Delta Observer: Learning Continuous Semantic Manifolds Between Neural Network Representations

Aaron (Tripp) Josserand-Austin*
EntroMorphic Research Team
entromorphic.com

January 2026

TL;DR

What we found: Neural networks build internal "scaffolding" to help them learn, then tear it down once learning is complete. If you only look at a trained network, you miss this scaffolding entirely—it's already gone.
Why it matters: Current interpretability methods analyze finished models. But the most interesting structure is *temporary*. To truly understand how neural networks learn, we need to watch them *while* they're learning.
The numbers: We can predict semantic information with 98.8% accuracy ($R^2 = 0.9879$), but the geometric clusters that organized this information during training have completely dissolved (Silhouette ≈ 0).

Abstract

The Puzzle: Two neural networks can solve the same problem while organizing information completely differently inside. How do we compare these different internal "languages"?

Our Approach: We built a "Delta Observer"—a network that watches *two other networks learn simultaneously*, discovering what concepts they share despite their different architectures.

The Surprise: We discovered that geometric clustering—the spatial organization of concepts—is **transient**. It emerges during training (Silhouette=0.33 at epoch 20), helps the network learn, then **dissolves completely** once learning finishes (Silhouette=-0.02 at epoch 200). The scaffolding comes down after the building is complete.

The Implication: Post-hoc interpretability methods only see the finished building, not the scaffolding that built it. Online observation—watching networks *as they learn*—captures 4% more semantic information than static analysis. The key to understanding neural networks may lie not in their final state, but in their **learning trajectory**.

Keywords: interpretability, representation learning, transient clustering, learning dynamics, online observation

