoint (143K steps). Values show fraction of total attribution mass captured by K highest-mass nodes, demonstrating strong localisation across roles and model scales.

Role	PYTHIA-14M	PYTHIA-410M	LLAMA-1B
BENEFICIARY	11 / 13 / 15	16 / 20 / 23	14 / 17 / 22
INSTRUMENT	11 / 14 / 17	19 / 24 / 28	15 / 21 / 25
LOCATION	12 / 15 / 17	19 / 23 / 27	17 / 21 / 25
TIME	10 / 14 / 18	18 / 24 / 28	14 / 17 / 20

Table 9: **Minimal node count required for coverage at the final checkpoint (143K steps).** Entries report the smallest number of nodes (k) needed to capture 80%, 90%, and 95% of total attribution mass for each role and model, where small number of nodes constitute the majority of mass.

Summary. Across all roles and model scales, we find that semantic-role circuits localise to highly compact subgraphs whose attributional mass is dominated by a small, stable subset of nodes. Final-step Top–20 mass consistently exceeds 0.83 (and reaches 0.97–0.99 for several roles; Table 8), and only ∼15–28 nodes are required to capture 95% of total mass (Table 9). These component sets remain stable across training checkpoints, with only minor turnover in the highest-mass nodes. Structural metrics further reveal a characteristic pattern of refinement: active node sets contract slightly over training while density increases, indicating consolidation around a pruned but increasingly interconnected core. Together, these results show that role circuits are both *spatially localised* and *structurally coherent*, forming compact causal pathways that become progressively more organised as training proceeds.

F.2 RQ2: Full Emergence Dynamics

This appendix complements the main-text analysis by providing the full emergence dynamics results for PYTHIA–14M. We analyse three signals across training: faithfulness, indispensability, and structural consolidation, and compare them to change-

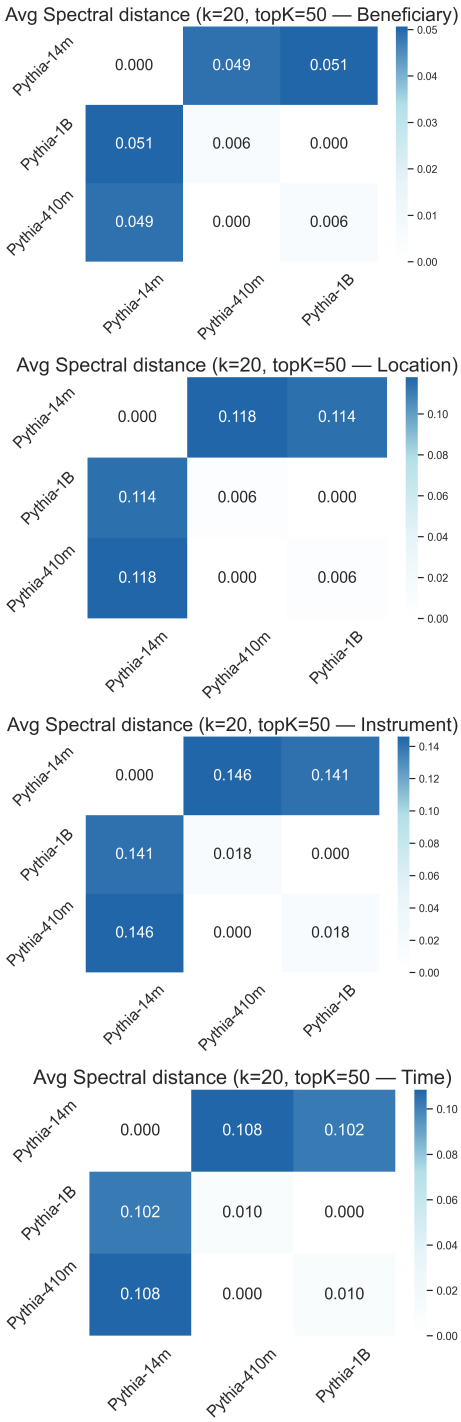

Figure 6: **Cross-scale spectral distance of role circuits.** Heatmaps show pairwise spectral distances between role circuits across PYTHIA model scales (14M, 410M, 1B), computed from the Top-50 edges and the first 20 eigenvalues. Lower values between larger models indicate greater similarity in information-flow geometry despite possible topological differences.

point estimates derived from piecewise linear fits.

Indispensability. All roles eventually become causally necessary, but the timings vary by more than four orders of magnitude. INSTRUMENT circuits are useful from the first checkpoint (t_{ind}=0), LOCATION becomes indispensable early (1k steps), and GOAL follows at 5K. TIME emerges only at mid-training (71k steps), reflecting exceptionally delayed functional reliance. These heterogeneous timings indicate that roles differ substantially in both cue learnability and the optimisation pressure required for the model to commit to a stable causal pathway.

Faithfulness trajectories. Faithfulness curves exhibit pronounced non-monotonicity, with early rises, sharp drops, and late partial recoveries. INSTRUMENT peaks early and declines; TIME rises initially, crashes after 5–10k steps, and partially recovers; GOAL and LOCATION show smoother but still multi-phase dynamics. Importantly, these fluctuations do not correspond to abrupt structural changes: functional utility is unstable even when the underlying structure is already highly concentrated.

Structural consolidation. In contrast to faithfulness, structural metrics evolve smoothly. Top-K node sets stabilise extremely early for most roles (512 steps for GOAL, LOCATION, and TIME; 2K for INSTRUMENT). Thus, the model identifies the relevant components long before those components become functionally indispensable. Structural consolidation is therefore not the bottleneck in circuit emergence.

Change-point analysis. Two-segment piecewise fits to Top-K mass trajectories yield very wide bootstrap confidence intervals (e.g. TIME: [54, 5k]; INSTRUMENT: [64, 8K]), confirming that structural sparsification is gradual rather than concentrated at a discrete transition. Visual inspection of sparsity curves reveals smooth monotonic growth without identifiable inflection points. The structural substrate evolves continuously even when functional utility displays sharp changes.

