Nick Clark · Published July 30, 2026 · PDF

When a Model Learns Something It Was Not Allowed To

A licensing term ends on an archival collection one model stewardship lead fine-tuned against, and she cannot say where inside her shipped checkpoint that content still does work, how deeply it reached, or with what force. She can name the files her data loader sampled. She cannot name the layers they reached. In her pipeline the governance decision was made once, at acquisition, and never entered the training loop again, so the thing she now needs to prove was never written down while it was still knowable. United States Patent Application 19/647,395 discloses an architecture that treats each training example as a proposed semantic mutation, evaluated for admissibility and routed across model depth before any gradient is applied.

The Tuesday the term ended

A model stewardship lead at a mid-sized company that ships a drafting assistant into a regulated profession opens a message from a licensor's counsel on an ordinary Tuesday morning. The two-year term on an archival collection her team fine-tuned against has ended. Counsel is asking a narrow, reasonable question: please confirm in writing that the collection no longer influences the deployed model, and describe the basis for that confirmation.

She can answer part of it. Her data loader kept a manifest, so she can list the documents that were sampled, the epochs in which they appeared, and the batch sizes. That is the extent of what her pipeline preserved. What counsel is actually asking about is influence, and influence is the one property her setup never recorded. The checkpoint she shipped in March was produced by eight weeks of gradient updates in which each sampled document contributed to every layer's parameters with equal structural authority. Nothing in her training loop was positioned to decline a document, to attenuate its contribution, or to confine it to a particular region of the model's depth, and nothing in it wrote down where a contribution went.

So she sits with a question she cannot answer from evidence, about an artifact she cannot inspect. Six customer-specific adapters descend from that March checkpoint. Her validation history, the evaluation results her customers reviewed before signing, and the behavioral baselines her safety reviewers approved are all anchored to it.

What she loses, and why it does not come back

The first loss is the answer itself. In her deployment, the moment at which the collection's depth of integration was knowable was the moment the gradients were applied, and that moment passed in March without being captured. She can retrain from a corpus with the collection removed, but the checkpoint that emerges will not be the March checkpoint minus one collection. It will be a different model, with different evaluation numbers, different failure modes, and a validation history that no longer matches anything her customers signed off on. She can buy the eight weeks of compute back. She cannot buy back the continuity.

The second loss is the six downstream adapters. Each was tuned against the March checkpoint's behavior, and each carries its own customer acceptance record. Were she to rebuild the base, her team would owe six re-tunings and six re-validations, each with

a customer review cycle that runs in quarters. For her purposes, the practical effect is that a single expired term propagates into every deployment her company has made this year.

The third loss is quieter and does not repair. Counsel's question established, in writing, that her organization could not answer it, and that fact will still be in the file when she goes to negotiate the renewal. She cannot make a credible forward commitment about the next corpus either, because her pipeline as configured today gives her no instrument that would let her honor it. The undertaking she would need to offer is an undertaking about depth and separability, and her training loop has no vocabulary for either.

Why the shape of her problem resists effort

Her difficulty is not that her team was careless. Her acquisition review was thorough and her legal intake caught the term length correctly. The difficulty is where in her pipeline the governance decision sits.

In her setup, governance happens once, upstream, at the point where content is admitted to the corpus. After that, inside her loader, the content becomes an undifferentiated sample in a shuffled stream, and the training loop she runs has no boundary at which the question could be asked again. Her forward pass computes a loss and her backward pass applies gradients, and between those two steps there is no place in her pipeline where a decision about a specific example could be rendered, recorded, or acted upon. The optimizer she uses does not know which document produced which gradient, which is precisely why she cannot ask it afterward.

The second structural feature is that the correction available to her is retrospective and approximate. Were she to attempt to remove the collection's influence after the fact, she would be attempting to reverse a contribution that gradient-based optimization diffused across a very large number of parameters through nonlinear dynamics. There

is no residue in her checkpoint that says which parameter changes belonged to which source. Her intervention would necessarily be an estimate of an influence she can only model, applied to a model she can only evaluate behaviorally.