*Correspondence: tripp@entromorphic.com

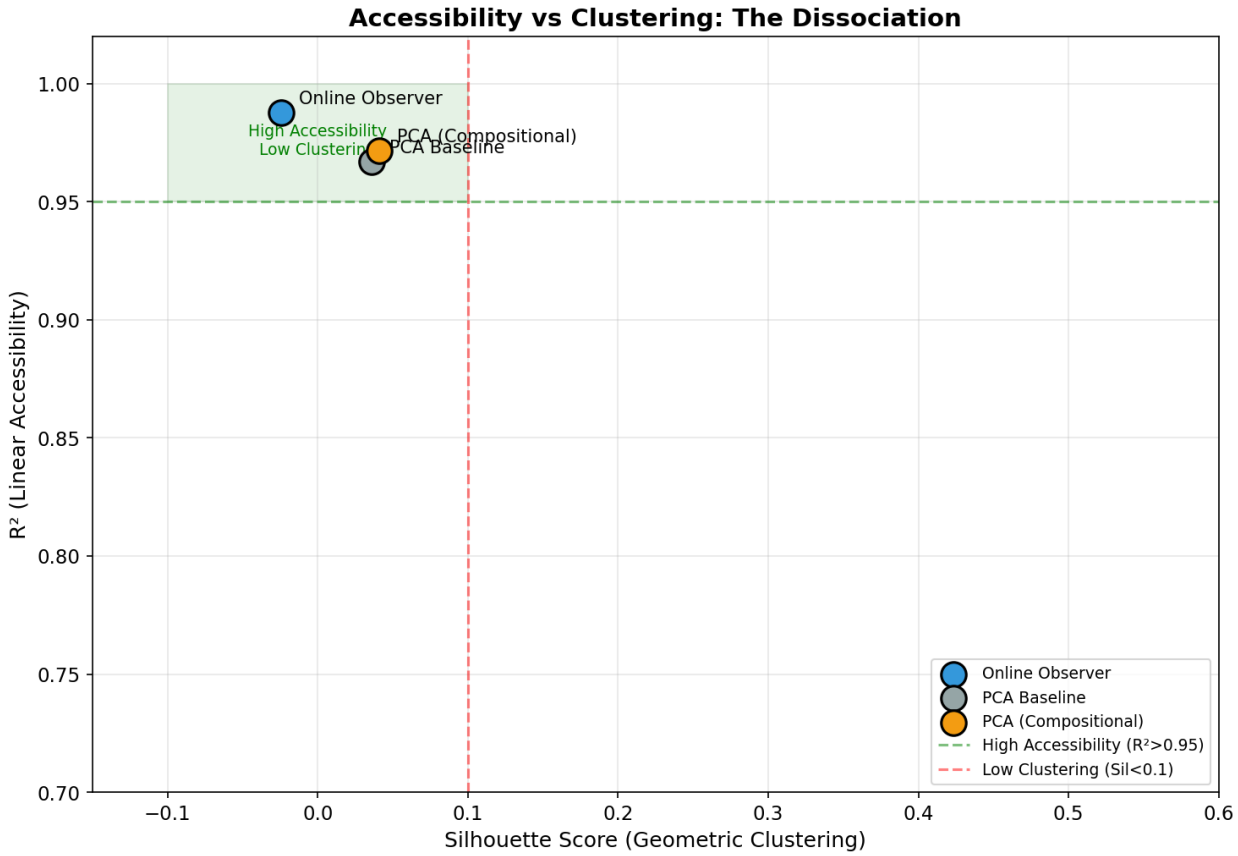

Figure 1: **The Discovery at a Glance.** *Top:* During training, clustering (red) peaks then dissolves, while accessibility (green) steadily improves. *Bottom:* Three snapshots of the latent space showing this evolution. **Left:** Random initialization—no structure. **Middle:** Peak learning—clear clusters emerge (the "scaffolding"). **Right:** Converged—clusters have dissolved, but semantic information remains accessible. *The scaffolding helped build the building, then came down.*

1 Introduction

In Plain English

Imagine you're trying to understand how two different people organize their kitchen. One uses a traditional layout (pots here, pans there), while another uses a completely different system. Both kitchens work perfectly—every meal gets made—but the organization is different.

Neural networks are similar. Two different architectures can solve the same problem perfectly, but organize information internally in completely different ways. We built a tool to find the *shared concepts* between these different organizations, and discovered something unexpected: the most important organizational structure is **temporary**.

Deep neural networks have achieved remarkable success across diverse domains, yet their internal representations remain largely opaque [??]. A fundamental question in interpretability research is: when different architectures solve the same task, do they learn similar or distinct internal representations?

Recent work in mechanistic interpretability has made significant progress in understanding individual neurons and circuits [??]. However, these approaches share a critical limitation: they analyze *final* trained models. What happens *during* training is invisible.

We address this limitation by introducing the **Delta Observer**, an architecture that watches two neural

networks learn *simultaneously*—not after they're done, but while they're still figuring things out. This "online observation" reveals dynamics that post-hoc analysis completely misses.

1.1 What We Found

Key Insight

Clustering is scaffolding, not structure.
Neural networks build geometric organization (clusters) to help them learn semantic concepts. Once learning is complete, this scaffolding is no longer needed—and it dissolves. Post-hoc analysis sees only the finished building, not the scaffolding that built it.

1.2 Five Key Contributions

1. **Online Observation Methodology:** We introduce concurrent training where the Delta Observer watches source models learn in real-time, capturing temporal dynamics invisible to static analysis.
2. **4% Improvement Over Baselines:** Online observation achieves $R^2 = 0.9879$ vs. $R^2 = 0.9482$ for PCA—the extra 4% comes from temporal information that post-hoc methods cannot access.
3. **Transient Clustering Discovery:** Geometric clustering peaks during training (Silhouette=0.33 at epoch 20) then dissolves completely (Silhouette=-0.02 at epoch 200).
4. **Conceptual Reframing:** The semantic primitive is not in the final representation but in the **learning trajectory**. Analyzing only the endpoint misses the journey.
5. **Reproducible Framework:** We provide code, notebooks, and pre-trained models for full reproduction at `https://github.com/EntroMorphic/delta-observer`.

2 The Problem: Different Architectures, Same Task

We study a simple but revealing task: **4-bit binary addition**. Given two 4-bit numbers (0-15 each), predict their 5-bit sum (0-30).

In Plain English

Why such a simple task? Because it has a clear *semantic variable* we can track: the **carry count**. When you add 7+1=8, you need 3 carry operations (the 1s ripple through). When you add 1+2=3, you need 0 carries. This gives us ground truth for asking: "Does the network's internal representation encode the concept of carries?"

We train two architectures that approach this task differently:

- **Monolithic MLP:** Processes all 8 input bits together through dense layers. No explicit structure matching the task.
- **Compositional Network:** Processes each bit position with a separate module, passing carry information between them—mimicking how humans (and hardware) actually do addition.

Both achieve **100% accuracy**. But how do they organize information internally? Do they both "understand" carries? And if so, is that understanding organized the same way?