Final-step sparsity. By convergence (143K steps), all circuits are highly compact: Top-20 mass ranges from 0.83–0.98, and only 15–18 nodes suffice to cover 95% of attribution. BENEFICIARY is the most concentrated (95% mass in

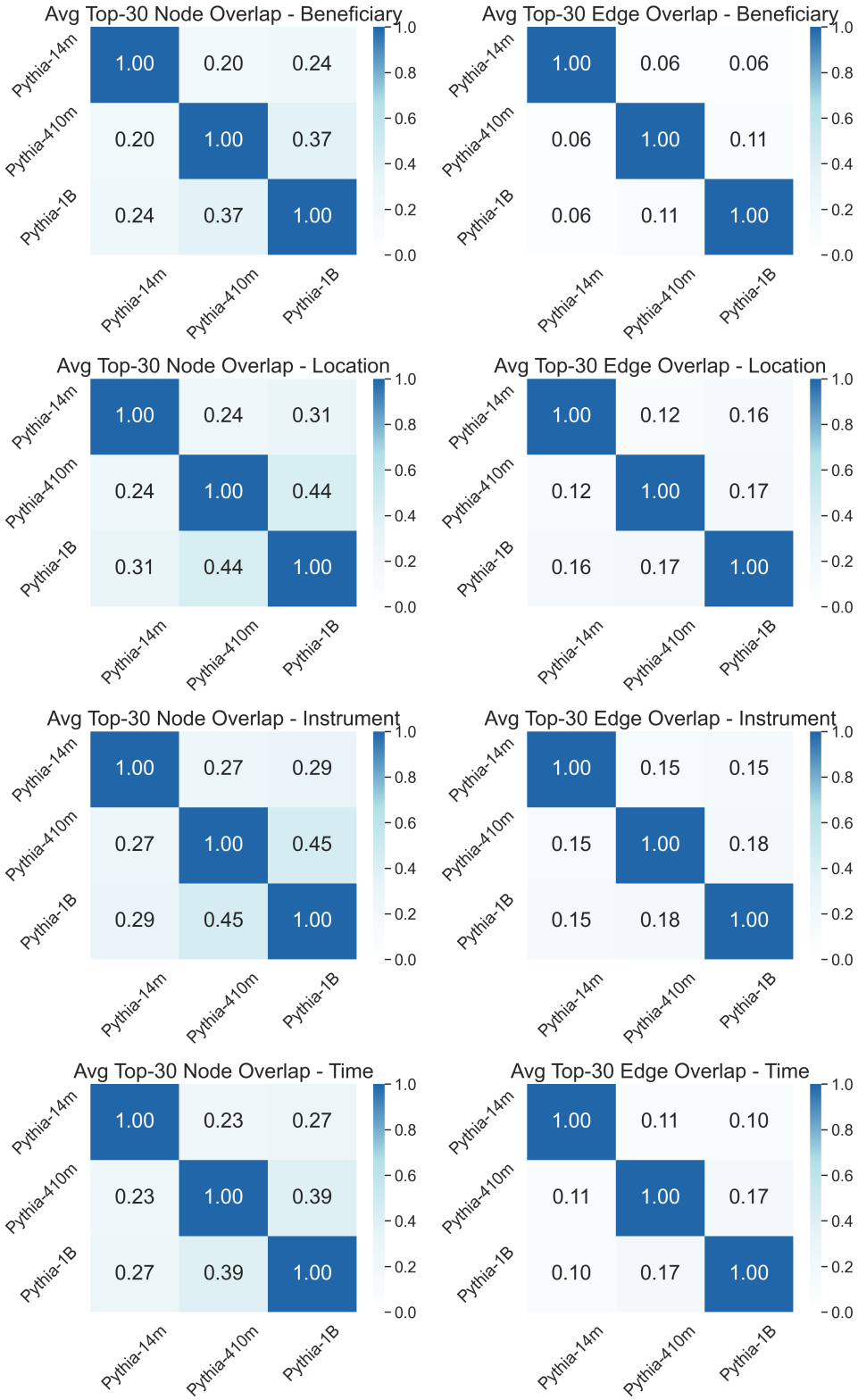

Figure 7: **Cross-scale overlap of high-importance components.** Heatmaps report average Top-30 node overlap (left) and Top-30 edge overlap (right) between role circuits across PYTHIA model scales. Node overlap consistently exceeds edge overlap, indicating reuse of key components with divergent routing across scales.

15 nodes), whereas INSTRUMENT and LOCATION have slightly broader but still compact top-tiers.

Summary. Across all roles in PYTHIA–14M, emergence is a gradual process in which *structural* properties stabilise early and monotonically, while *functional* utility develops in a noisy, role-dependent manner. Indispensability can lag far behind consolidation, indicating that circuits may be structurally "pre-allocated" long before the model consistently relies on them. Together, these results support the conclusion that semantic-role circuits do not undergo discrete phase transitions but instead emerge through continuous refinement shaped by heterogeneous task signals and optimisation dynamics.

F.3 Additional Semantic Roles

In addition to the four core roles analysed in the main paper, we applied the COMPASS pipeline to four further predicate–argument relations frequently used in semantic role labelling, including PATH, SOURCE, and TOPIC. Figure 9 reports their emergence trajectories (faithfulness, density, Top-K mass) across training checkpoints, and Tables 10, 11, and 12 demonstrate the indispensability, concentrations and change-point estimation for them, respectively.

Overall, these supplementary roles exhibit comparable qualitative patterns identified for the main roles. Structural metrics (density and Top-K mass) increase smoothly and monotonically, consistent with gradual sparsification rather than discrete phase transitions. In contrast, faithfulness trajectories are highly variable: PATH exhibits high volatility: faithfulness increases through early training, then spikes mid-training (step 71k) before collapsing back, indicating heavy competition with other predictive cues. SOURCE shows high faithfulness early training, the declines and remains consistently throughout training despite structural consolidation. TOPIC displays an early spike before settling into moderate stability. GOAL maintains moderate faithfulness with fluctuations mid-training, never achieving the stability seen in others. These patterns match the dissociation seen in the primary roles: *structural sparsity is stable and monotonic, but functional utility is noisy and task-pressure dependent*. As before, structure often stabilises long before a role becomes functionally useful.

Role	t_{ind} (steps)	t_{cons} (steps)
GOAL	0	1,000
PATH	64	1,000
SOURCE	512	5,000
TOPIC	512	5,000

Table 10: **Indispensability and consolidation timings for additional roles (PYTHIA-1B model).** The table reports the earliest step at which each role becomes indispensable (t_{ind}), and the step at which structural consolidation occurs (t_{cons})

Role	Top-5	Top-10	Top-20	k for 80/90/95%
GOAL	0.507	0.708	**0.931**	14 / 18 / 22
PATH	0.348	0.594	**0.889**	16 / 21 / 25
SOURCE	0.421	0.655	**0.917**	15 / 19 / 23
TOPIC	0.444	0.657	**0.908**	15 / 20 / 25

Table 11: **Final-step concentration of additional role circuits.** Similar to results in the main paper, a small k values show a limited subset of nodes constituent the majority part of computation.

