Tracking Representation Dynamics in Large Language Models with Persistent Homology

Naman Malhotra NAMAN.MALHOTRA24@IMPERIAL.AC.UK
Jay Ambadkar JAY.AMBADKAR24@IMPERIAL.AC.UK
Abhinav Gupta ABHINAV.GUPTA24@IMPERIAL.AC.UK
Kushal Kasivel KUSHAL.KASIVEL24@IMPERIAL.AC.UK
Abbas Schwarz ABBAS.SCHWARZ24@IMPERIAL.AC.UK
Kamillo Ferry K.FERRY@IMPERIAL.AC.UK
Anthea Monod A.MONOD@IMPERIAL.AC.UK
Imperial College London, South Kensington Campus, London SW7 2AZ

Abstract

Large language models are commonly aligned through supervised fine-tuning, yet little is known about how their internal representations evolve during this process. We study alignment dynamics using persistent homology by tracking the topology of activation spaces throughout fine-tuning. Across four transformer language models ranging from 1B to 7B parameters and three alignment objectives corresponding to helpful, harmless, and mixed training data, we find that the majority of topological reorganization occurs during the earliest stages of training. A dense checkpoint analysis reveals a transient peak in topological activity followed by rapid stabilization. We further show that different alignment objectives induce distinguishable topological trajectories, while instruction-tuned and pretrained models exhibit qualitatively different patterns of evolution. Our results suggest that persistent homology provides a complementary perspective on alignment, revealing representation-level changes that are not apparent from behavioral metrics alone.

Keywords: Persistent homology, Large Language Models, Representation Geometry, Alignment

1. Introduction

Large language models (LLMs) are typically aligned through supervised fine-tuning and preference-based training in order to improve helpfulness, harmlessness, and instruction-following behavior. While the behavioral effects of alignment have been extensively studied, considerably less is known about how the internal representations of a model evolve during the alignment process itself.

Recent work in mechanistic interpretability has shown that meaningful changes in a model's internal state can occur even when behavioral metrics remain unchanged. This phenomenon motivates the search for complementary approaches that characterize representation dynamics directly. Here, we study these dynamics using *persistent homology* (PH), a tool from topological data analysis that extracts multiscale topological structure from high-dimensional point clouds. Building on recent applications of PH to LLM activations, we investigate how the topology of latent activation spaces evolves during alignment fine-tuning.

We analyze four transformer models ranging from 1B to 7B parameters and track activation clouds across training checkpoints under helpful, harmless, and mixed alignment objectives derived from the Anthropic HH-RLHF dataset (Helpful and Harmless Reinforcement Learning from Human Feedback (Bai et al., 2022)), a standard alignment dataset containing human preference judgments. By computing PH at successive stages of training, we obtain a topological view of representation dynamics throughout the alignment process.

Contributions. Using persistent homology to study activation spaces throughout alignment fine-tuning, we make the following contributions:

- Across different model sizes, a dense checkpoint analysis reveals a transient peak in topological reorganization during early training, followed by rapid stabilization.
- Different alignment objectives induce distinguishable topological trajectories despite often producing similar behavioral outcomes.
- The observed dynamics depend strongly on the model's initial state, with instruction-tuned and pretrained models exhibiting qualitatively different patterns of evolution.
- We release a fully reproducible implementation of the complete analysis pipeline available at `tracking-representation-dynamics-with-persistent-homology` from activation extraction and persistent homology computation to statistical evaluation, to facilitate future topological studies of large language models.

Thus, our findings suggest that persistent homology provides a useful lens for studying representation dynamics during alignment and reveals aspects of fine-tuning that are not apparent from behavioral metrics alone.

2. Background

We briefly review PH and overview the context of understanding representation dynamics and alignment in LLMs.

2.1. Persistent Homology

Persistent homology (PH) is a tool from topological data analysis (Edelsbrunner et al., 2002; Zomorodian and Carlsson, 2005; Carlsson, 2009; Otter et al., 2017) that quantifies the multiscale topology of data. Given a point cloud X equipped with a distance metric, PH constructs a nested sequence of simplicial complexes called a *filtration* by progressively increasing a scale parameter $\varepsilon \geq 0$. In this work, we use the *Vietoris–Rips* filtration: at scale ε, two points are connected by an edge whenever their distance is at most ε, and higher-dimensional simplices are included whenever all pairwise distances between their vertices are at most ε. As ε increases, topological features such as connected components and loops appear and disappear. These features are summarized by a *barcode*, which records the birth and death scales (b_i, d_i) of each topological feature.

In our setting, PH is applied to activation point clouds derived from latent token representations in LLMs. We restrict attention to homological dimensions H_0 and H_1, corresponding to connected components and loops, respectively.

2.2. Related Work

LLMs are typically aligned through supervised fine-tuning and preference optimization in order to produce more helpful and harmless behavior (Bai et al., 2022; Ouyang et al., 2022). Although alignment often appears to modify only a small subset of model behaviors (Zhou et al., 2023; Jain et al., 2024; Lee et al., 2024; Arditi et al., 2024), substantial internal reorganization can occur before these behavioral changes become apparent (Barak et al., 2022; Nanda et al., 2023). Understanding the dynamics of internal representations during alignment therefore remains an important open problem.

PH has been used to characterize representations in deep neural networks (Naitzat et al., 2020; Rieck et al., 2019; Moor et al., 2020) and, more recently, in LLMs (Gardinazzi et al., 2025; Fay et al., 2026; Tang et al., 2026). In contrast to prior work, which has largely focused on static representations or specific training phenomena, we use PH to study the evolution of representation topology throughout alignment fine-tuning in contemporary LLMs.

3. Representation Topology Pipeline

We track the evolution of latent activation spaces throughout alignment fine-tuning. At multiple training checkpoints, we extract activation point clouds from a fixed evaluation set, compute persistent homology, summarize the resulting barcodes using topological descriptors, and compare their evolution across alignment objectives and model families.

Models and alignment objectives. We study four decoder-only transformer language models with a range of 1B–7B parameters: TinyLlama-1.1B (Zhang et al., 2024), Gemma-3-1B (Mesnard et al., 2024), Phi-3-mini-3.8B (instruction-tuned) (Abdin et al., 2024), and Mistral-7B (v0.1) (Jiang et al., 2023).

Each model is fine-tuned using Low-Rank Adaptation (LoRA) (Hu et al., 2022), freezing the pretrained model weights and performing low-rank updates. This reduces the number of trainable parameters compared to full fine-tuning under three alignment objectives derived from the HH-RLHF dataset (Bai et al., 2022). In all experiments, the same optimization hyperparameters and random seed are used, only differing in the training objective.

The dataset consists of pairs of candidate assistant responses together with human preference labels. We consider three training regimes: *helpful* (H), encouraging informative responses to user queries; *harmless* (S), encouraging safe and non-harmful behavior; and *mixed* (M), combining both objectives. These objectives provide distinct but related

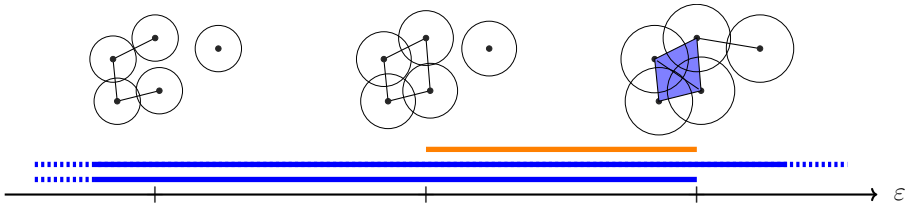

Figure 1: Example of a Vietoris–Rips filtration with three values of ε and its barcode. Blue bars represent connected components; the orange bar represents a loop.

alignment signals, allowing us to investigate how different forms of alignment influence the topology of latent activation spaces during fine-tuning.

Training checkpoints are saved every 50 optimization steps up to 300 steps for the 1B models, with longer schedules used for larger models. To study the earliest stages of training in greater detail, we additionally perform a dense checkpoint analysis every 5 steps up to step 50 (or the corresponding early-training window for larger models).

Activation point clouds and persistent homology. At each checkpoint, we evaluate the model on a fixed set of 750 prompts comprising three evaluation conditions: 250 harmful prompts (potentially unsafe requests drawn from the harmless-training subset of HH-RLHF), 250 helpful prompts (benign informational requests from the helpful subset), and 250 benign control prompts. We emphasize that evaluation conditions describe the prompts used to probe the model and are distinct from the training objectives. For each prompt and selected transformer layer, we extract the final-token hidden state and treat it as a point in the model's latent representation space. Repeating this procedure over all prompts yields an activation point cloud for a given layer and checkpoint. Following the methodology from Fay et al. (2026), we extract activations from five evenly spaced layers throughout the network depth.

For each activation point cloud, we compute its Vietoris–Rips complex using the Euclidean distance. We restrict attention to the 0- and 1-dimensional components (H_0 resp. H_1), corresponding to connected components and loops, respectively. PH was computed using the Ripser (Bauer, 2021) software.

Topological summaries. To obtain fixed-length representations suitable for comparison across checkpoints and objectives, each barcode is summarized by the same 41 features as in Fay et al. (2026). The summary combines barcode counts, persistence entropy, moments of the birth, death, and persistence distributions, and several scale-invariant statistics computed separately for H_0 and H_1. Among these, *persistence entropy* (Chintakunta et al., 2015) is the Shannon entropy of the normalized bar lengths, $-\sum_i p_i \log p_i$ with $p_i = \ell_i / \sum_j \ell_j$, where ℓ_i is the length of the ith bar. The full definitions of all 41 descriptors are given in Appendix B.

4. Analysis Framework

Each activation point cloud is repeatedly subsampled in order to estimate variability arising from finite prompt sets. Since these subsamples overlap and therefore do not constitute independent observations, all statistical inference is performed at the cell level rather than the subsample level. Treating overlapping subsamples as independent observations severely inflates significance (App. Fig. 13). Cell-level aggregation avoids this pseudoreplication. Throughout, a *cell* denotes a fixed combination of model, objective, evaluation condition, layer, and checkpoint.

Cell aggregation and effect sizes. For each cell, we average the $B = 64$ subsample summaries into a single mean vector $\bar{\mathbf{z}} = \frac{1}{B}\sum_{i=1}^{B} \mathbf{z}_i \in \mathbb{R}^{41}$. A (model, objective, condition, layer) trajectory is then represented by the sequence $\bar{\mathbf{z}}(t_0), \ldots, \bar{\mathbf{z}}(t_T)$. Features are standardized within each model prior to multivariate analysis.

We report both univariate and multivariate effect sizes. At the feature level, we use Hedges' g and Cliff's δ together with percentile-bootstrap 95% confidence intervals. For the full 41-dimensional representation, we use the *energy distance* (Székely and Rizzo, 2013), $\mathcal{E}(A,B) = \left(2\,\overline{\|a-b\|} - \overline{\|a-a'\|} - \overline{\|b-b'\|}\right)^{1/2}$, which remains stable in settings with relatively few cell-level observations.

Objective separation. To quantify separation between alignment objectives, we define the *trajectory dispersion* statistic

$$U := \sum_t \Big(|\bar{\mathbf{z}}_H(t) - \bar{\mathbf{z}}_M(t)|^2 + |\bar{\mathbf{z}}_H(t) - \bar{\mathbf{z}}_S(t)|^2 + |\bar{\mathbf{z}}_S(t) - \bar{\mathbf{z}}_M(t)|^2 \Big), \tag{1}$$

which measures the aggregate separation of the helpful, harmless, and mixed trajectories (U is reported in App. Table 2 and discussed in §5.2).

Significance is assessed via permutation tests over whole trajectories rather than individual subsamples. Because a single trajectory per objective yields a degenerate permutation test, we additionally perform three-seed replications for TinyLlama-1.1B, Gemma-3-1B, and Mistral-7B, producing nine trajectories per model for objective-separation analyses.

To distinguish separation in magnitude from separation in direction, we further characterize the *direction* of each objective's reorganization: For each objective, we define its topological displacement as the vector $\bar{\mathbf{z}}(t_T) - \bar{\mathbf{z}}(t_0)$ in the standardized 41-dimensional summary space. The *displacement-direction cosine* between two objectives is the cosine similarity of their displacement vectors, measuring whether fine-tuning moves the topology in the same direction.