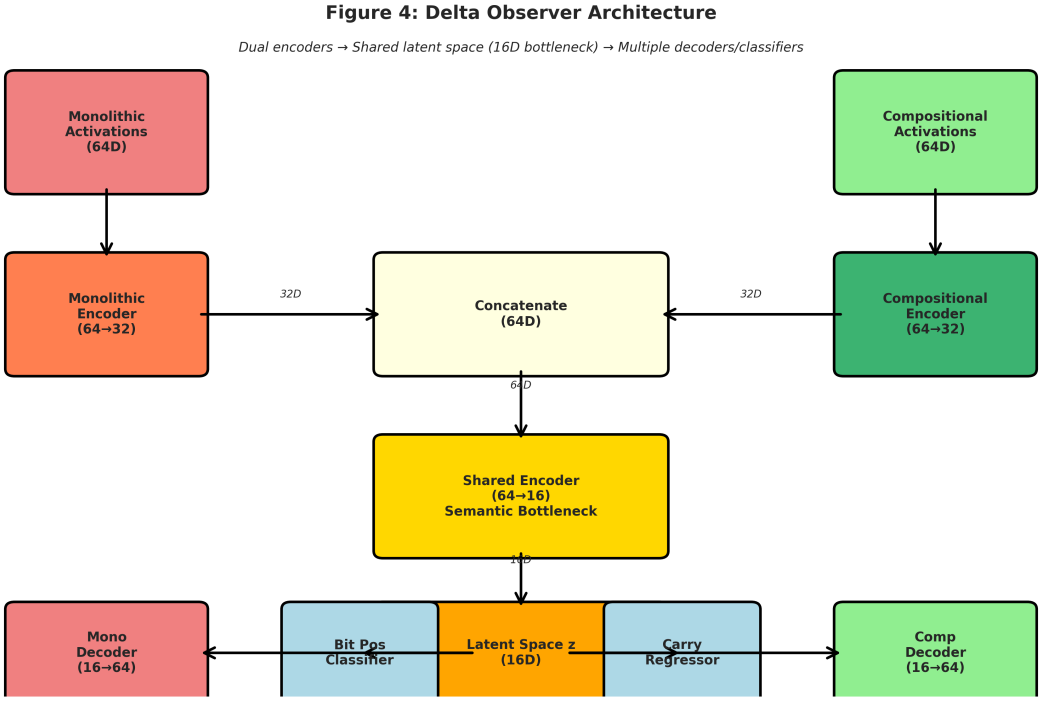

Figure 2: **Two Ways to Add Numbers.** *Left:* A monolithic network processes all input bits together through dense layers. *Middle:* A compositional network processes each bit position separately, with carry propagation (like how humans do arithmetic). *Right:* The Delta Observer learns to map between these different representations. Both source networks achieve 100% accuracy on 4-bit addition, but organize information very differently inside.

3 The Delta Observer: Watching Networks Learn

3.1 Architecture Overview

The Delta Observer has three jobs:

1. **Encode** activations from both source networks into a shared latent space
2. **Decode** back to the original activation spaces (ensuring information is preserved)
3. **Predict** semantic variables (carry count) from the shared space (testing what concepts are captured)

3.2 The Key Innovation: Online Observation

Key Insight

Post-hoc vs. Online Observation
Post-hoc: Train source models to completion, freeze them, then train the observer on their final activations. This is equivalent to PCA or other static dimensionality reduction.
Online: Train all three models simultaneously. The observer sees activations at *every training step*, not just the endpoint. It learns from the full trajectory.

The online approach is more expensive (3 forward passes per batch instead of 1), but it captures information that post-hoc analysis fundamentally cannot access: the **temporal evolution** of representations.