F.4 Circuit heterogeneity

While our results demonstrate that semantic role circuits consistently localise to compact subgraphs, the *internal organisation* of these circuits varies systematically across roles and training stages. This variation reflects both the diversity of semantic cues associated with different roles and the flexibility of the model's computational pathways. To characterise these differences, we examine the fine-grained structure of each circuit, its dominant components, routing patterns, and evolution over training, using causal-flow visualisations derived from the top-K=30 nodes, with edges ranked by attribution magnitude at the 95th percentile threshold. These analyses reveal systematic and role-dependent heterogeneity in circuit structure, both at convergence and throughout developmental trajectories.

F.4.1 Architectural Stratification at Convergence

We identify four recurrent circuit architectural types at convergence (step 143K), distinguished by attention-head addition, integration depth, and reliance on value composition operations. Table 13 summarises the final circuit types, reporting node counts, attention head involvement, and dominant computational pathways.

Type 1: Lexical pattern matching. GOAL and PATH rely predominantly on MLP computation

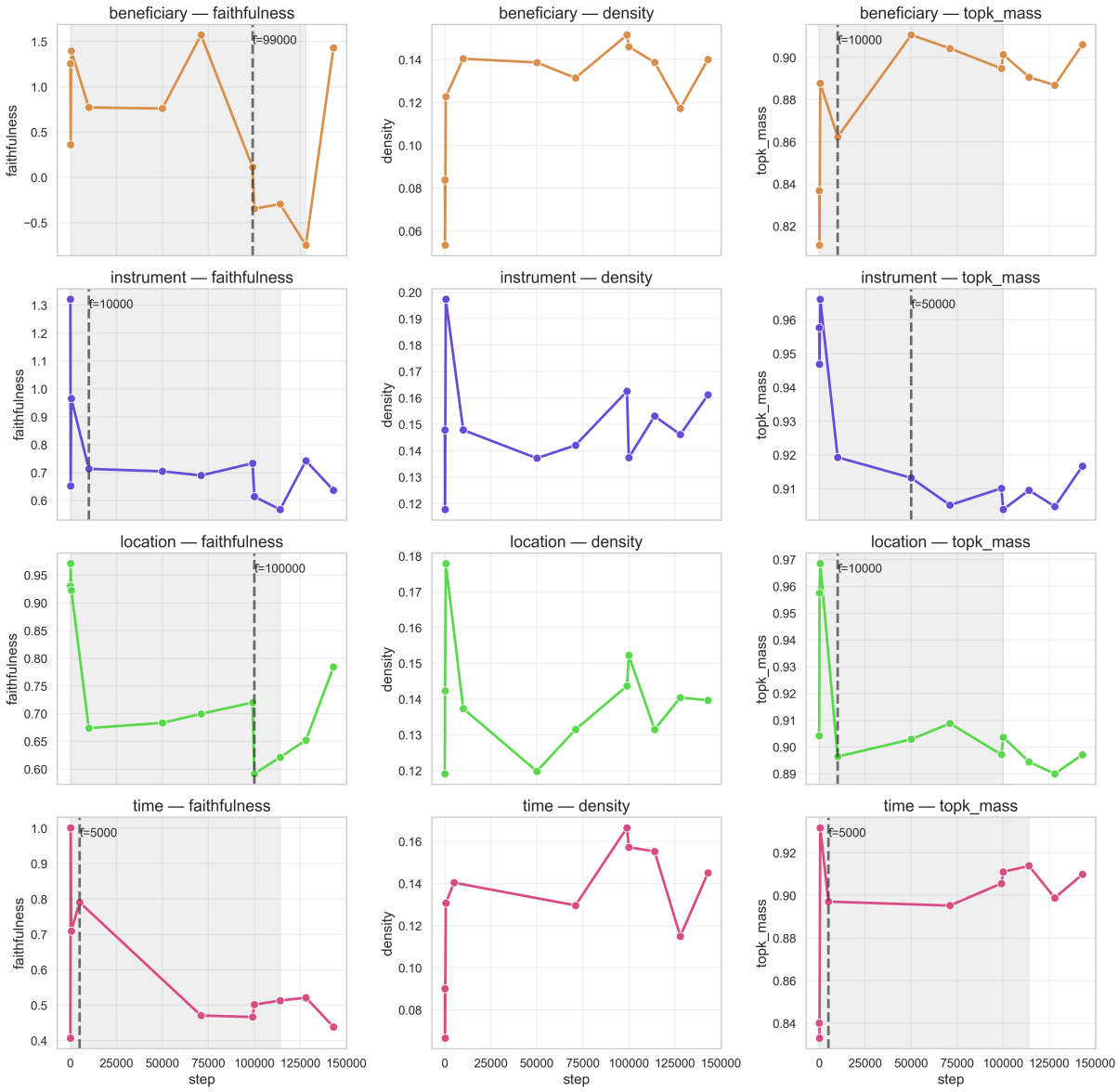

Figure 8: Emergence trajectories for each role in PYTHIA-1B. Left: faithfulness; middle: edge density; right: Top-K node mass (Top-20). Structural measures change smoothly over training, while faithfulness exhibits role-specific non-monotonicity.

Role	$\hat{t}_c$ (steps)	95% CI
GOAL	10,000	[128, 10,000]
PATH	10,000	[128, 128,000]
SOURCE	5,000	[128, 99,000]
TOPIC	5,000	[128, 99,000]

Table 12: **Change-point estimates for Top-K node mass.** Estimates are obtained via two-segment piecewise linear regression, with 95% bootstrap confidence intervals shown for each role; wide intervals indicate gradual structural evolution.

with minimal or zero attention. PATH exhibits minimal attention involvement (18 nodes, 2 heads), we hypothesise a0.h6 provides early syntactic composition, while a12.h2 performs targeted value composition for the final pre-logit refinement. The dominant causal flow remains through the MLP backbone, with attention providing what seems to be an auxiliary support. GOAL (18 nodes, 3 heads) implements a clean two-stage architecture: mid-layer feature extraction at a6.h0, followed by late-stage integration at a12.h2+V (with value composition edge) and final refinement at a14.h5. The circuit exhibits sparse connectivity with clear information bottlenecks at m6 and m12. Both roles mark a highly regular, closed set of expressions (e.g., path markers), enabling direct lexical classification without extensive need for contextual integration.