Early-shock analysis. To quantify topological reorganization, we compare persistence diagrams at consecutive checkpoints using the Wasserstein distance. Given two persistence diagrams D, D', the order-p Wasserstein distance is

$$W_p(D, D') = \left(\inf_{\gamma} \sum_{x \in D} \|x - \gamma(x)\|_\infty^p \right)^{1/p}, \tag{2}$$

where γ ranges over partial matchings between D and D' in which unmatched points may be transported to the diagonal (Cohen-Steiner et al., 2010; Mileyko et al., 2011). We use $p = 2$ throughout, computed with the PERSIM library; see Kerber et al. (2017) for efficient computation. Applying (2) between consecutive checkpoints yields a scalar rate of topological change, which we refer to as the *topological velocity* v_t, computed separately for H_0 and H_1 barcodes. We summarize the *concentration* of topological change by

$$C := \frac{\sum_{t \le T/3} v_t}{\sum_t v_t}, \tag{3}$$

the fraction of total topological movement occurring within the first third of training (C is evaluated against the checkpoint-order null in App. Table 1). We use the first third as a coarse, model-agnostic notion of an early training phase, allowing comparisons across models with different numbers of checkpoints without committing to a particular change-point or peak location. We deliberately use a broad early-training window rather than the first peak itself, since peak location varies across models and objectives.

To assess whether the observed concentration is genuinely temporal rather than a consequence of total movement, we construct a checkpoint-order null by randomly permuting checkpoint order and recomputing both v_t and C (3).

Behavioral comparisons. To compare representation-level and behavioral dynamics, we track changes in the model's output distribution throughout fine-tuning. For each checkpoint, we compute the Kullback–Leibler divergence (Kullback and Leibler, 1951) from the initial model and define a behavioral velocity using the Jensen–Shannon divergence (Lin, 1991) between consecutive checkpoints, averaged over a fixed evaluation set. Here $\mathrm{JS}(P,Q) = \frac{1}{2}\mathrm{KL}(P\|M) + \frac{1}{2}\mathrm{KL}(Q\|M)$ with $M = \frac{1}{2}(P+Q)$ (see Appendix C.1). This yields a behavioral trajectory on the same checkpoint axis as the topological velocity, allowing for the comparison of timing of representation and behavioral change.

Controls. We perform several control analyses to assess the robustness and specificity of the observed topological phenomena. First, we compare PH against standard geometric summaries computed on the same activation clouds, including centroid drift, total variance, and mean pairwise distance. Second, we compare the observed topological signatures to an isotropic Gaussian baseline matched in dimension and sample size. Finally, we investigate the role of token selection by reconstructing activation clouds from the final 16 token states of each prompt and repeating the corresponding topological analyses. Additional robustness checks are reported in Appendix A.

5. Results and Discussion

Unless otherwise noted, all analyses use the second-to-last probed layer on the harmful evaluation condition. Features are standardized within each model and permutation tests use at least 2000 randomizations. We organize the results around three research questions: (RQ1) When does topological reorganization occur? (RQ2) How do alignment objectives differ? (RQ3) How do topological and behavioral dynamics relate?

5.1. Topology Reorganizes Early During Fine-Tuning (RQ1)

Across all models, the majority of topological reorganization occurs during the earliest stages of fine-tuning. A dense checkpoint analysis reveals a transient rise–peak–decay pattern in the H_1 Wasserstein velocity, followed by rapid stabilization (Fig. 2). The timing of this peak depends on model scale, occurring later in the 1B models and earlier in Phi-3 and Mistral-7B. Thus, while an early reorganization phase is present across all models, its precise timing varies with model size.

To assess whether this concentration of topological change reflects genuine temporal structure rather than the total amount of movement, we compare the observed trajectories against a checkpoint-order permutation null. The early concentration statistic exceeds the null expectation in 11 of 12 model–objective combinations and remains significant in all but one case. Moreover, the same rise–peak–decay pattern is reproduced across independent fine-tuning seeds for TinyLlama, Gemma, and Mistral (Appendix A.4), indicating that the phenomenon is robust to initialization and data ordering. If the early peak merely reflected larger parameter updates early in optimization, one would expect all geometric summaries to track it identically. Instead, centroid drift, variance, and mean pairwise

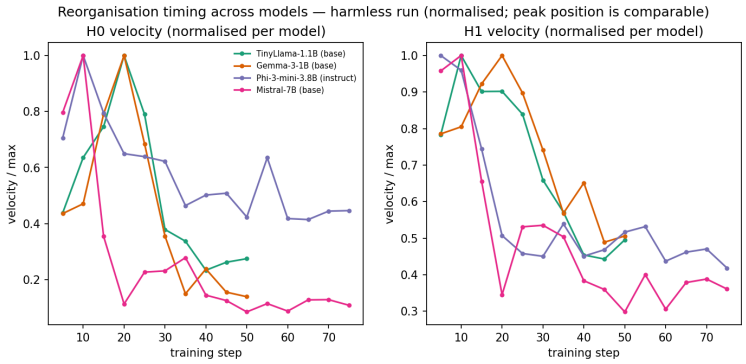

Figure 2: **The early transient topological peak (RQ1).** Dense early-window H_0 (left) and H_1 (right) Wasserstein velocity, normalized per model so the peak position is comparable (magnitudes are scale-dependent). The reorganization is a transient rise–peak–decay whose peak moves earlier with model size.

distance collapse to baseline after the burst while the H_1 Wasserstein velocity remains elevated (App. Figs. 10, 11), indicating sustained reorganization of the point cloud's shape beyond its scale and position.

For Gemma-3-1B, the H_1 velocity rises, peaks near step 20, and decays along nearly the same curve for every seed and objective (App. Fig. 21, left). The early concentration exceeds the checkpoint-order null in all nine seed-objective combinations. The early shock is therefore robust across scales, objectives, and seeds.

5.2. Objectives Leave Distinct, Oppositely-Signed Signatures (RQ2)

The three alignment objectives induce distinguishable topological trajectories. A small subset of H_1-persistence features carries most of the separation signal, with effect sizes reaching Hedges' $|g| \approx 1$–2 (Fig. 3, left). At the multivariate level, energy distances remain consistently positive across models, indicating persistent separation in the full topological summary vectors. Throughout, we report cell-level effect sizes with bootstrap confidence intervals rather than subsample-level significance tests.

The qualitative direction of topological change depends strongly on the starting state (Fig. 4, right). In the three base models, fine-tuning reduces H_0 persistence entropy relative to the initial checkpoint, with the harmless objective typically producing the strongest compression. In contrast, the instruction-tuned Phi-3 exhibits the opposite behavior, increasing H_0 persistence entropy above its initial value. Thus, initialization appears to play a stronger role than model scale in determining whether alignment induces topological compression or enrichment.

Within each model, however, the three objectives evolve along a largely shared axis. Pairwise displacement cosines remain strongly positive (median 0.79–0.94; Fig. 3, right), indicating that helpful, harmless, and mixed fine-tuning differ primarily in the magnitude rather than the direction of their topological reorganization, with harmless fine-tuning generally producing the largest displacement.

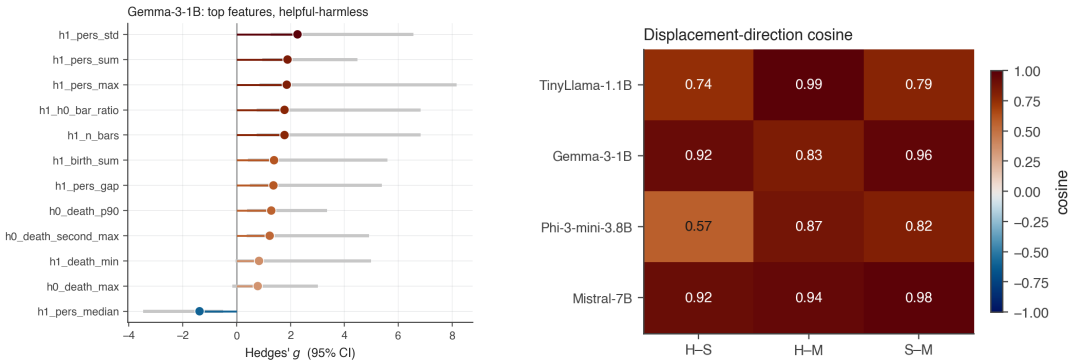

Figure 3: **Objective Separation (RQ2).** *Left:* Per-feature effect sizes (Hedges' g, 95% bootstrap CI) for the most-separated pair (Gemma); a few H_1-persistence features carry $|g| \sim 1$–2. *Right:* Objective displacement-direction cosines (model $\times$ pair) are all strongly positive; the objectives move the topology the same way, separating in magnitude.

Finally, objective separation emerges primarily at the endpoint of training. During the early transient reorganization phase, the three objectives remain nearly indistinguishable ($U = 23.6$, see (1), $p = 0.92$; App. Fig. 21), suggesting that the initial topological burst is largely shared across objectives. Additional token-pooling experiments indicate that the resulting objective-specific signatures are concentrated near the decision token (App. A).

5.3. Topology Reveals Changes Beyond Coarse Behavioral Metrics (RQ3)

Whether alignment becomes behaviorally visible depends strongly on the starting state rather than model scale. Across TinyLlama, Gemma, and Mistral, refusal rates on harmful prompts remain close to their $\approx$ 2–3% noise floor throughout fine-tuning and exhibit no clear phase transition (Fig. 4, left). In contrast, the instruction-tuned Phi-3 undergoes substantial behavioral change, with refusal rates increasing to 8–40% and separating by objective. These observations suggest that, under low-rank fine-tuning, pretrained models can undergo substantial representation-level reorganization without corresponding changes in coarse behavioral metrics.

A more sensitive behavioral measure based on Jensen–Shannon divergence between successive next-token distributions confirms that the model's output distribution does evolve during fine-tuning (Appendix C.1). However, topological velocity consistently peaks no later than behavioral velocity and in several cases slightly earlier (Appendix Fig. 8; Table 3). Together, these findings suggest that persistent homology captures representation-level changes that are only partially reflected in observable behavior.

5.4. Robustness and Geometric Controls

Several control analyses indicate that the observed topological phenomena do not reduce to simple geometric statistics. Relative to isotropic Gaussian point clouds matched in

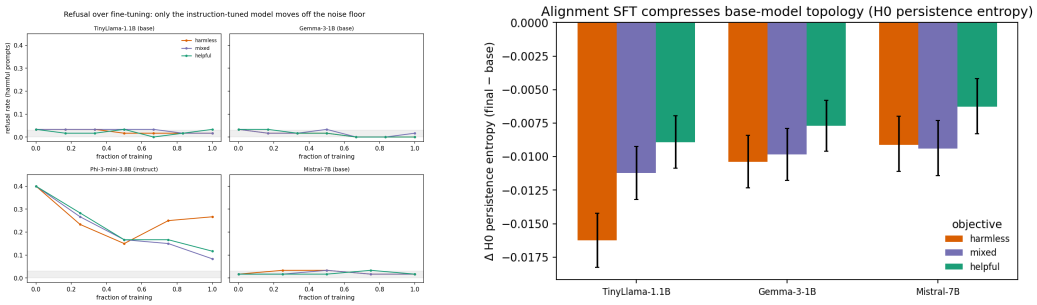

Figure 4: **Behavior and Objective Ordering (RQ3, RQ2).** *Left:* Refusal on harmful prompts over fine-tuning. All three base models stay on the ≈ 2–3% noise floor (shaded); only the instruction-tuned Phi-3 leaves it, reaching 8–40%, and there the objectives separate. *Right:* Base-to-final change in H_0 persistence entropy (headline layer, bootstrap 95% CI): base models compress (below zero, harmless most), Phi-3 enriches distinct, oppositely-signed signatures set by initialization.

dimension and sample size, the signal does not reside in individual scalar summaries such as raw H_0 entropy alone, but in the multivariate evolution of the topological descriptors.

Moreover, although centroid drift, total variance, and mean pairwise distance all detect the initial transient reorganization, these geometric statistics return rapidly to baseline, whereas the H_1 Wasserstein velocity remains elevated (App. Figs. 10, 11). PH thus tracks ongoing representation reorganization that is not captured by changes in the point cloud's centroid, scale, or overall spread. Additional controls are reported in Appendix A.1.