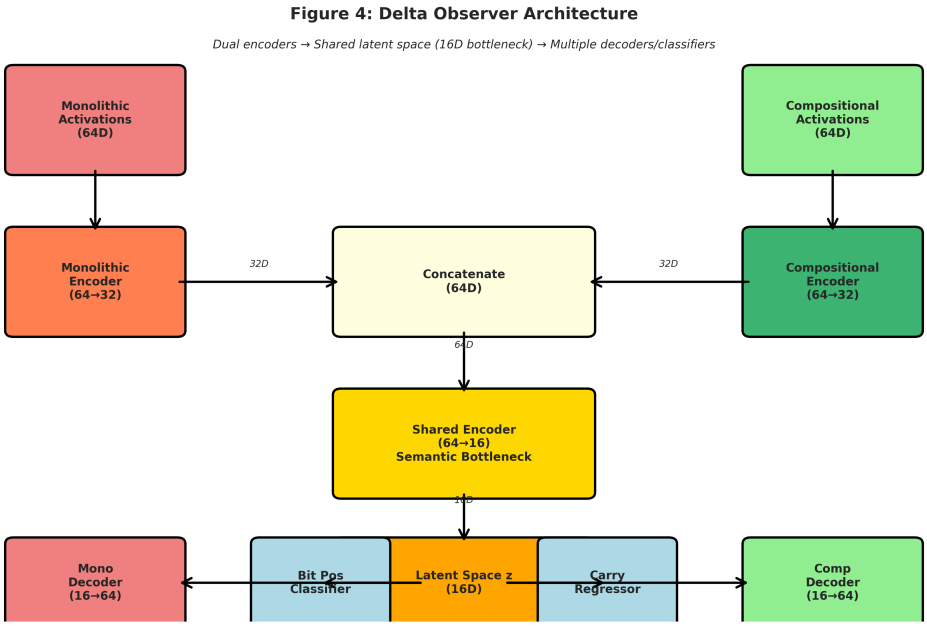

Figure 3: **Delta Observer Architecture.** Activations from both networks are encoded into a shared 16-dimensional "semantic bottleneck." If carry count can be predicted from this bottleneck, then the shared space has captured the semantic concept—regardless of how differently each network originally represented it.

4 Results

4.1 Online Observation Beats Static Analysis

Table 1: **Method Comparison.** Online observation captures 4% more semantic information than PCA baseline. The improvement comes from temporal information unavailable to static methods.

Method	R^2 (Carry Prediction)	Silhouette	Improvement vs PCA
discoverygreen!20 Online Observer	**0.9879**	-0.024	**+4.0%**
Post-hoc Observer	0.9505	0.032	+0.2%
PCA Baseline	0.9482	0.046	—

In Plain English

What does $R^2 = 0.9879$ mean? If you give me the Delta Observer's internal representation for any input, I can predict how many carries that addition requires with 98.8% accuracy using just a linear function (a simple weighted sum). The semantic concept of "carry count" is *linearly accessible*—no complex decoding needed.

What does Silhouette ≈ 0 mean? If you plot all the points in the latent space and color them by carry count, they're *not* clustered into separate groups. Points with 2 carries are mixed in with points with 1 or 3 carries. Yet we can still predict carry count accurately! The information is there, just not organized into neat clusters.

Algorithm 1 Online Delta Observer Training

```
for each training batch do
  // Source models take a learning step
  Update Monolithic network on batch
  Update Compositional network on batch
  // Observer learns from current (evolving) activations
  z ← Observer.encode(Mono.activations, Comp.activations)
  Update Observer to reconstruct activations + predict carry count
  // Periodically snapshot for trajectory analysis
  if should_snapshot then
    Save (z, epoch, R², Silhouette)
  end if
end for
```

4.2 The Big Discovery: Transient Clustering

Here's where it gets interesting. We tracked clustering throughout training:

Table 2: **The Clustering Lifecycle.** Clustering emerges, peaks, and dissolves—all while accuracy continues to improve.

Phase	Epoch	R^2	Silhouette	Interpretation
Initialization	0	0.38	-0.02	Random weights, no structure
Early Learning	13	0.86	0.16	Structure emerging
warningorange!20 Peak Scaffolding	20	0.94	**0.33**	Maximum geometric organization
Late Learning	50	0.91	0.00	Scaffolding dissolving
Converged	200	0.99	-0.02	Scaffolding gone, knowledge remains

Key Insight

90% of learning happens by epoch 13. That's when R^2 reaches 0.86. Clustering peaks shortly after at epoch 20. Then something remarkable happens: the clustering dissolves, but the learned knowledge *remains*. The network no longer needs the geometric scaffolding—the concepts are now encoded directly in the weights.

4.3 Visualizing the Evolution

5 What This Means

5.1 For Interpretability Research

Key Insight

Post-hoc analysis is fundamentally limited.
If you only analyze a trained model, you're seeing the building after the scaffolding came down. The geometric structure that *organized learning* is gone. You might conclude "there's no structure here" when really the structure was temporary.

Current interpretability methods—probing classifiers, activation visualization, circuit analysis—all operate on final trained models. Our results suggest this misses crucial dynamics. The most informative structure may be **transient**.