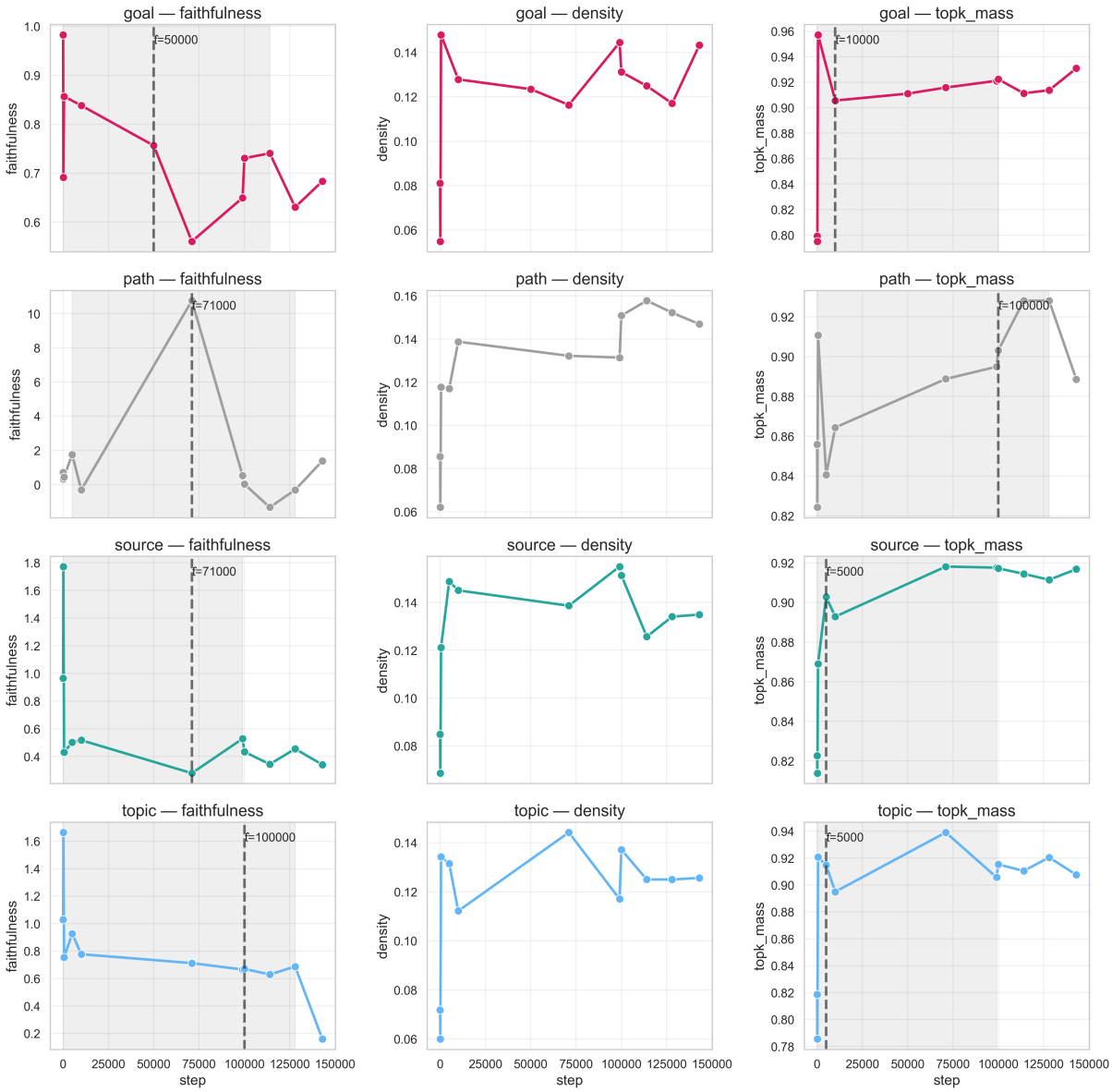

Figure 9: **Emergence trajectories for supplementary semantic roles for the PYTHIA-1B model.** Each row corresponds to a role; columns show faithfulness, density, and Top-K mass across training.

Type 2: Multi-stage compositional integration. TOPIC, and SOURCE converge to moderate-complexity architectures (19 nodes, 4–5 attention heads) characterised by systematic multi-stage processing, with m0 being connected to the majority of other nodes in the circuit. TOPIC (19 nodes, 4 heads) adds early integration of a0.h2, suggesting initial scaffold detection, then follows mid-to-late integration (a6.h0→a12.h2→a14.h5+V). Notably, value composition shifts to the final integration head (a14.h5+V) rather than at a12.h2, indicating part of feature extraction occurs at the pre-logit stage. SOURCE (19 nodes, 5 heads) exhibits the richest attention architecture in this group: a3.h7, assumably for early instrumental/causal cue detection, a6.h0 for mid-layer feature gathering, and three late-stage heads (a12.h2, a14.h5, a15.h5) for final disambiguation. The circuit displays clear information flow through mid-layer MLPs (m3, m6) converging to late-stage integration at m12 and m15, with a15.h5 providing a final refinement pathway.