The third feature is that her lever is uniform when her obligation is not. In her March run, an unrestricted reference text and a term-limited archival document were treated identically by the training machinery: both reached the shallow layers where lexical patterns are encoded, and both reached the deep layers where abstract structure is encoded. Her legal obligations differ sharply between those two documents. Her architecture offered her no way to express the difference at the point where the difference would have mattered.

What the filed architecture does

United States Patent Application 19/647,395 discloses, in accordance with an embodiment, a training loop reconceived as a governed execution environment, in which each training iteration constitutes a proposed semantic mutation to the model's accumulated knowledge state. The semantic execution substrate is positioned at training-loop boundaries, operating between the forward-pass loss computation and the backward-pass gradient application. Gradients are computed as in conventional training; the gradient signal is then modulated, gated, or selectively routed across model depth according to an admissibility determination that the substrate renders.

Referring to FIG. 11A of the disclosure, a training batch (1100) provides input to a semantic substrate (1102), which feeds a depth profile router (1104), from which three paths diverge to shallow layers (1106), middle layers (1108), and deep layers (1110). Referring to FIG. 11B, a training loop (1112) feeds the semantic substrate (1102), which produces an admissibility determination (1114), yielding a depth profile (1116) that routes to an aggregation module (1118).

In an embodiment, each training example is required to carry semantic metadata sufficient for the substrate to render a determination: an entropy band classification, a slope position within the platform's trust-slope hierarchy, a content provenance record identifying source, acquisition pathway, and chain of custody, and a policy scope identifying governance constraints such as licensing terms, usage restrictions, temporal validity bounds, and exclusion mandates. An example consisting solely of raw content without such metadata is described as inadmissible by default. Non-training, the refusal to integrate an example, is described as a valid computational result rather than an error condition, and the refusal is recorded in the training provenance log as a governed event with the identity of the example and the reason.

The disclosure describes determinations that go beyond admit or reject. In an embodiment, the substrate renders graded determinations expressed as a training depth profile: a structured object comprising a per-layer or per-block contribution weight vector, where a weight of one permits the full gradient signal to reach that block, a weight of zero prevents any gradient from that example reaching it, an intermediate weight attenuates by the specified factor, and a weight greater than one amplifies. Three complementary techniques for applying the profile are disclosed: gated residual connections, attention-based depth selection, and layer-specific scaling factors.

Section 11.5 addresses the situation on her desk directly. In an embodiment, content admitted under time-limited licensing agreements is trained with a suppressed depth profile, in which contribution weights for deeper blocks are set to zero or near zero, confining the example's influence to shallower layers so that de-emphasis can be pursued through targeted shallow-layer adjustment rather than model-wide intervention. Content from a governed exclusion corpus may receive a zero-weight depth profile, described as setting the contribution weight to zero at every layer. Where multiple policies apply to one example, the disclosure describes hierarchical resolution in which the most restrictive policy prevails, with the resolution recorded for later audit. Referring to FIG. 11D, a freely licensed content node (1130), a time-limited

content node (1132), and an exclusion corpus content node (1134) each feed a policy resolution module (1136), which produces an approved retention set (1138) and an excluded content set (1140).

The provenance side is what would give her something to send counsel. In an embodiment, the training provenance log is a chronologically ordered, append-only structure, each entry timestamped, sequentially numbered, and annotated with epoch, iteration, and batch index, recording the entropy band, slope position, the depth aggregation profile applied, the per-block contribution weight actually reached, the governing policy objects, the content provenance record, and the admissibility determination with its reason. Forward queries trace a source through the profile and weights that governed its integration. Reverse queries begin from an observed behavior and identify the training content that was structurally permitted to influence the active layer blocks. Section 11.7 further discloses a memorization assessment that classifies an observed similarity as shallow memorization, deep memorization, or absent memorization, based on what the log shows about the depth at which the similar content was integrated.