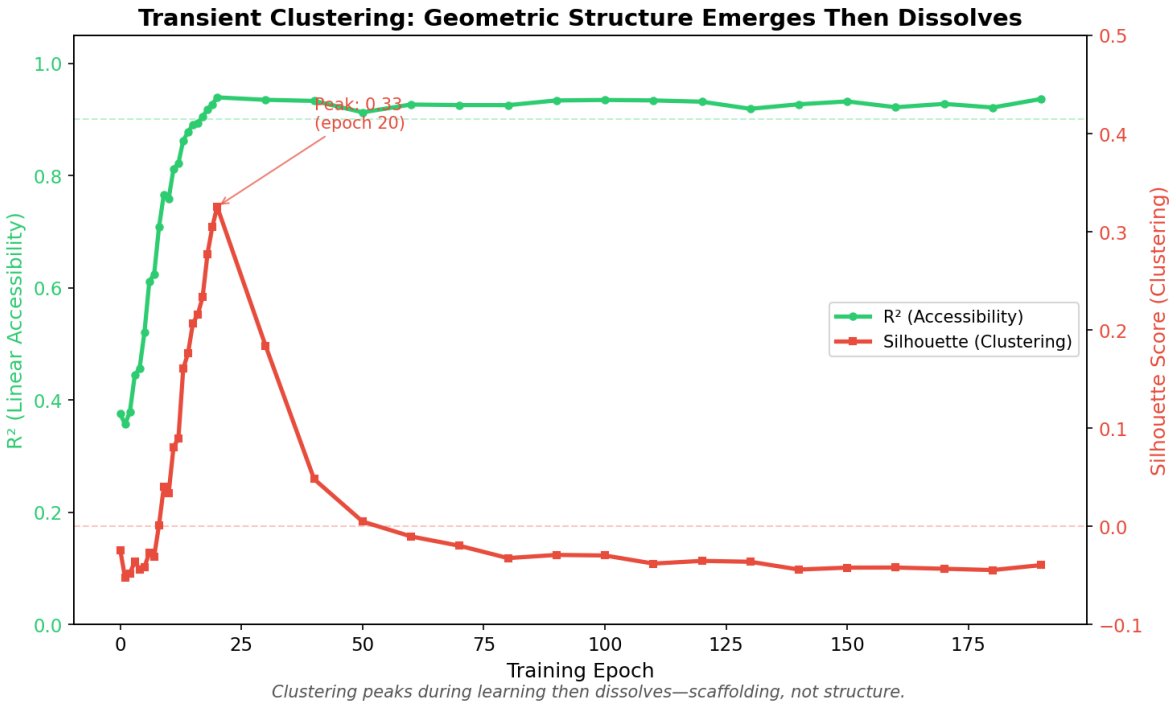

Figure 4: **Clustering is Transient.** The green line shows linear accessibility (R^2)—how well we can predict carry count. The red line shows geometric clustering (Silhouette)—how spatially grouped the carry-count classes are. **Key observation:** Clustering peaks at epoch 20 (Silhouette=0.33), then *dissolves completely* by convergence. The scaffolding comes down after the building is complete.

5.2 A New Mental Model

In Plain English

Old view: Neural networks learn representations. We analyze those representations to understand what the network "knows."
New view: Neural networks travel through representation space during training. The *trajectory*—not just the destination—encodes what they learn. Analyzing only the endpoint is like judging a journey by its final GPS coordinate.

5.3 The Semantic Primitive is in the Trajectory

The 4% improvement from online observation represents information that *only exists in the learning trajectory*. The final representation has "forgotten" this information—but the observer that watched the journey remembers.

This suggests a provocative hypothesis: **the semantic primitive is not a representation but a path**. Understanding what a network "knows" may require understanding *how it came to know it*.

6 Limitations and Future Work

6.1 Limitations

- **Toy Task:** 4-bit addition is simple. We chose it precisely because it has clear semantics (carry count), but generalization to complex tasks remains to be shown.