6. Conclusion and Future Directions

Across all models studied, alignment fine-tuning induces an early transient phase of topological reorganization whose timing varies with model scale. Different alignment objectives produce distinguishable trajectories, while their direction depends strongly on initialization. PH therefore provides a useful lens for studying representation dynamics during alignment beyond what is captured by coarse behavioral metrics.

Our results suggest several practical uses: First, the hidden-progress probe (App. Fig. 28) shows that the 41-dimensional topological summary identifies which fine-tuning run a checkpoint belongs to by step 50–100, even where refusal rates remain at the noise floor. Thus, topological summaries reveal representation-level changes during early low-rank fine-tuning that precede detectable behavioral changes, suggesting applications to alignment auditing. The rapid stabilization of persistence entropy (App. Fig. 28, right) further hints at a representation-level convergence criterion for checkpoint selection or early stopping.

Acknowledgments

K.F. and A.M. are supported by the EPSRC AI Hub on Mathematical Foundations of Intelligence: An "Erlangen Programme" for AI [EP/Y028872/1].

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, Shuohang Wang, Xin Wang, Yu Wang, Rachel Ward, Wen Wen, Philipp Witte, Haiping Wu, Xiaoxia Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Jilong Xue, Sonali Yadav, Fan Yang, Jianwei Yang, Yifan Yang, Ziyi Yang, Donghan Yu, Lu Yuan, Chenruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

Andy Arditi, Oscar Obeso, Aaqueel Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *arXiv preprint arXiv:2406.04093*, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Boaz Barak, Benjamin L. Edelman, Surbhi Goel, Sham Kakade, Eran Malach, and Cyril Zhang. Hidden progress in deep learning: SGD learns parities near the computational limit. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. arXiv:2207.08799.

Ulrich Bauer. Ripser: Efficient computation of Vietoris–Rips persistence barcodes. *Journal of Applied and Computational Topology*, 5(3):391–423, 2021.

Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46 (2):255–308, 2009.

Harish Chintakunta, Thanos Gentimis, Rocio Gonzalez-Diaz, Maria-Jose Jimenez, and Hamid Krim. An entropy-based persistence barcode. *Pattern Recognition*, 48(2):391–401, 2015.

David Cohen-Steiner, Herbert Edelsbrunner, John Harer, and Yuriy Mileyko. Lipschitz functions have L_p-stable persistence. *Foundations of Computational Mathematics*, 10(2): 127–139, 2010.

Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4):511–533, 2002.

Aideen Fay, Inés García-Redondo, Qiquan Wang, Haim Dubossarsky, and Anthea Monod. The shape of adversarial influence: Characterizing LLM latent spaces with persistent homology. In *International Conference on Learning Representations (ICLR)*, 2026.

Yuri Gardinazzi, Karthik Viswanathan, Giada Panerai, Alessio Ansuini, Alberto Cazzaniga, and Matteo Biagetti. Persistent topological features in large language models. In *International Conference on Machine Learning (ICML)*, 2025.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Samyak Jain, Robert Kirk, Ekdeep Singh Lubana, Robert P. Dick, Hidenori Tanaka, Edward Grefenstette, Tim Rocktäschel, and David Scott Krueger. Mechanistically analyzing the effects of fine-tuning on procedurally defined tasks. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2311.12786.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Michael Kerber, Dmitriy Morozov, and Arnur Nigmetov. Geometry helps to compare persistence diagrams. *ACM Journal of Experimental Algorithmics*, 22:1–20, 2017.

Solomon Kullback and Richard A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

Andrew Lee, Xiaoyan Bai, Itamar Pres, Martin Wattenberg, Jonathan K. Kummerfeld, and Rada Mihalcea. A mechanistic understanding of alignment algorithms: A case study on DPO and toxicity. In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2401.01967.

Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.

Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Anna Tacchetti, Andrea amd Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L. Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu-hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. Gemma: Open models based on Gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

Yuriy Mileyko, Sayan Mukherjee, and John Harer. Probability measures on the space of persistence diagrams. *Inverse Problems*, 27(12):124007, 2011.

Michael Moor, Max Horn, Bastian Rieck, and Karsten Borgwardt. Topological autoencoders. In *International Conference on Machine Learning (ICML)*, 2020.

Gregory Naitzat, Andrey Zhitnikov, and Lek-Heng Lim. Topology of deep neural networks. *Journal of Machine Learning Research*, 21(184):1–40, 2020.

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

Nina Otter, Mason A. Porter, Ulrike Tillmann, Peter Grindrod, and Heather A. Harrington. A roadmap for the computation of persistent homology. *EPJ Data Science*, 6(1):17, 2017.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Bastian Rieck, Matteo Togninalli, Christian Bock, Michael Moor, Max Horn, Thomas Gumbsch, and Karsten Borgwardt. Neural persistence: A complexity measure for deep neural networks using algebraic topology. In *International Conference on Learning Representations (ICLR)*, 2019.

Gábor J Székely and Maria L Rizzo. Energy statistics: A class of statistics based on distances. *Journal of Statistical Planning and Inference*, 143(8):1249–1272, 2013.

Tang, Wang, Inés García-Redondo, and Anthea Monod. Topological signatures of grokking. *arXiv preprint arXiv:2605.06352*, 2026.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. TinyLlama: An open-source small language model. *arXiv preprint arXiv:2401.02385*, 2024.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. LIMA: Less is more for alignment. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2305.11206.

Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33(2):249–274, 2005.

Appendix A. Supplementary Results

A.1. Robustness Checks and Controls

We performed a series of controls to verify that the observed topological phenomena are not artifacts of checkpoint ordering, simple geometric statistics, token selection, or the use of overlapping subsamples.

Checkpoint-order null. Permuting checkpoint order and recomputing the H_1 Wasserstein velocity destroys the early concentration of topological movement. Under the observed checkpoint ordering, the velocity is strongly concentrated within the first third of training, whereas random checkpoint permutations distribute the same total movement approximately uniformly across time (Figs. 6–7, Table 1).

Geometric baselines. We compare persistent homology with conventional geometric statistics computed on the identical activation clouds, namely centroid drift, total variance, and mean pairwise distance. All measures exhibit the initial transient burst. However, the geometric statistics rapidly return to baseline, whereas the H_1 Wasserstein velocity remains elevated (Figs. 10 and 11), indicating that persistent homology is not redundant with first- and second-order geometric moments.

Isotropic-Gaussian floor. An isotropic Gaussian point cloud matched in sample size and ambient dimension produces approximately 238 artifactual H_1 bars and an H_0 persistence entropy of approximately 5.07, close to the scalar entropy values observed after fine-tuning. This motivates the multivariate topological summaries used in §5.4.

Token-level pooling. Pooling the final 16 token representations substantially weakens objective separation (Fig. 26), localizing the discriminative structure to the decision token.

Subsample pseudoreplication. Treating the 64 overlapping subsamples of an activation cloud as independent observations produces highly inflated significance levels (Fig. 13). Consequently, all confirmatory inference in this work is performed at the cell level and reported using effect sizes.

Model	Significant (p<0.05)	p range	Peak position
TinyLlama-1.1B	3/3	0.002–0.003	first third
Gemma-3-1B	2/3	0.024–0.12	first third
Phi-3-mini-3.8B	3/3	0.008–0.010	earliest
Mistral-7B	3/3	≈ 0.001	earliest

Table 1: Early-shock checkpoint-order null per model across the three objectives (H_1 Wasserstein velocity). The early velocity concentration beats the time-shuffled null in 11/12 cells. See `results/metrics/exp2_cell/shuffle_null.csv` for the full per-cell values.

Model	Start	U (standardized)	Single-run U-test	Energy dist. range
TinyLlama-1.1B	base	2.33	degenerate (p undef.)	0.31–0.55
Gemma-3-1B	base	73.7	degenerate (p undef.)	1.02–1.35
Phi-3-mini-3.8B	instruct	12.0	degenerate (p undef.)	—
Mistral-7B	base	0.44	degenerate (p undef.)	—

Table 2: Cell-level objective separation. The all-pairs U (1) is invariant to relabeling with one trajectory per objective, so the single-run permutation test is degenerate for every model.

Model	Objective	Topo. peak	Func. peak	Lead (steps)
Gemma-3-1B	helpful	15	20	+5
Gemma-3-1B	harmless	20	20	0
Gemma-3-1B	mixed	20	20	0
Phi-3-mini	helpful	10	15	+5
Phi-3-mini	harmless	5	10	+5
Phi-3-mini	mixed	5	5	0
Mistral-7B	helpful	5	10	+5
Mistral-7B	harmless	10	10	0
Mistral-7B	mixed	5	10	+5

Table 3: Velocity-peak lead-lag (§5.3). Lead = function-peak − topology-peak step. Topology never lags, leading by at most one checkpoint. Source: `results/metrics/exp2_cell/leadlag.csv`.

A.2. Objective Separation

Table 2 reports the standardized objective-separation statistic U (1) together with the associated single-run degeneracy. Because each objective contributes only a single trajectory, permuting objective labels leaves the pairwise trajectory dispersion invariant and therefore yields a degenerate permutation test. This motivates the seed-replicated analysis of §A.4.

Cell-level energy distances are shown in Fig. 12, while per-feature effect sizes and objective displacement-direction cosines are shown in Figs. 14 and 15, respectively.

A.3. Early Reorganization Across Models

Figure 16 shows the z-scored feature-by-step reorganization maps for all four models. In every case, the dominant reorganization is concentrated in an early high-activity band. As model scale increases, this band shifts progressively earlier in training.

Figure 17 summarizes the peak step of each of the 41 barcode features across models. Figure 18 shows the corresponding per-model heatmaps at their native resolutions. The associated dense-window velocity trajectories are given in Fig. 2 of the main text.

A.4. Seed Replication

To assess robustness, we replicated the dense early-window experiments using three independent seeds, varying both LoRA initialization and data order.

Across TinyLlama-1.1B, Gemma-3-1B, and Mistral-7B, the early-window H_1 velocity follows nearly identical rise–peak–decay trajectories across seeds and objectives (Figs. 19, 21, and 23).

The seed-replicated objective-separation test of (1) becomes non-degenerate once multiple trajectories per objective are available. For all three models, the observed statistic lies comfortably within its trajectory-permutation null:

- TinyLlama-1.1B: $U = 0.74$, $p = 0.998$;
- Gemma-3-1B: $U = 23.6$, $p = 0.92$;
- Mistral-7B: $U = 0.76$, $p = 0.93$.

Thus, the alignment objectives are statistically indistinguishable during the initial transient burst. The additional seeds also reproduce the original reorganization maps (Figs. 20, 22, and 24), demonstrating that the early transient phenomenon is robust to both LoRA initialization and data ordering.

Absolute values of U are not directly comparable across models, since each model induces its own standardized feature scale.

A.5. Additional Analyses

Figure 5 shows representative persistence diagrams and barcodes for Gemma-3-1B under the harmless objective. Figure 25 illustrates persistence landscapes and their associated bootstrap summaries.

Table 1 reports the checkpoint-order permutation null for all model-objective combinations. The observed early concentration exceeds the shuffled null in 11/12 cells, with only the helpful run of Gemma-3-1B failing to reach significance.

Table 3 reports the peak-step comparison between topological and functional velocities. Topology never lags function and leads by at most one dense-window checkpoint.

Figure 26 shows that token-level pooling largely eliminates objective separation, localizing the discriminative signal to the decision token.

Figure 27 gives topology-versus-refusal overlays for representative base and instruction-tuned models.

Finally, Fig. 28 presents the hidden-progress probe and saturating-exponential fit to H_0 persistence entropy, while Fig. 29 shows the exploratory subsample-level landscape-permutation heatmap discussed in §5.4.

Appendix B. Compute and Implementation Details

B.1. Computational Environment

Hardware. Experiments were conducted on a single Apple M3 Max workstation with a 16-core CPU, integrated 40-core GPU, and 64 GB unified memory running macOS 26. Model training and extraction use the Metal Performance Shaders (MPS) backend through PyTorch with `bfloat16` precision and scaled-dot-product attention (`sdpa`). Device selection follows the hierarchy `mps`> `cuda` > `cpu`, allowing the same codebase to execute unchanged on CUDA-enabled systems.