Type 3: Balanced hybrid architectures. INSTRUMENT and LOCATION exhibit moderate complexity (20 nodes, 4 heads each) with balanced MLP-attention integration. INSTRUMENT follows a hierarchical pattern: a3.h7 for early cue detection (likely detecting instrumental markers), a6.h0 for mid-layer extraction, and late-stage integration at

Type	Role	Nodes	Heads	Architecture
1	Path	18	2	Predominantly MLP with early framing (a0.h6) and late value composition (a12.h2+V)
	Goal	18	3	Mid-layer extraction (a6.h0) with late integration (a12.h2+V, a14.h5)
2	Topic	19	4	Early frame detection (a0.h2), mid-layer (a6.h0), late integration (a12.h2, a14.h5+V)
	Source	19	5	Rich multi-stage: early (a3.h7), mid (a6.h0), late (a12.h2, a14.h5, a15.h5)
3	Instrument	20	4	Mid-to-late extraction and integration (a3.h7, a6.h0, a12.h2, a14.h5)
	Location	20	4	Distributed integration (a1.h0, a12.h2, a13.h4, a14.h5)
4	Beneficiary	22	6	Complex late-stage architecture (a1.h0, a6.h0, a12.h2, a14.h5+V, a15.h0, a15.h5)
	Time	22	7	Diverse multi-stage architecture with integration of several components (a3.h7, a3.h5, a6.h0, a9.h2, a10.h1, a12.h2+V, a14.h5+V, a15)

Table 13: **Circuit categorisation across semantic roles at convergence.** Circuits stratify into four architectural types reflecting the computational demands of each role. Node counts reflect circuits extracted at the 95th percentile threshold with top-K=30 edges. "+V" indicates value composition edges; heads are listed in order of prominence in the causal-flow diagrams.

a12.h2 and a14.h5. The circuit exhibits a clear bottleneck at m6, with multiple pathways converging to m12 and m14. Notably, INSTRUMENT lacks value composition at convergence, relying instead on positional routing through residual connections. LOCATION shows a more distributed architecture with attention heads spread across early (a1.h0), mid-late (a12.h2, a13.h4), and final (a14.h5) layers. The circuit maintains sparse MLP connectivity (m1, m13, m14) with attention providing targeted integration at multiple depths. Like INSTRUMENT, LOCATION lacks value composition edges, suggesting both roles have do not rely as much as others on active feature extraction mechanisms in favour of simpler, more efficient routing strategies by convergence.

Type 4: Complex late-stage integration with distributed refinement. BENEFICIARY and TIME exhibit the most elaborate architectures at convergence (22 nodes each, 6–7 attention heads), maintaining rich late-stage connectivity with distributed processing across multiple layers. The circuit recruits a1.h0 for early processing, a6.h0 for mid-layer extraction, and four late-stage heads: a12.h2 for primary integration, a14.h5+V for feature extraction, and two heads (a15.h0, a15.h5) for final disambiguation. This architecture suggests BENEFICIARY requires parallel processing pathways to resolve persistent ambiguities, and probably the model maintains alternative hypotheses until the last computation step. The circuit's relative complexity (22 nodes vs. 16–20 for other roles) and late-stage density indicate that beneficiary marking, despite being syntactically constrained (e.g., "for . . . "), requires more elaborate compositional reasoning than other participant roles, likely to distinguish benefactive readings from alternative interpretations. TIME (22 nodes, 7 heads) exhibits similarly complex multi-stage processing with attention distributed across early (a3.h7), mid-layer (a6.h0, a9.h2, a10.h1), and late integration (a12.h2, a14.h5+V, a15). The circuit shows convergent information flow through mid-layer MLPs (m6, m9, m10) to late-stage integration nodes (m12, m14), with value composition at a14.h5 enabling feature extraction for temporal disambiguation. This architectural complexity is surprising given that temporal expressions often involve closed-class markers ("during the", "at the"), but likely reflects the need to distinguish temporal from locative interpretations of ambiguous scaffolds ("at the") and to resolve context-dependent temporal reference.

F.4.2 Developmental Trajectories Across Training

Circuit evolution from initialisation to convergence (step 143K) reveals role-specific developmental patterns. Figures 10, 11 and 12 present a sample of causal-flow diagrams at three key checkpoints, early training (step 32), mid-training (step 71000), and convergence (step 143000). These snapshots expose systematic differences in how semantic-role circuits emerge and stabilise. We study them for all roles below.

Universal early complexity followed by selective pruning. All roles begin training with diffuse connectivity and elevated node counts (15–23 nodes at step 32), reflecting initial hypothesis exploration. At initialisation, most circuits exhibit dense early-layer attention recruitment (multiple a0.h* heads) and extensive value composition edges, suggesting the model initially explores compositional integration strategies broadly across all roles. By

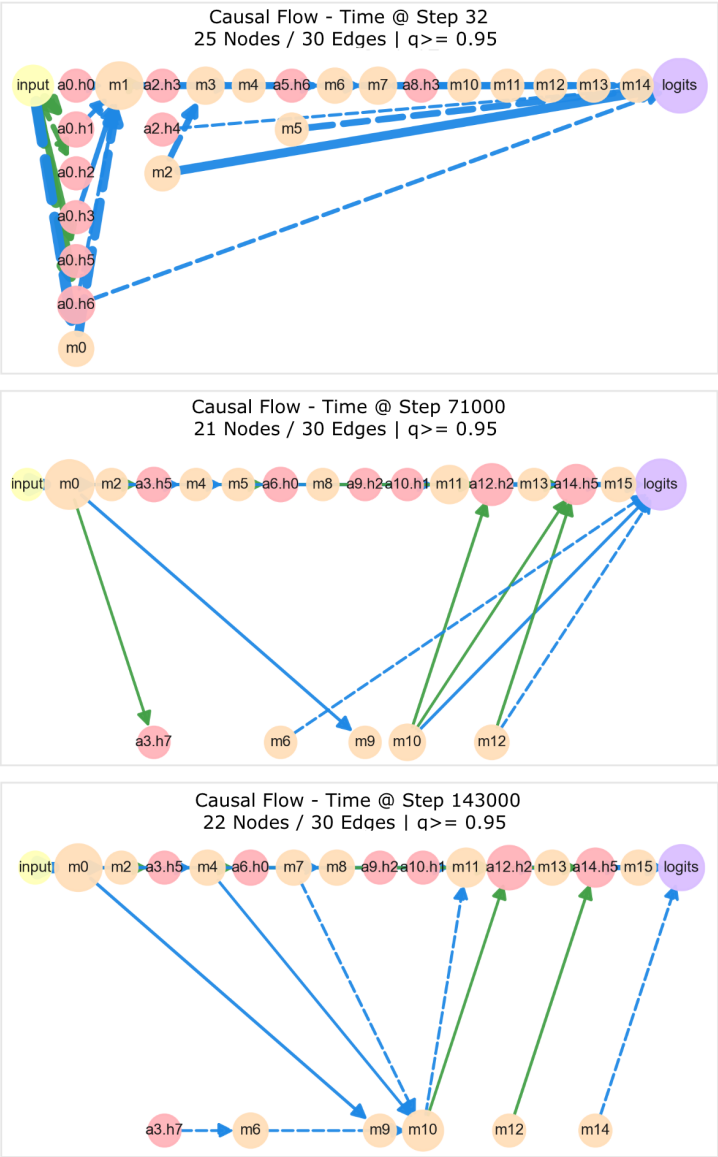

Figure 10: **Developmental trajectory for TIME role across training.** Circuit evolution from initialisation (step 32) through mid-training (step 71000) to convergence (step 143000) demonstrates progressive attention elimination.

mid-training (step 71000), circuits have begun to differentiate sharply: TIME achieves near-complete attention elimination (16 nodes, pure MLP), while BENEFICIARY maintains a rich multi-head architecture (21 nodes, 7 heads). At convergence (step 143000), final node counts (16–22) represent reductions from initialisation, with architectural consolidation complete and further refinement involving only edge-weight adjustments rather than topological restructuring.