Where the disclosed architecture stops short of her situation

It is prospective, and her problem is retrospective. The subject matter described here governs a training run while it is happening. It would give her successor at the next run a per-example record and a depth profile; it does not reconstruct the March checkpoint's history, because for that run no such determinations were made and no such log exists. For her current obligation, the disclosure offers a way to not be in this position again rather than a way out of it.

It would also require work she has not done. In an embodiment, the disclosure describes the training corpus being semantically enriched before training, with entropy band, slope position, provenance, and policy scope attached to each unit. Her present acquisition pipeline captures license terms in a legal intake system, not as structured

metadata riding alongside each document into the data loader, so adoption for her would begin with the enrichment layer, not the training loop. The disclosure further notes that content without a verified anchored identity is flagged as provenance-incomplete, and that governance policy may restrict such content to shallow layers, which for her archival collection would depend on what structural identity her acquisition pathway preserved.

The reverse query would not settle the attribution question she might most want settled. The disclosure states that a reverse query does not definitively attribute a model behavior to specific training content, because the nonlinear dynamics of gradient-based optimization preclude exact attribution. What it describes producing is a bounded attribution set, substantially narrower than the full corpus. Were she to rely on it, she would be offering counsel a scoped set and a depth record, not a proof of exclusive causation.

Finally, the outcomes are conditioned on the policies actually declared. A suppressed depth profile confines influence to the blocks its weight vector specifies, and the described separability follows from those declared weights. Where a policy for her collection was never authored, or authored with a scope that does not match the term counsel is asking about, the architecture would faithfully record and enforce the wrong thing. The disclosure also describes block-level granularity rather than per-layer granularity for deep networks, so the resolution of any answer she could give would be the resolution of her block structure. And nothing here speaks to what her licensor's counsel will accept as sufficient evidence.

Disclosure Scope

This article is a technical description of subject matter disclosed in United States Patent Application 19/647,395. It describes embodiments and architectural mechanisms set out in that filing, using the filing's own mechanism names and reference numerals. Nothing in this article characterizes the scope of any claim, and nothing here is an

admission regarding the state of the art. The party, deployment, and circumstances described above are illustrative and do not describe any actual person, organization, or product.

Training Governance (/training-governance)

All 49 steps → (/inventive-steps)

Govern what the model learns, at what depth, with what provenance.

Chapter 11 (/patents/19-647395/chapters/training-governance)

PRIMARY TECHNICAL DISCLOSURE

- Depth-Selective Training Governance for Machine Learning Systems (/articles/depth-selective-training-governance-for-machine-learning-systems)

SECONDARY TECHNICAL

- Training Examples as Proposed Semantic Mutations in Governed Training (/articles/training-governance/mutation-proposals)
- Entropy-Band-Indexed Training Depth Profiles (/articles/training-governance/entropy-depth-profiles)
- Depth-Selective Gradient Routing for Governed Training (/articles/training-governance/gradient-routing)
- Training-Level Memorization Detection (/articles/training-governance/memorization-detection)
- Differential Privacy Through Depth-Selective Routing (/articles/training-governance/differential-privacy)
- Governed Fine-Tuning With Verifiable Provenance (/articles/training-governance/fine-tuning-provenance)
- The Training Loop as a Governed Execution Environment (/articles/training-governance/governed-training-loop)
- Policy-Governed Knowledge Retention and Suppression (/articles/training-governance/knowledge-retention)
- Provenance-Traceable Training Dynamics (/articles/training-governance/provenance-tracing)
- Curriculum-Integrated Depth Scheduling (/articles/training-governance/curriculum-depth)
- Affect-Modulated Training Depth (/articles/training-governance/affect-modulated-depth)

- Training-Inference Governance Integration (/articles/training-governance/training-inference-integration)
- Training Governance for Human-Relatable Agents (/articles/training-governance/human-relatable-training)
- Governed Training with Depth-Selective Gradient Aggregation (/articles/training-governance/edge-fleet-training)