Software. The training and extraction pipeline uses PyTorch, Transformers, PEFT, Accelerate, and Datasets. Persistent homology computations are performed using `ripser`, while persistence distances are computed with `persim`. Statistical analyses and visualizations use NumPy, SciPy, pandas, scikit-learn, and Matplotlib. The topology and statistics modules are deliberately implemented independently of deep-learning frameworks.

Reproducibility. The software environment is managed using `uv` (Python 3.11). Every run records the random seed, model revision, and software versions alongside the extracted activations. Persistence diagrams are cached per (run, condition, layer) tuple, allowing all analyses to be reproduced directly from stored activations and diagrams without repeating model training or extraction. Code, configurations, and anonymized software accompany the submission.

B.2. Models and Fine-Tuning

We study four open-weight decoder-only language models spanning approximately 1B–7B parameters:

- TinyLlama-1.1B (`TinyLlama/TinyLlama-1.1B-intermediate-step-1431k-3T`),
- Gemma-3-1B (`google/gemma-3-1b-pt`),
- Phi-3-mini-3.8B (`microsoft/Phi-3-mini-4k-instruct`), and
- Mistral-7B (`mistralai/Mistral-7B-v0.1`).

Phi-3-mini-3.8B is instruction-tuned, while the remaining models are pretrained base models.

Each model is fine-tuned by supervised fine-tuning on the preferred (*chosen*) responses of the helpful, harmless, and mixed HH-RLHF subsets (Bai et al., 2022). We use LoRA with rank $r = 8$, $\alpha = 16$, dropout 0.05, and `task_type=CAUSAL_LM`, applied to the attention projections. Optimization uses a learning rate of 2×10^{-4}, micro-batches of size 4, gradient accumulation 8 (effective batch size 32), and a maximum sequence length of 256. All objectives use identical hyperparameters and random seeds.

The 1B models are trained for 300 steps on 3000 examples with checkpoints every 50 steps. The larger models are trained for 400 steps on 6000 examples with checkpoints every 100 steps. To resolve the earliest stages of alignment, we additionally perform a dense early-window analysis that checkpoints every 5 steps up to step 50 while keeping the same optimization schedule.

B.3. Activation Extraction and Persistent Homology

At each checkpoint, we evaluate the model on a fixed evaluation set consisting of 250 prompts per condition and extract the final-token hidden state at five evenly spaced transformer layers. This yields one activation point cloud of $n = 250$ points for each (condition, layer) pair.

For each cloud, we draw $B = 64$ overlapping subsamples of size $m = 160$ and compute Vietoris–Rips PH up to dimension one using the Euclidean distance. PH is computed with `ripser`, and topological velocities are measured using order-2 Wasserstein distances between consecutive-checkpoint persistence diagrams via `persim`. Unless otherwise stated, analyses are performed on the resulting barcode summaries rather than on the raw diagrams.

The behavioral analyses use the same saved checkpoints and evaluation prompts. Refusal rates and response lengths are computed on a fixed 60-prompt subset with at most 64 generated tokens.

B.4. The 41-Dimensional Barcode Summary

Each persistence diagram is reduced to a fixed-length vector $\mathbf{z} \in \mathbb{R}^{41}$ using the descriptor family of Fay et al. (2026). All statistics are computed on finite bars only, with the essential infinite H_0 bar removed.

The summary consists of 19 statistics describing H_0, 21 statistics describing H_1, and one cross-degree feature:

- H_0 **(19):** number of bars, summary statistics of component death times, persistence entropy, and salience measures based on the largest and second-largest deaths;
- H_1 **(21):** number of bars together with summary statistics of loop birth times, death times, and persistences, persistence entropy, salience measures, and the mean birth/death ratio;
- **Cross (1):** the ratio n_{H_1}/n_{H_0}.

Persistence entropy is defined by

$$-\sum_i p_i \log p_i, \qquad p_i = \frac{\ell_i}{\sum_j \ell_j},$$

where ℓ_i denotes the length of the ith bar. Together, these descriptors summarize changes in component mergers, loop structure, and persistence distributions in a form that is directly comparable across checkpoints and objectives.

Appendix C. Additional Statistical Details and Robustness Checks

C.1. Statistical Inference and Robustness

Cell-level inference. Each activation cloud is represented by $B = 64$ overlapping subsamples. These subsamples provide an estimate of variability arising from finite prompt sets but do not constitute independent observations. Consequently, all confirmatory statistical inference is performed at the cell level, where a cell corresponds to a fixed combination of model, objective, evaluation condition, layer, and checkpoint. Subsample-level analyses are used only for exploratory visualizations, such as PCA projections and persistence-landscape maps.

Effect sizes and confidence intervals. Our primary measures of difference are per-feature effect sizes (Hedges' g and Cliff's δ) and the multivariate energy distance. Ridge-regularized Mahalanobis distances were computed as a cross-check and yielded qualitatively similar conclusions. All effect sizes are reported with percentile bootstrap 95% confidence intervals obtained from at least 2000 bootstrap resamples of the appropriate experimental unit.

Trajectory-level permutation testing. The objective-separation statistic U is assessed by permuting whole trajectories rather than individual subsamples. With only one trajectory per objective, permutation testing is degenerate because the all-pairs trajectory dispersion is invariant under relabeling. We therefore perform three-seed replications for TinyLlama-1.1B, Gemma-3-1B, and Mistral-7B, producing nine trajectories per model.

Under the null hypothesis that objective labels are exchangeable, we consider all size-preserving assignments of these nine trajectories into three groups of size three. The number of such assignments is

$$\frac{9!}{(3!)^3} = 1680,$$

yielding an exact permutation test with minimum attainable p-value

$$p_{\min} = \frac{1}{1681} \approx 5.95 \times 10^{-4},$$

using the Phipson–Smyth correction

$$p = \frac{1 + \#\{T_{\text{perm}} \geq T_{\text{obs}}\}}{1 + N_{\text{perm}}}.$$

Checkpoint-order null and behavioral velocity. To assess whether topological reorganization is genuinely concentrated early in training, we randomly permute checkpoint order and recompute both the topological velocity and early-concentration statistic. Behavioral velocity is defined analogously using the Jensen–Shannon divergence between consecutive next-token distributions:

$$\beta_t = \overline{\mathrm{JS}(P_t(\cdot|x), P_{t-1}(\cdot|x))}$$

where the overline denotes averaging over evaluation prompts. Jensen–Shannon divergence is used because it is symmetric, bounded, and remains finite even when consecutive checkpoints assign probability mass to disjoint sets of tokens.

Additional robustness analyses. We additionally compare PH summaries with conventional geometric statistics computed on the same activation point clouds, namely centroid drift, total variance, and mean pairwise distance. We further compare the observed signatures with isotropic Gaussian point clouds matched in sample size and ambient dimension and reconstruct activation clouds using the final sixteen token states of each prompt to assess sensitivity to token selection. These analyses are reported above in Appendix A.

Multiple comparisons. For exploratory subsample-level landscape comparisons, we control the family-wise error rate using the Holm–Bonferroni procedure. Because the corresponding tests treat heavily overlapping subsamples as independent observations, these corrected p-values are regarded as exploratory only. Throughout the paper, we therefore use effect sizes for confirmatory inference and reserve p-values primarily for null and control tests.

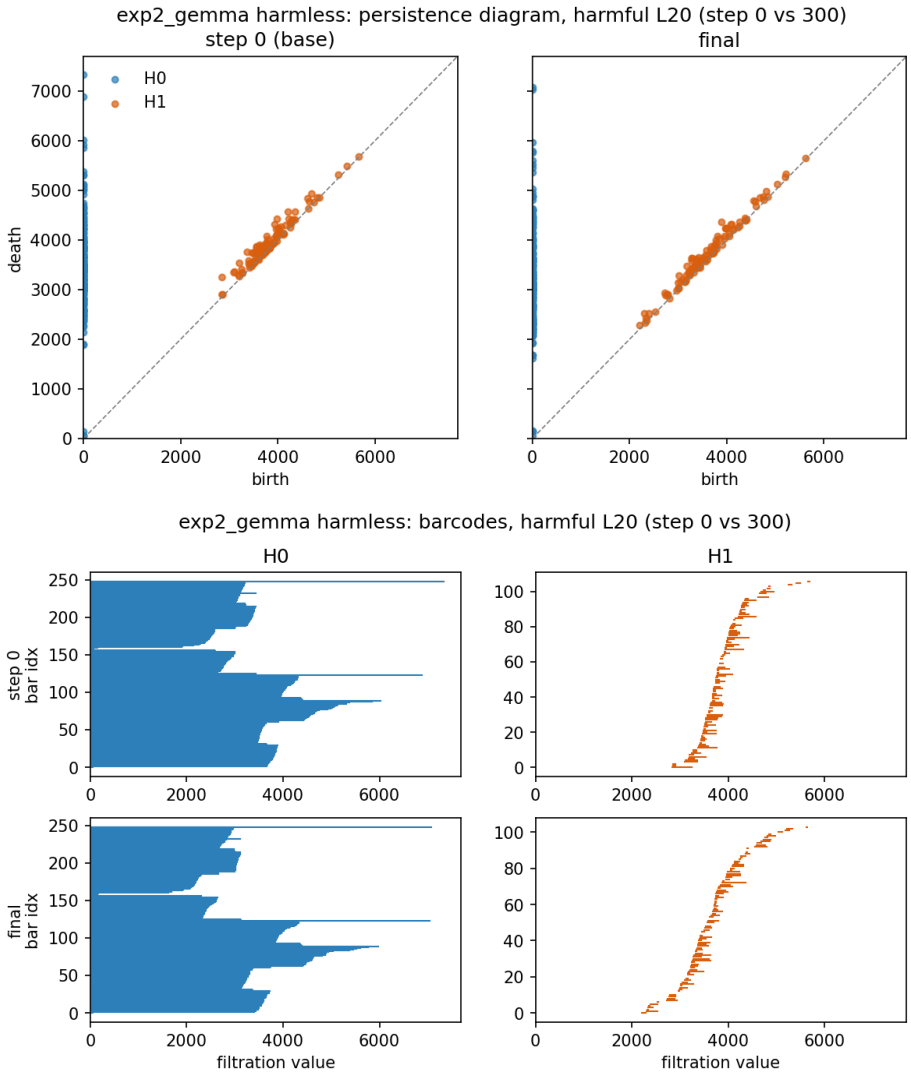

Figure 5: Gemma-3-1B persistence diagram (top) and barcode (bottom), harmless run, layer 20. Fine-tuning mildly shortens and thins the bars.

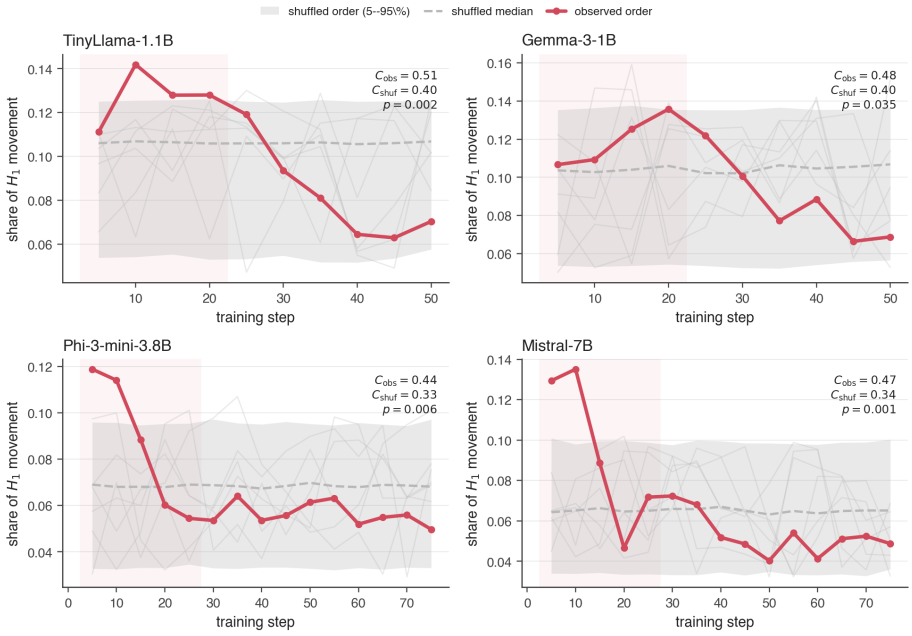

Figure 6: **Permuting the checkpoint order removes the early peak.** For each dense model (harmless run), the share of total H_1 Wasserstein movement at each step under the observed checkpoint order (red, concentrated in the shaded first-third window) versus 1000 random checkpoint orders (grey median, 5–95% band, and a few individual traces).