Stratified consolidation and shared infrastructure. Roles follow distinct consolidation trajectories aligned with semantic complexity. What we assume are more **Template-matching roles** (GOAL, PATH) eliminate most attention: PATH retains minimal heads (a0.h6, a12.h2+V). **Integration-dependent roles** (INSTRUMENT, LOCATION) stabilise core architectures by mid-training with only minor late refinement, converging to 2–4 attention heads with mid-to-late integration. **Multi-phase processing roles** (TOPIC, SOURCE, BENEFICIARY) undergo non-monotonic reorganisation: TOPIC contracts then reinstates early framing (a0.h2) and late value composition (a14.h5+V); SOURCE adds a3.h7 only at convergence; BENEFICIARY and TIME maintain persistent complexity with many heads likely required for parallel disambiguation.

Across trajectories, shared integration hubs emerge at predictable stages: a12.h2 (7/8 roles)

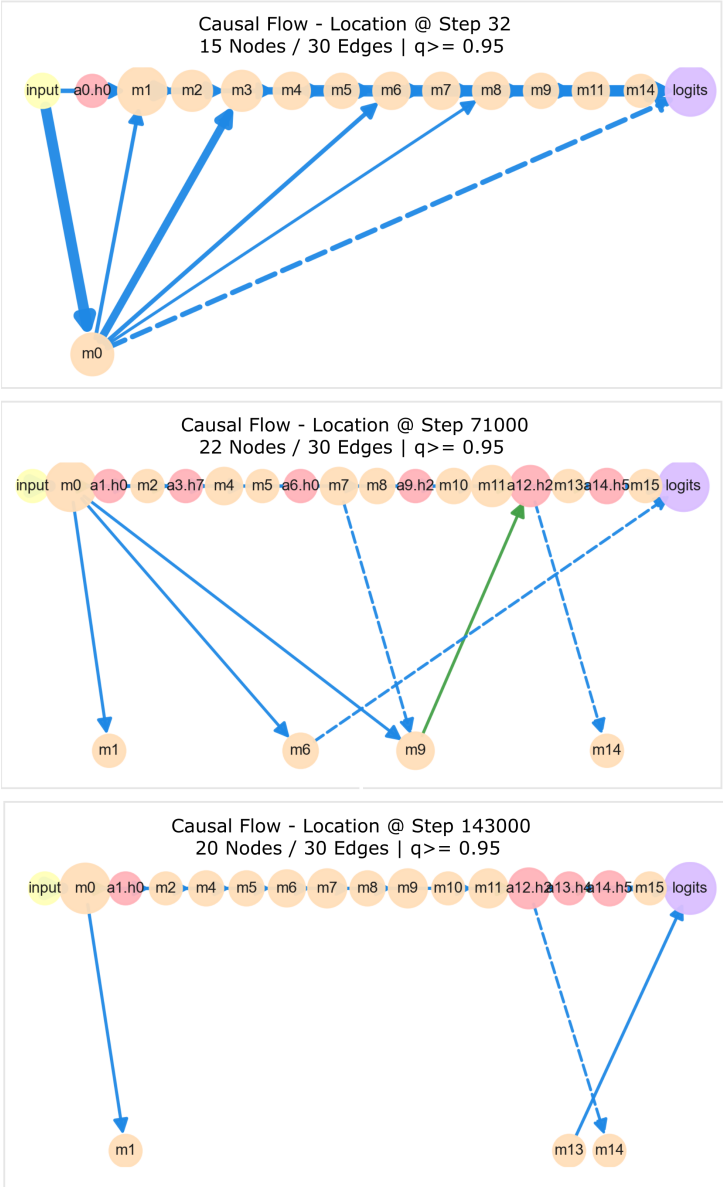

Figure 11: **Developmental trajectory for LOCATION role across training.** Circuit evolution follows an expand then contract pattern with mid-training exploration followed by late-stage pruning.

stabilises by mid-training as universal late-stage integrator; a14.h5 (7/8 roles) emerges slightly later for pre-logit refinement; a6.h0 (5/8 roles) provides mid-layer extraction for compositional/discourse roles. Early-layer attention (a0.h*, a1.h*) is systematically pruned except in PATH, TOPIC, LOCATION, and BENEFICIARY, where explicit frame detection remains necessary. Value composition follows more of explore/prune/retain dynamics: broad at initialisation (6/8 roles), divergent at mid-training (4 roles eliminate, 1 intensifies to 3 V-ops), and selective at convergence (4 roles: PATH, GOAL, TOPIC, BENEFICIARY). V-edges shift spatially from early layers (a0.h*, a3.h*) to late integration heads (a12.h2, a14.h5), indicating feature extraction is preserved only where disambiguation proves irreducible.

F.5 Paraphrase-Based Within-Role Scaffold Controls (Pythia–1B)

To test whether role circuits encode *abstract semantic roles* rather than overfitting to a particular lexicalised scaffold, we construct a small within-role paraphrase control set for two representative roles: LOCATION and INSTRUMENT, and we study them over our reference model PYTHIA–1B.