APPLICATIONS · GENERAL

- Rights-Compliant Model Training Through Depth-Selective Gradient Routing (/articles/training-governance/rights-compliant-training)
- Regulated Industry Model Governance: Verifiable Training Provenance for AI Compliance (/articles/training-governance/regulated-model-governance)
- Training Governance for Medical AI: Auditable, Depth-Controlled Clinical Model Training (/articles/training-governance/medical-ai-training)
- Training Governance for Legal AI: Encoding Precedent Hierarchy Into the Model (/articles/training-governance/legal-ai-training)
- Training Governance for Financial AI: Auditable Model Risk Management Under SR 11-7 (/articles/training-governance/financial-model-training)
- Training Governance for Defense AI (/articles/training-governance/defense-ai-training)
- How to Train an Educational AI Tutor That Learns Pedagogy Deeply and Misconceptions Shallowly (/articles/training-governance/educational-model-training)
- Copyright-Compliant Training for Creative and Generative AI Models (/articles/training-governance/creative-ai-training)
- Training-Data Provenance for Regulated Autonomy: UNECE, FDA, and EU AI Act Compliance (/articles/training-governance/regulated-autonomy-training)
- Tamper-Evident Fleet Training Records for Maritime and Agricultural Operations Without Cellular Connectivity (/articles/training-governance/maritime-fleet-training)
- 21 CFR Part 11 Compliance for AI Model Training: Audit Trails, Electronic Signatures, and Provenance (/articles/training-governance/cfr-21-part-11)
- **When a Model Learns Something It Was Not Allowed To (/articles/training-governance/training-governance-stakes)**

APPLICATIONS · SPECIFIC

- OpenAI Training vs Governed Depth-Selective Training (/articles/training-governance/openai-training)
- Anthropic Constitutional Training vs Governed Training: What Depth-Selective Provenance Adds (/articles/training-governance/anthropic-training)

- Stable Diffusion Training vs Governed Provenance: The Missing Layer in Stability AI's Pipeline (/articles/training-governance/stability-ai)
- Midjourney vs Governed Training: Depth-Selective Provenance for Generative Art (/articles/training-governance/midjourney)
- Scale AI Alternative: Governed Learning Beyond Data Labeling (/articles/training-governance/scale-ai)
- Labelbox Alternative for Governed Training: Annotation Workflows vs Learning Dynamics (/articles/training-governance/labelbox)
- Snorkel AI Programs Labels but Does Not Govern Gradient Depth (/articles/training-governance/snorkel-ai)
- Weights & Biases Alternative for Governed Training: Tracking vs Depth-Selective Governance (/articles/training-governance/weights-biases)
- Determined AI Alternative: Governed Training Beyond Compute Orchestration (/articles/training-governance/determined-ai)
- MosaicML Optimizes Training Efficiency, Not Learning Governance (/articles/training-governance/mosaic-ml)
- Tesla Shadow Mode vs Governed Fleet Training (/articles/training-governance/tesla-shadow-mode)
- Symbotic Warehouse Automation vs Governed Training Provenance (/articles/training-governance/symbotic-warehouse)
- Governed Alternative to Google Gemini and Vertex AI Tuning (/articles/training-governance/google-gemini-tuning)
- OpenAI Fine-Tuning and RFT vs Governed, Depth-Selective Training (/articles/training-governance/openai-fine-tuning-rft)
- PathAI Alternative: Governed Digital-Pathology Training (/articles/training-governance/pathai-pathology)
- Tempus AI vs Governed Medical-AI Training: Training-Step Provenance (/articles/training-governance/tempus-ai-medical)

HOW-TO GUIDES

- How to Build an Auditable, Rights-Compliant AI Training Pipeline (/articles/guides/build-an-auditable-training-pipeline)
- How to Meet the EU AI Act's Training-Data Transparency Requirements: A Provenance-Bound Architecture (/articles/guides/eu-ai-act-training-data-transparency)
- How to Prove an AI Model Was Trained Only on Licensed Data (/articles/guides/prove-model-trained-on-licensed-data)

TERMINOLOGY

– depth-selective aggregation (/glossary#depth-selective-aggregation)

Training Governance overview → (/training-governance)