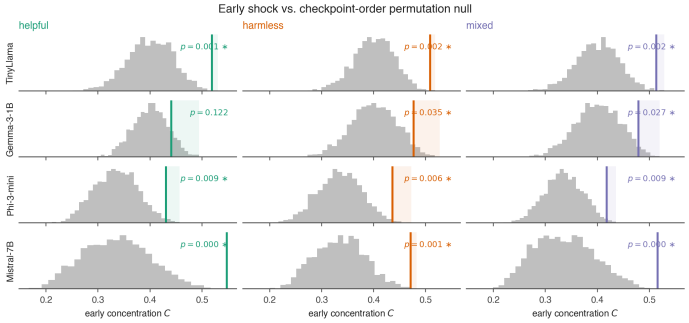

Figure 7: **The early shock against a checkpoint-order permutation null** (§5.1). For each (model, objective), the null distribution of the early-velocity concentration C (H_1 Wasserstein) under randomly shuffled checkpoint order (grey, 2000 permutations), with the observed C as a colored line (shaded tail) and its permutation p; $*\,p{<}0.05$.

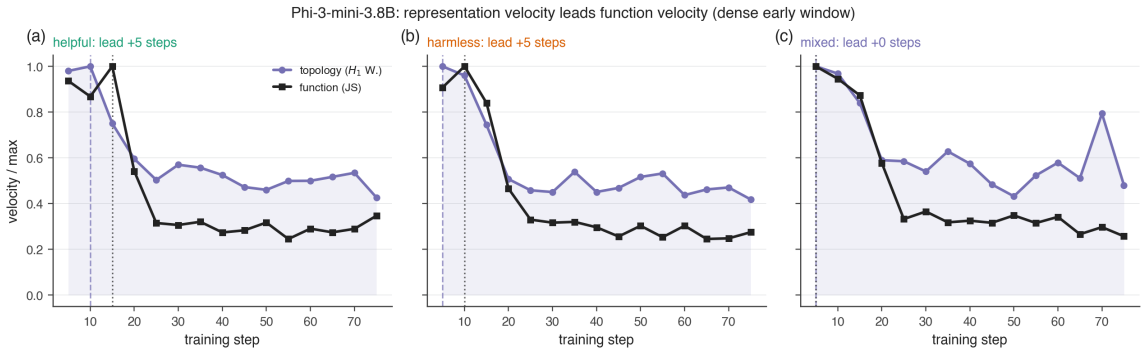

Figure 8: **Representation leads function (Phi-3, dense).** H_1 Wasserstein velocity (topology, purple) and JS velocity of the output distribution (function, black) on a shared checkpoint axis. Both peak in the first 5-15 steps with topology marginally first (dashed vs. dotted markers). Topology then retains a longer tail. Phi-3 is the model whose refusal genuinely moves, so this is the sharpest available timing test.

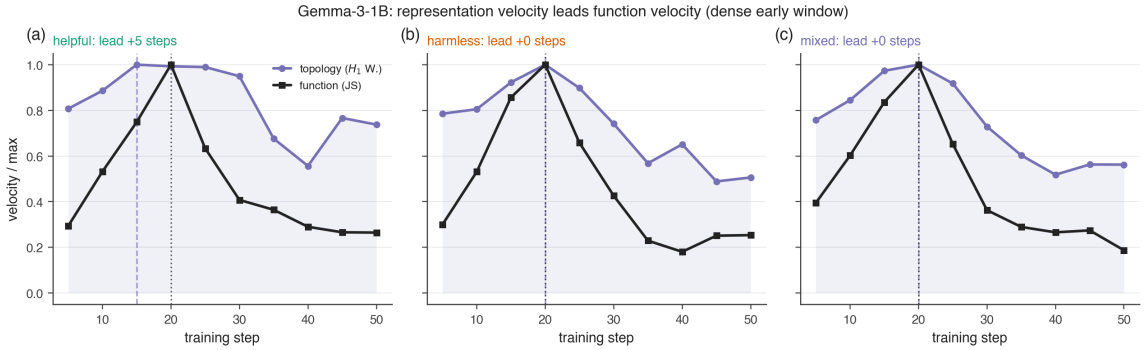

Figure 9: Gemma-3-1B representation (H_1 Wasserstein) vs. function (JS) velocity companion to Fig. 8. Both peak near step 20. Topology leads by at most one checkpoint.

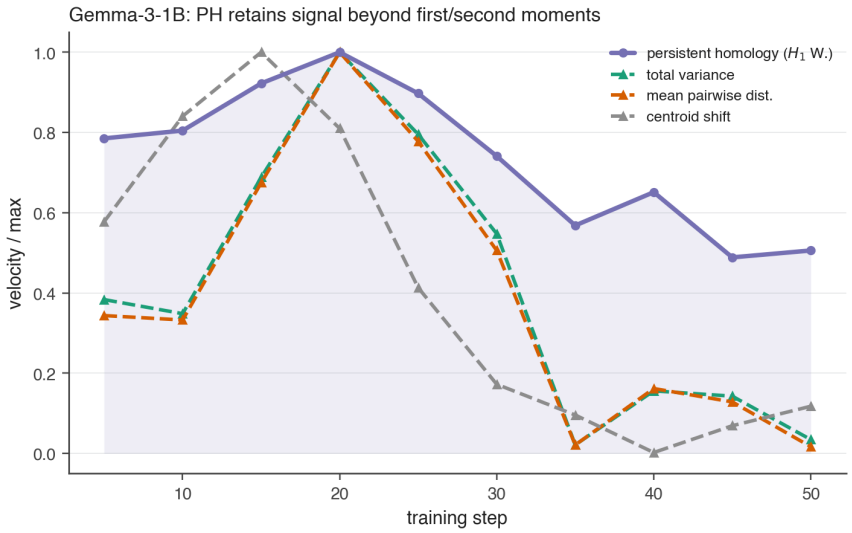

Figure 10: **PH vs. geometry (Gemma, harmless, dense).** Normalized velocity of the H_1 Wasserstein metric against non-topological baselines on the same clouds (§5.4). All rise in the early burst, but centroid drift and variance collapse to floor by step $\sim$35 while PH stays elevated; PH is therefore not redundant with first/second moments.

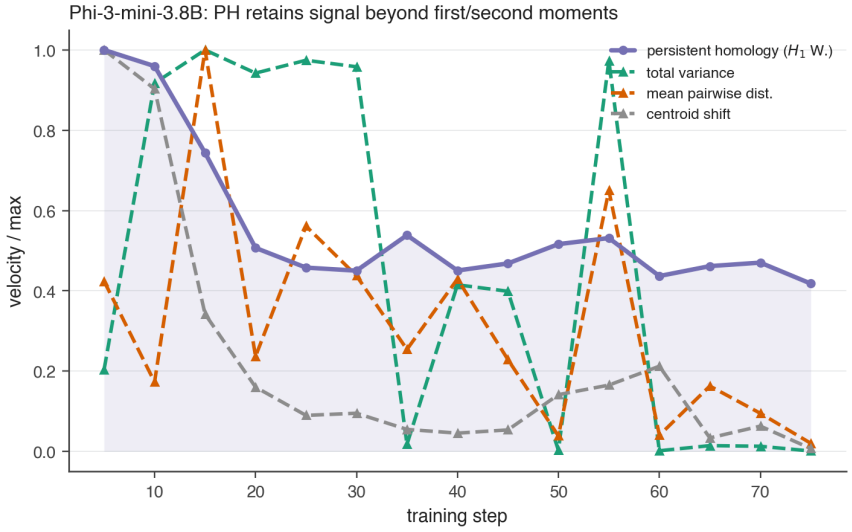

Figure 11: Phi-3 geometric-baseline ablation (companion to Fig. 10): the H_1 Wasserstein velocity retains a longer-lived signal than centroid drift/variance/mean pairwise distance, which collapse after the early burst.

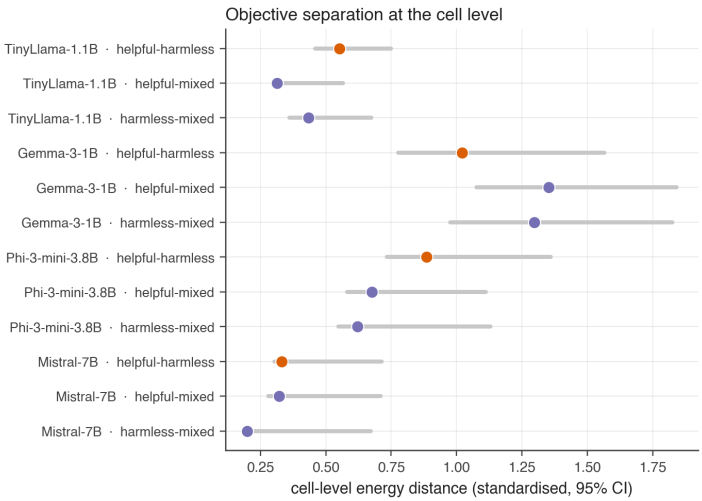

Figure 12: Cell-level energy distance between objectives per pair, per model (standardized, 95% bootstrap CI). Cell-level energy distances are modest and accompanied by wide intervals, illustrating the more conservative, de-biased view of objective separation.

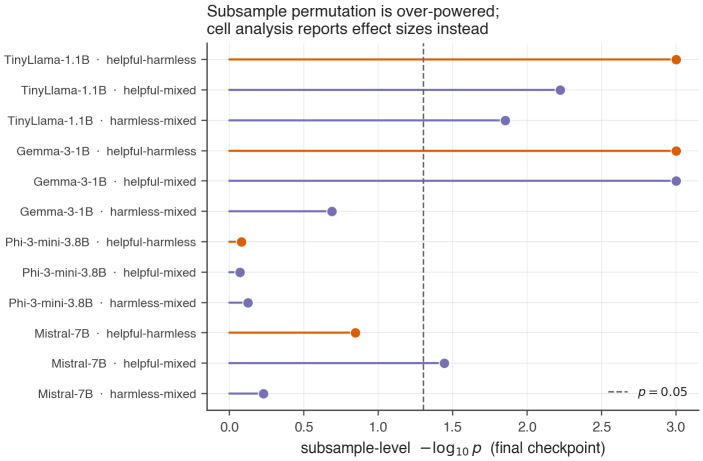

Figure 13: Pseudoreplication inflates significance. Subsample-level permutation $-\log_{10} p$ at the final checkpoint, per (model, pair). Treating 64 overlapping subsamples as independent drives p far below 0.05; the cell-level analysis reports effect sizes (Fig. 12, Fig. 3) instead.