Paraphrase construction. For each filtered example $(x^{(r)}, y^{(r)})$ in the role-cross dataset (Appendix B), we construct a within-role paraphrase by

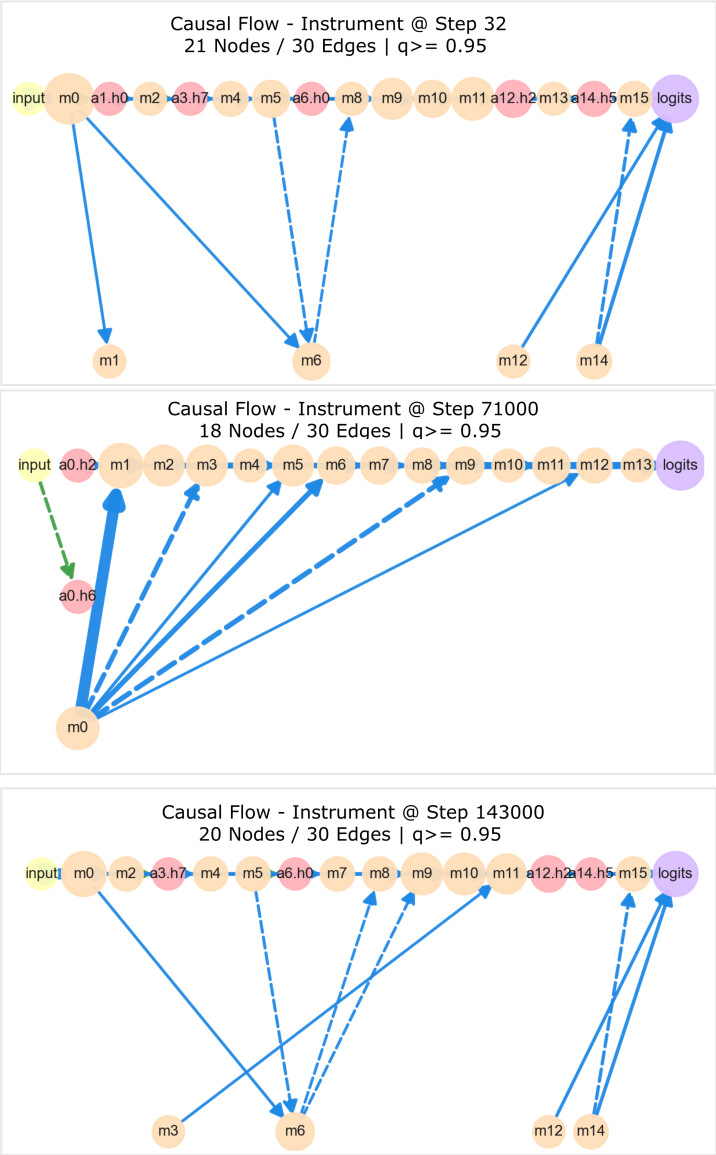

Figure 12: **Developmental trajectory for INSTRUMENT role across training.** Circuit evolution demonstrates stable mid-layer consolidation with minimal late-stage refinement.

replacing the original scaffold with an alternative:

$$x_{\text{para}}^{(r)} = \text{“The agent verb theme scaffold}^{(r')}\text{”} \tag{15}$$

by replacing $\text{scaffold}^{(r)}$ with an alternative scaffold $\text{scaffold}^{(r')} \in \mathcal{S}^{(r)}$ such that:

- $\text{scaffold}^{(r')} \neq \text{scaffold}^{(r)}$
- $|\text{toks}(\text{scaffold}^{(r')})| = |\text{toks}(\text{scaffold}^{(r)})|$ (parity constraint)
- agent, verb, theme, and target head $y^{(r)}$ are unchanged

For example,

$$\mathcal{S}^{(\text{LOCATION})} = \{\text{“in the”, “at the”, “near the”}\}.$$
$$\mathcal{S}^{(\text{INSTRUMENT})} = \{\text{“with the”, “using the”, “by the”}\}.$$

We then apply the same filtering criterion as for the main dataset (Section B): we retain only paraphrased prompts $x_{\text{para}}^{(r)}$ for which the model continues to predict the original role-appropriate target $y^{(r)}$ as the most probable next token. This yields a paraphrase set

$$\mathcal{D}_{\text{para}}^{(r)} = \{(x_{\text{para}}^{(r)}, y^{(r)})\} \tag{16}$$

in which all examples are (i) semantically equivalent to the originals at the level of role assignment,

(ii) realised with different surface scaffolds, and (iii) correctly handled by the base model.

Circuit consistency evaluation. For each role $r \in \{\text{LOCATION}, \text{INSTRUMENT}\}$, we recompute EAP-IG scores on the paraphrase set and compare them with the original role-cross circuit $\mathcal{C}^{(r)}$ using:

- Top-K node overlap ($K = 20$),
- Edge-weight rank correlation,
- Faithfulness of original circuits on paraphrased prompts.

Summary of control results. Table 15 summarises the results for INSTRUMENT and LOCATION. Across both roles, we observe:

Metric	n	Pearson r	p-value	Spearman ρ
Circuit Size (Nodes)	8	-0.066	0.876	0.049
Attention Heads	8	-0.283	0.497	-0.268
Sparsity (Top-20 Mass)	8	0.296	0.476	0.311
Concentration (Gini)	8	0.326	0.430	0.357
Consolidation Time (t_{cons})	8	-0.054	0.899	0.062

Table 14: Correlations between role frequency (sample size) and circuit properties at convergence (Pythia-1B, step 143K). All correlations show, indicating weak relationships between role frequency and circuit properties. Sample sizes range from 491 to 1,212 examples per role.

Role	Faithfulness	Top–20 Mass
INSTRUMENT	0.657 (+0.021)	0.925 (+0.009)
LOCATION	0.731 (−0.053)	0.920 (+0.023)

Table 15: Paraphrase-control evaluation for two roles in PYTHIA–1B. Scores reported for paraphrased inputs at the final training step. Parentheses denote the change relative to the original (non-paraphrased) dataset.

Circuit Property	n	Pearson r	p-value	Spearman ρ
Circuit Size (nodes)	8	0.346	0.401	0.439
Attention Heads	8	0.310	0.455	0.244
Sparsity (Top-20 mass)	8	0.106	0.803	0.095
Concentration (Gini)	8	0.430	0.288	0.381

Table 16: Corpus frequencies from 5,000 documents sampled from The Pile and the correlations between corpus scaffold frequency and circuit properties at convergence (Pythia-1B, step 143K). All correlations show $|r| \leq 0.430$, $p \geq 0.288$ (not significant).

1. **Stable faithfulness under paraphrase**: the original circuits maintain similar causal impact despite scaffold changes (Instrument: $+0.02$; Location: -0.05).
2. **Consistent sparsity structure**: Top–20 mass shifts by less than 0.03 for both roles, indicating that high-importance components remain largely unchanged.
3. **Robust abstraction beyond surface cues**: circuits respond similarly across distinct paraphrastic templates, supporting the interpretation that they encode predicate–argument binding rather than memorised lexical patterns.