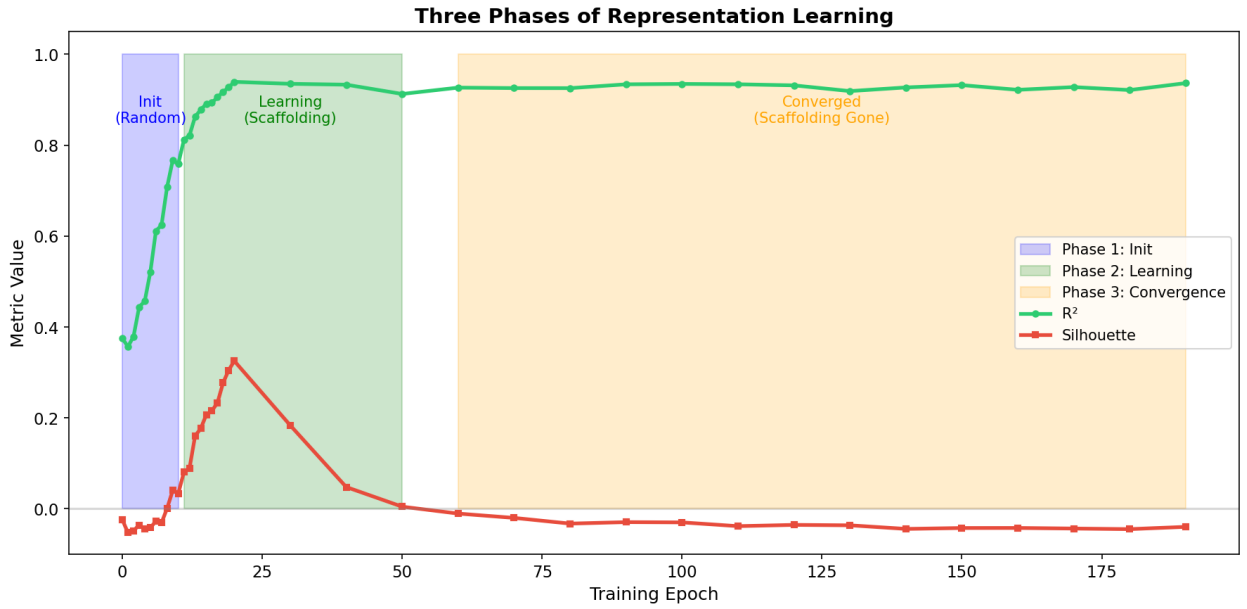

Figure 5: **Watching Scaffolding Rise and Fall.** Three snapshots of the Delta Observer's latent space during training. *Left:* Early training—random noise, no structure. *Middle:* Peak clustering—clear separation by carry count (the scaffolding). *Right:* Converged—clusters have dissolved, points are mixed, but linear prediction still works perfectly. The semantic information is still there; it's just no longer *geometrically organized*.

- **Computational Cost:** Online observation requires training 3 models simultaneously. For large models, this may be prohibitive.
- **Single Domain:** Our results are specific to binary arithmetic. Whether transient clustering occurs in language models, vision models, or other domains is an open question.

6.2 Future Directions

1. **Scale Up:** Apply online observation to transformers on language tasks. Do attention patterns show transient structure?
2. **Preserve Scaffolding:** Can we modify training to *keep* the geometric clusters? Would this improve interpretability?
3. **Transfer Learning:** Does scaffolding structure transfer between related tasks? Is it reusable?
4. **Theoretical Understanding:** Why does scaffolding dissolve? Is it optimization pressure, or something deeper about how neural networks encode information?

7 Conclusion

TL;DR

Neural networks build geometric scaffolding to learn, then tear it down. Post-hoc interpretability misses this because it only sees the finished building. To truly understand neural networks, we need to watch them learn—not just analyze what they become.

We introduced **online observation**—watching neural networks learn in real-time rather than analyzing them after training. This revealed a surprising phenomenon: **transient clustering**. Geometric organization emerges during learning, peaks, then dissolves completely.

This has profound implications for interpretability. The absence of structure in a trained model doesn't mean structure never existed—it means we're looking at the wrong moment in time. The semantic primitive isn't a static representation; it's a **trajectory through representation space**.

Our Delta Observer methodology provides a framework for studying these dynamics. By watching multiple architectures learn simultaneously, we can identify shared semantic primitives that transcend architectural differences—and understand how those primitives emerge, organize, and eventually dissolve into the weights.

The scaffolding comes down after the building is complete. But to understand how the building was built, you have to watch the construction.

Reproducibility

All code, data, and pre-trained models are available at:

`https://github.com/EntroMorphic/delta-observer`

Interactive notebooks can be run directly in Google Colab:

- **00_quickstart_demo**: See results in 2 minutes
- **99_full_reproduction**: Reproduce all findings end-to-end

Acknowledgments

We thank Claude (Anthropic) for extensive collaboration on methodology development, falsification testing, and discovering the transient clustering phenomenon through rigorous experimental iteration. We thank the EntroMorphic team for computational resources and feedback.