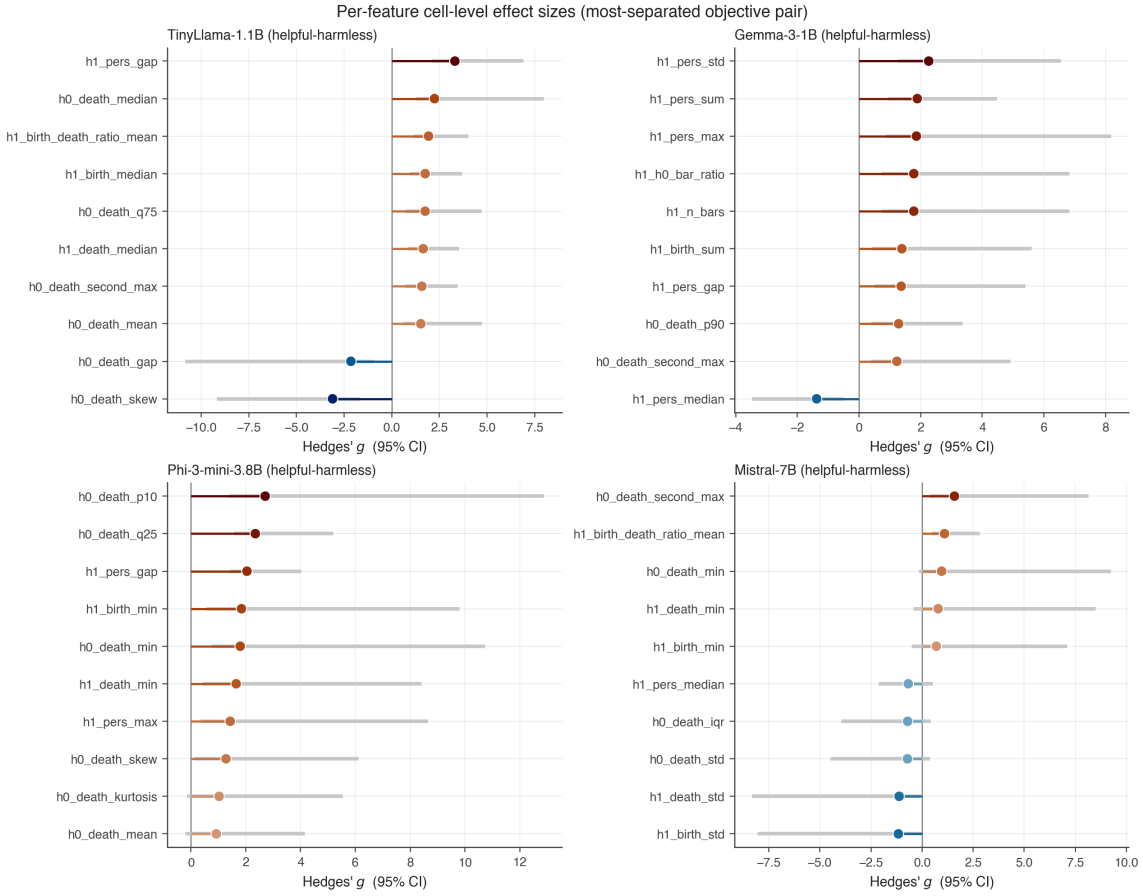

Figure 14: Per-feature cell-level effect sizes (Hedges' g, 95% CI) for the most-separated objective pair of each model. This is the full version of Fig. 3 (left).

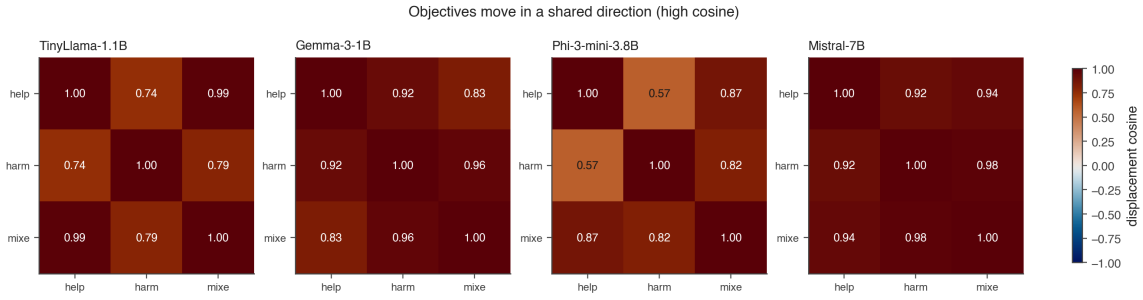

Figure 15: Per-model objective displacement-direction cosine matrices; every off-diagonal entry is strongly positive. This is a companion to Fig. 3, right.

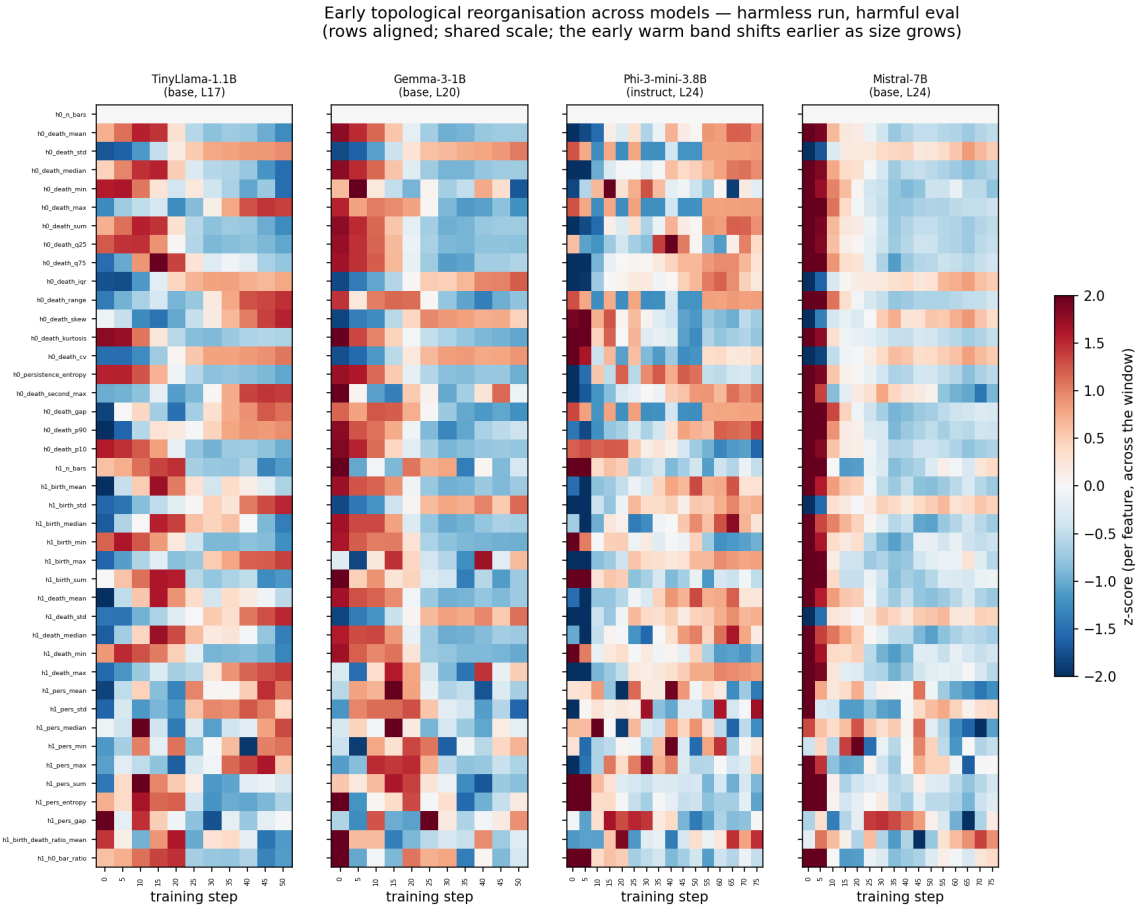

Figure 16: Per-feature reorganisation across the four models (harmless run, harmful eval), each column z-scored across its own window with one shared diverging scale and identical 41-feature row order.

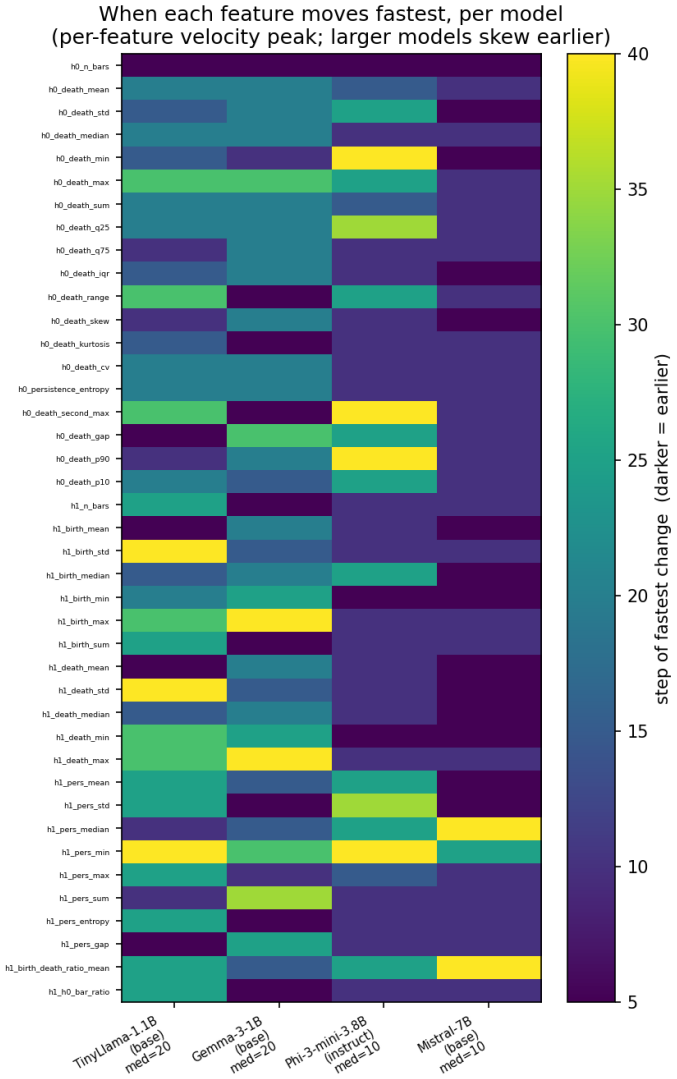

Figure 17: Dense early-window per-feature peak step (41 features × 4 models): larger models skew earlier, the per-feature companion to the velocity burst of Fig. 2.

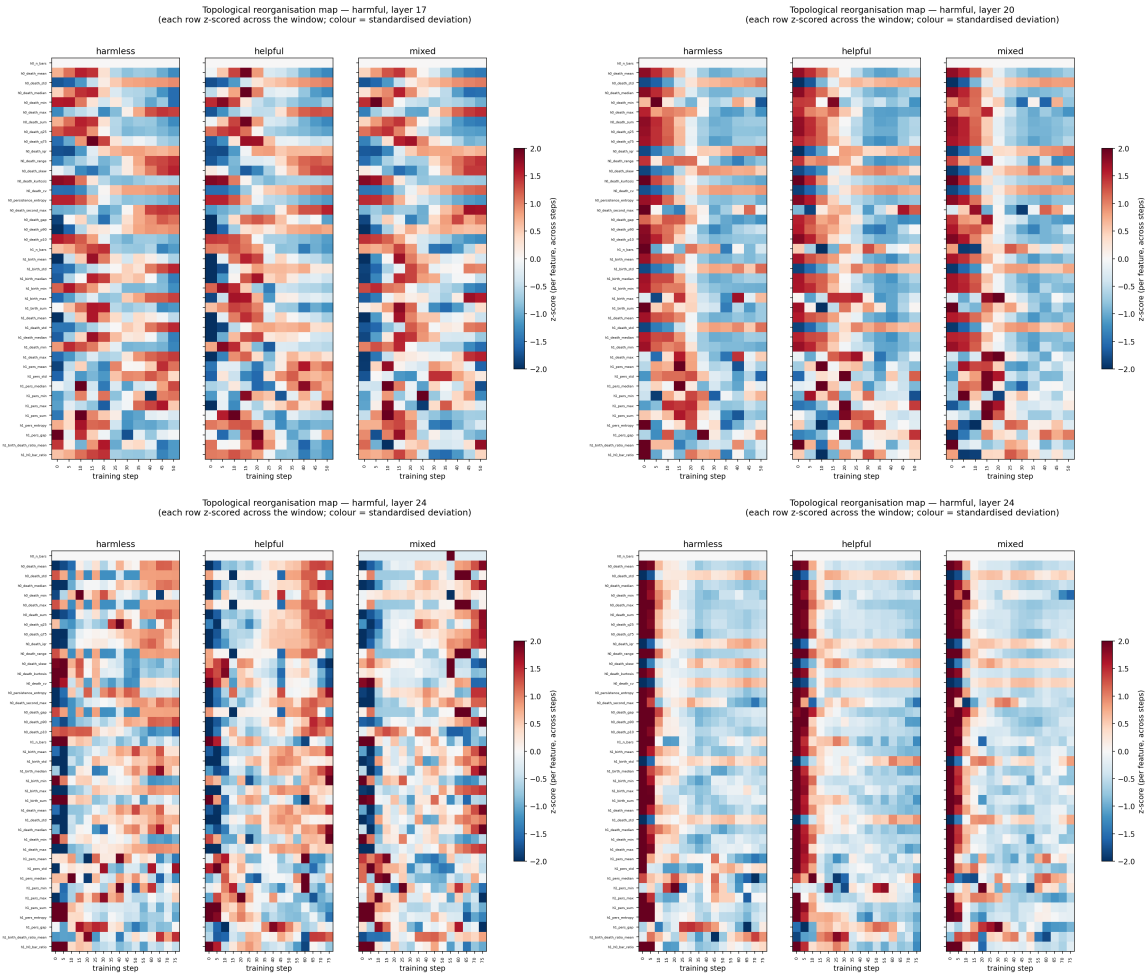

Figure 18: Per-model dense reorganisation maps (harmless run): TinyLlama-1.1B (top left), Gemma-3-1B (top right), Phi-3-mini-3.8B (bottom left), Mistral-7B (bottom right). The early high-activity band concentrates and moves earlier as model size grows.