These findings reinforce that the circuits identified by our study capture the semantic structure associated with predicate–argument roles, rather than superficial or memorised prepositional cues.

F.6 Role Frequency and Circuit Properties

F.6.1 Sample Size and Circuit Properties

To test whether circuit architecture is determined by the availability of samples in our filtered dataset, we examined correlations between sample sizes and circuit properties at convergence (Pythia-1B, step 143K). If circuit complexity were driven primarily by the volume of valid examples, we would expect positive correlations between sample sizes and measures of circuit size or structural complexity.

Correlation Analysis. Table 14 reports Pearson and Spearman correlations between filtered sample sizes and five circuit properties: circuit size (nodes), attention head count, sparsity (Top-20 mass), concentration (Gini coefficient), and consolidation timing (t_{cons}). All correlations show $|r| < 0.35$ with all $p > 0.4$, indicating negligible relationships between sample size and circuit architecture. The strongest correlation (Gini: $r = 0.326$, $p = 0.430$) remains statistically

Role-Type Stratification. In contrast to the absence of sample-size effects, circuit architecture stratifies clearly by role type (Table 13). Roles requiring complex disambiguation (Type 4: BENEFICIARY, TIME) develop the largest circuits (22 nodes, 6–7 attention heads) independent of their sample sizes (621 vs. 753 examples). Conversely, lexical pattern matching (Type 1: PATH) converges to a minimal architecture (18 nodes, 2 heads) despite moderate sample availability (707 examples). Multi-stage compositional roles (Type 2) and balanced hybrid architectures (Type 3) occupy intermediate positions, with circuit complexity determined by semantic processing demands rather than example count.

Method	Faithful	Path-spec.	Granularity	Scalable	Notes / Risks
Linear probing	○	×	token/residual	✓	Correlational, not causal
Attribution Patching (EAP) (Nanda, 2023; Syed et al., 2024)	○	✓	head/MLP edge	✓	Gradient-based; false negatives
AtP* (Kramár et al., 2024)	✓	✓	head/MLP edge	✓✓	improved faithfulness; residual false negatives
Temporal EAP-IG*	✓	✓	head/MLP edge	✓✓	Baseline/path sensitivity
Path patching (Goldowsky-Dill et al., 2023b)	✓	✓✓	head/MLP path	○	Expensive over checkpoints
Causal scrubbing (Chan et al., 2022)	✓✓	✓✓	hypothesis-level	×	High implementation cost
SAE (Templeton et al., 2024)	✓	✓	feature-level	○	SAE training; feature drift
Transcoders (Dunefsky et al., 2024)	✓	✓	MLP sublayer	○	Surrogate training; not causal
Circuit tracing / attribution graphs (Ameisen et al., 2025)	○	✓✓	feature–feature	×	Surrogate fidelity; prompt-specific

Table 17: Comparison of interpretability methods along criteria induced by RQ1/RQ2. ✓✓=strong; ✓=good; ○=partial; ×=weak.

The consistent absence of correlations between sample sizes and all circuit properties demonstrates that architectural complexity is not determined by the number of valid examples in our filtered dataset.

F.6.2 Scaffold Frequency in Training Corpus

To validate that the selected semantic roles are well-represented in the training data and to test whether circuit architecture can be explained by training signal strength alone, we analysed scaffold frequencies in the Pythia training corpus (The Pile (Gao et al., 2020)). We sampled 5000 documents (∼50M tokens) using stratified sampling (every 4,200th document) to ensure representative coverage across Pile subsets, and counted occurrences of the prepositional scaffolds used in our role-conditioned continuation task. We then correlated these corpus frequencies with circuit properties at convergence to test whether training frequency determines architectural complexity.

Corpus Frequencies. All roles are well-represented in the training data, ranging from 517.9 (TOPIC) to 5,498.9 (LOCATION) instances per million tokens. Locative and temporal roles exhibit the highest frequencies (LOCATION: 5,498.9/1M; TIME: 5,452.3/1M), reflecting the prevalence of spatial and temporal modification in natural language. Directional roles show intermediate frequencies (GOAL: 4,147.6/1M; SOURCE: 1,057.2/1M; PATH: 598.2/1M), while participant and propositional roles appear less frequently but remain well represented (INSTRUMENT: 1,990.5/1M; BENEFICIARY: 1,523.3/1M; TOPIC: 517.9/1M).

Correlation with Circuit Architecture. Table 16 reports Pearson and Spearman correlations between corpus scaffold frequencies and circuit properties at convergence (step 143K). All correlations are weak to moderate ($|r| \leq 0.430$) and statistically non-significant ($p > 0.28$), indicating that training frequency does not reliably predict circuit architecture. The strongest correlation (Gini coefficient: $r = 0.430$, $p = 0.288$) explains only 18% of variance and is not statistically significant. Circuit size and attention head count show weak positive trends ($r = 0.346$ and $r = 0.310$ respectively), while sparsity (Top-20 mass) shows virtually no relationship ($r = 0.106$, $p = 0.803$). We note that one of these relationships approaches statistical significance at $\alpha = 0.05$ level.

The absence of systematic frequency effects is most clearly demonstrated by roles with nearly identical corpus frequencies but divergent architectures. LOCATION and TIME appear at virtually the same frequency in the training data (5,498.9 vs. 5,452.3 per million tokens, a difference of 0.8%), but develop different circuit structures. LOCATION converges to a balanced hybrid architecture (20 nodes, 4 attention heads) with distributed integration across layers (a1.h0, a12.h2, a13.h4, a14.h5), while TIME develops a more complex multi-stage architecture (22 nodes, 7 attention heads) with rich attention involvement distributed across early (a3.h7), mid-layer (a6.h0, a9.h2, a10.h1), and late integration stages (a12.h2, a14.h5, a15). This architectural difference, despite a near-identical training signal, demonstrates that circuit organisation cannot be predicted from corpus frequency alone. The results indicate that instead of building circuits proportional to how often each role appears in training, models develop architectures tailored to the semantic processing demands of each role. This suggests that circuit organisation reflects functional requirements, the computational work needed to bind each semantic role, rather than statistical regularities in the training corpus.

G Method Comparison

We provide a summary comparing mechanistic interpretability methods, along with our choice in Table 17.