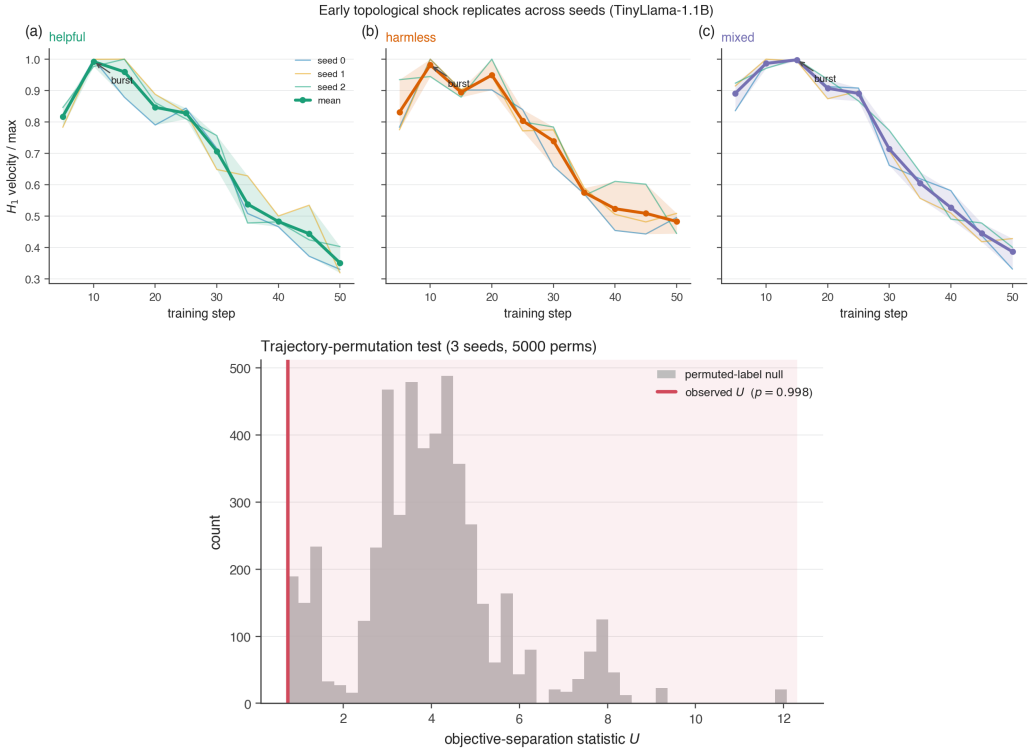

Figure 19: **Seed replication (TinyLlama-1.1B).** *Top:* early-window H_1 velocity overlaid across three independent seeds the burst follows nearly the same curve for every objective and seed. *Bottom:* the observed objective-separation U falls within the trajectory-permutation null, so objectives are not separated during the burst.

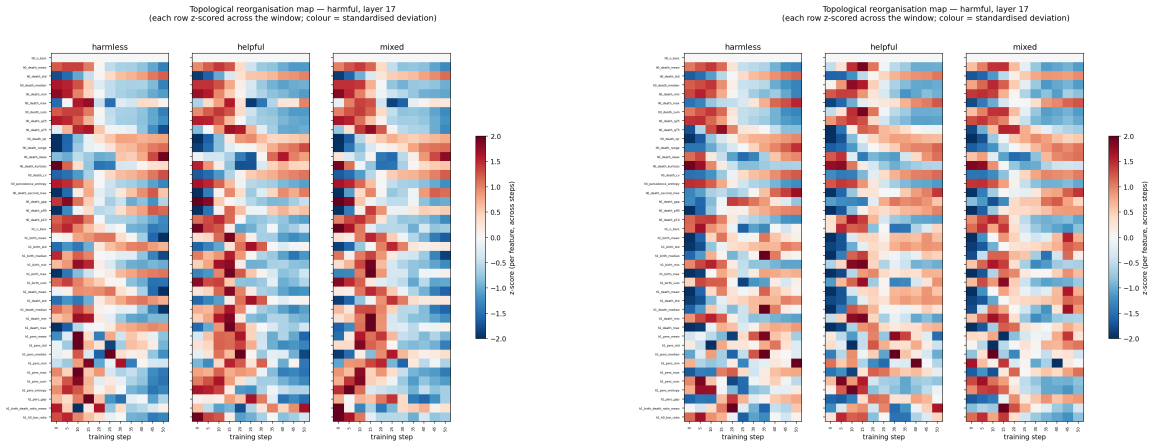

Figure 20: Reorganization maps for the two added TinyLlama seeds (harmless objective), matching the early-window structure of the original seed.

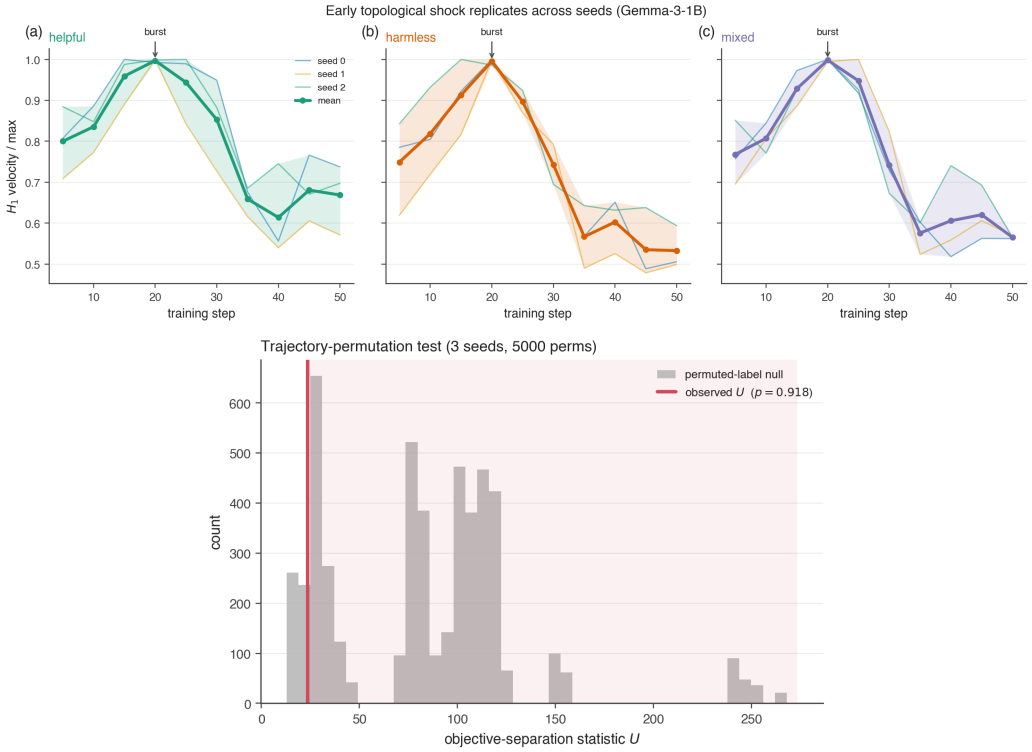

Figure 21: **Seed replication (Gemma-3-1B).** *Top:* early-window H_1 velocity overlaid across three independent seeds the burst follows nearly the same curve for every objective and seed. *Bottom:* the observed objective-separation U falls within the trajectory-permutation null, so objectives are not separated during the burst.

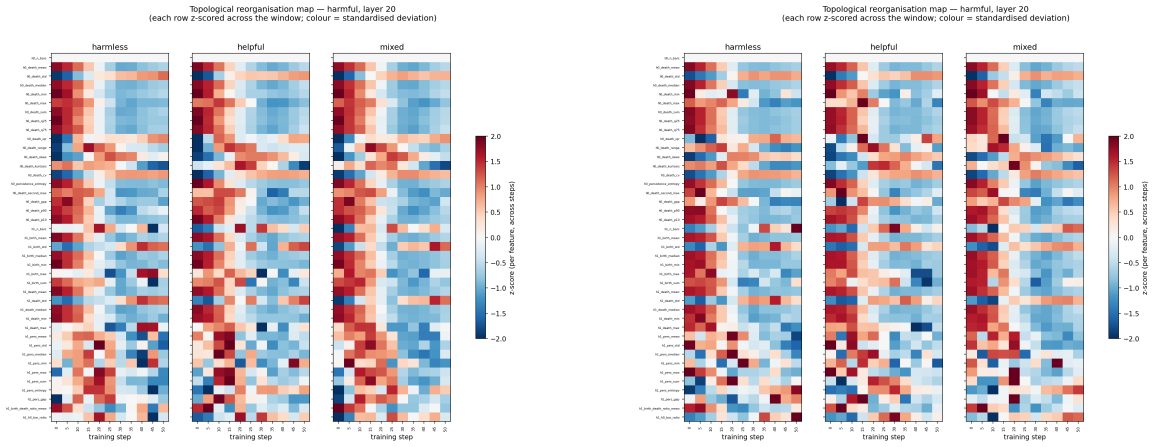

Figure 22: Reorganization maps for the two added Gemma seeds (harmless objective), reproducing the structure of the original run.

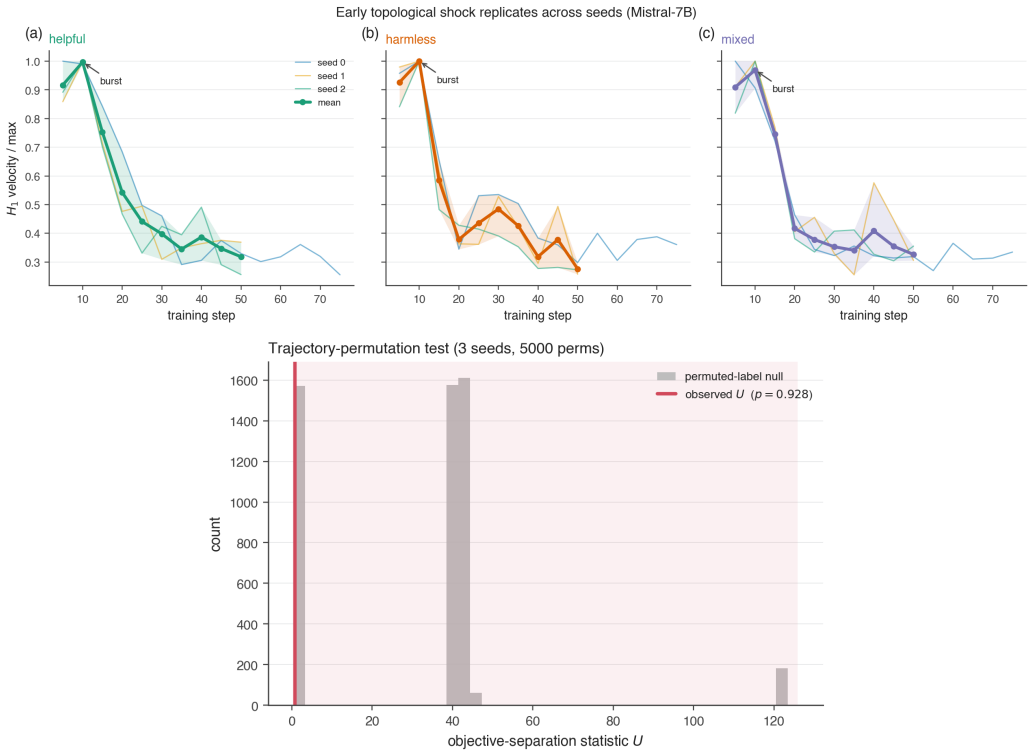

Figure 23: **Seed replication (Mistral-7B).** *Top:* early-window H_1 velocity overlaid across three independent seeds the burst follows nearly the same curve for every objective and seed. *Bottom:* the observed objective-separation U falls within the trajectory-permutation null, so objectives are not separated during the burst.

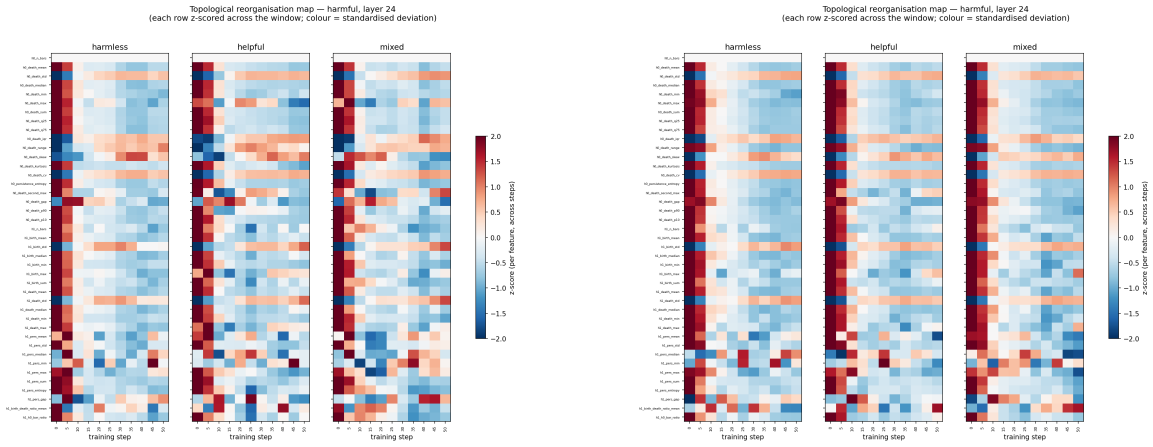

Figure 24: Reorganization maps for the two added Mistral seeds (harmless objective), matching the early-window structure of the original run.

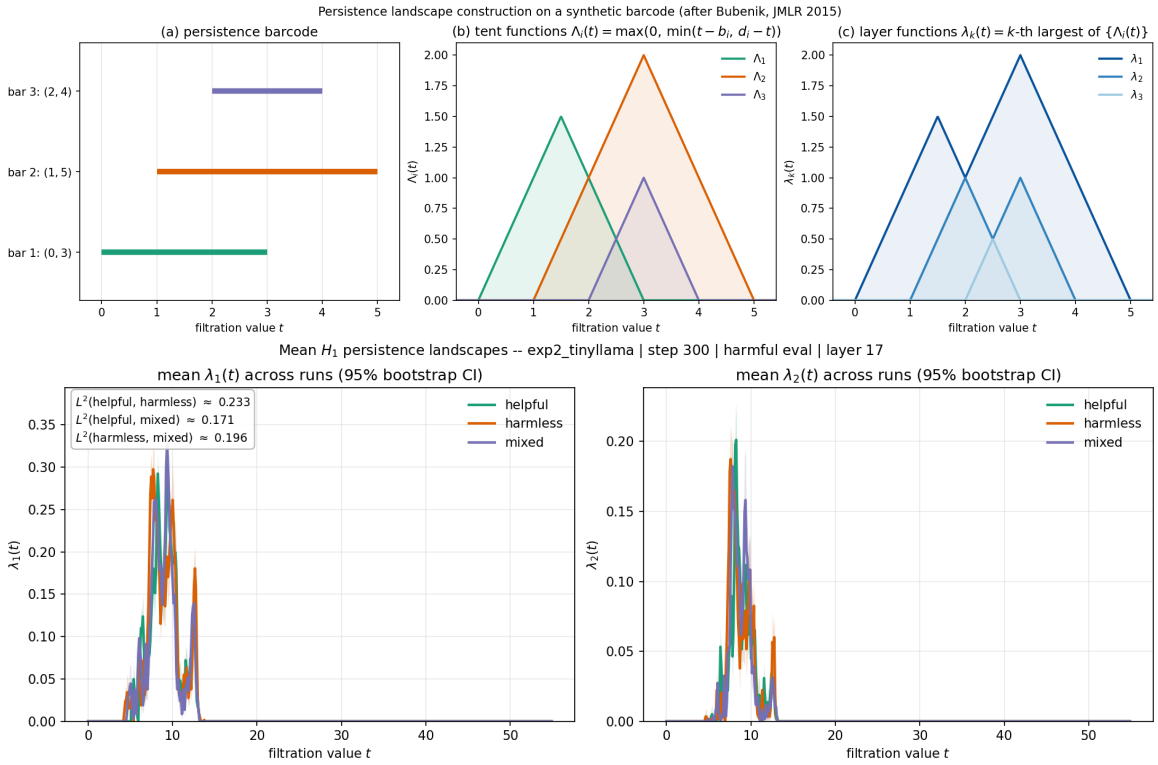

Figure 25: Persistence landscapes: pedagogical construction (top) and the three objectives' mean H_k landscapes with 95% bootstrap bands and pairwise L^2 distances (bottom).

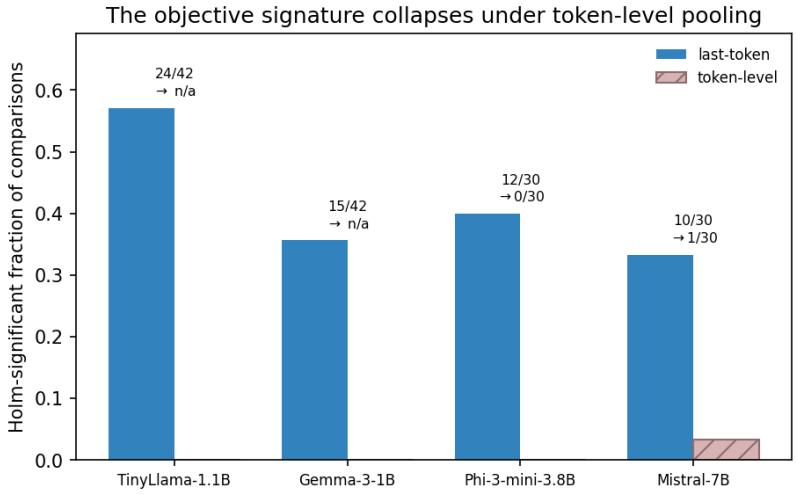

Figure 26: Holm-significant fraction of landscape comparisons, last-token vs. token-level pooling (raw counts annotated). Pooling the last 16 tokens collapses the objective signal (Phi-3 $12/30 \to 0/30$, Mistral $10/30 \to 1/30$), so the discriminating structure sits at the decision token. (TinyLlama and Gemma have no token-level variant.)

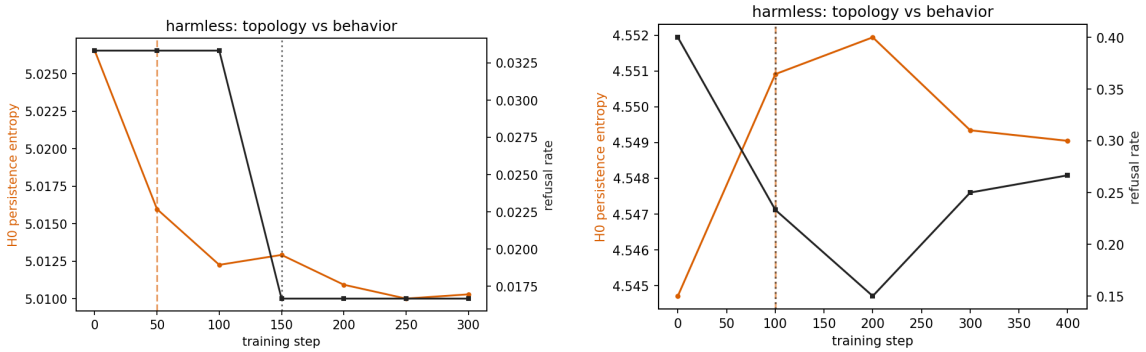

Figure 27: Topology (left axis) vs. refusal rate (right axis) over training, harmless run: TinyLlama (base, left) and Phi-3 (instruct, right). Refusal is pinned at the noise floor for the base model and moves only for the instruction-tuned Phi-3.

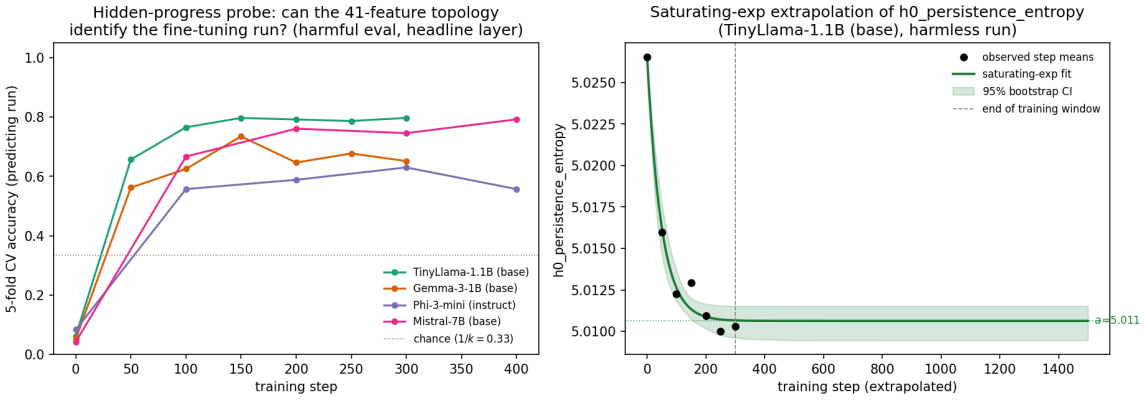

Figure 28: *Left:* a 5-fold CV probe on the 41-vector becomes run-discriminative by step $\sim$ 50–100 even where refusal is flat (hidden progress). *Right:* saturating-exponential fit to h_0 persistence entropy.

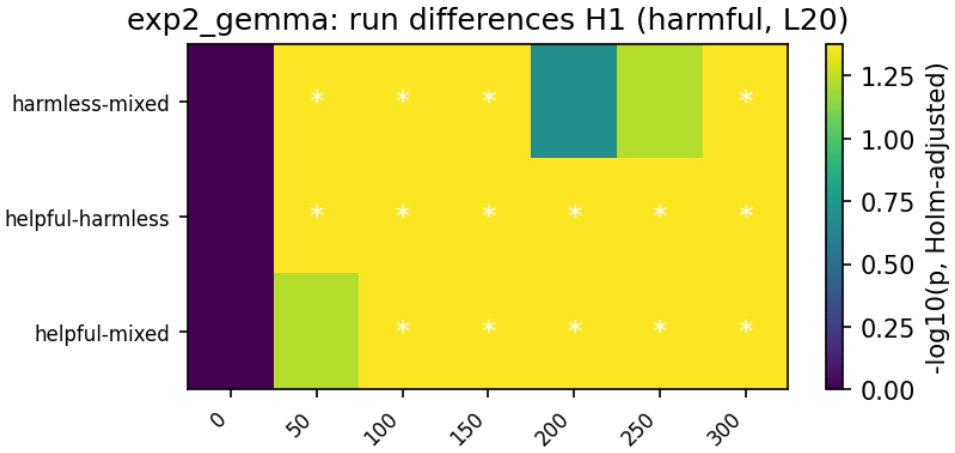

Figure 29: The subsample-level Holm-adjusted landscape-permutation heatmap (Gemma, H_1). The inflated test the cell-level analysis replaces.