Provenance-Traceable Training Dynamics

by Nick Clark | Published March 27, 2026 | PDF

Every parameter change in a governed model is traceable to the specific training examples that caused it. Provenance-traceable training dynamics record the complete causal chain from data point through gradient computation to parameter update, creating an audit trail that enables precise attribution of model behavior to training inputs.

What It Is

Provenance-traceable training dynamics record the causal relationship between training examples and parameter changes at a granularity sufficient for meaningful attribution. Each parameter update is annotated with the training examples that contributed to it, their governance classifications, and the depth profiles under which they were admitted.

Why It Matters

When a model produces unexpected or undesirable behavior, provenance tracing enables investigation of which training examples contributed to that behavior. When a model is audited for rights compliance, provenance tracing enables verification that all contributing examples were properly licensed. Without provenance, model behavior is an opaque function of its training data with no mechanism for attribution.

How It Works

The training lineage records, for each training step, the admitted examples, their classifications, the gradient contributions to each parameter group, and the resulting parameter deltas. This creates a queryable record where any parameter change can be traced back to its contributing examples and forward from any example to its parameter influence.

Provenance queries can answer questions like: which examples most influenced this parameter group? Which governance policy admitted the examples that shaped this capability? What would happen if a specific example were removed from training?

What It Enables

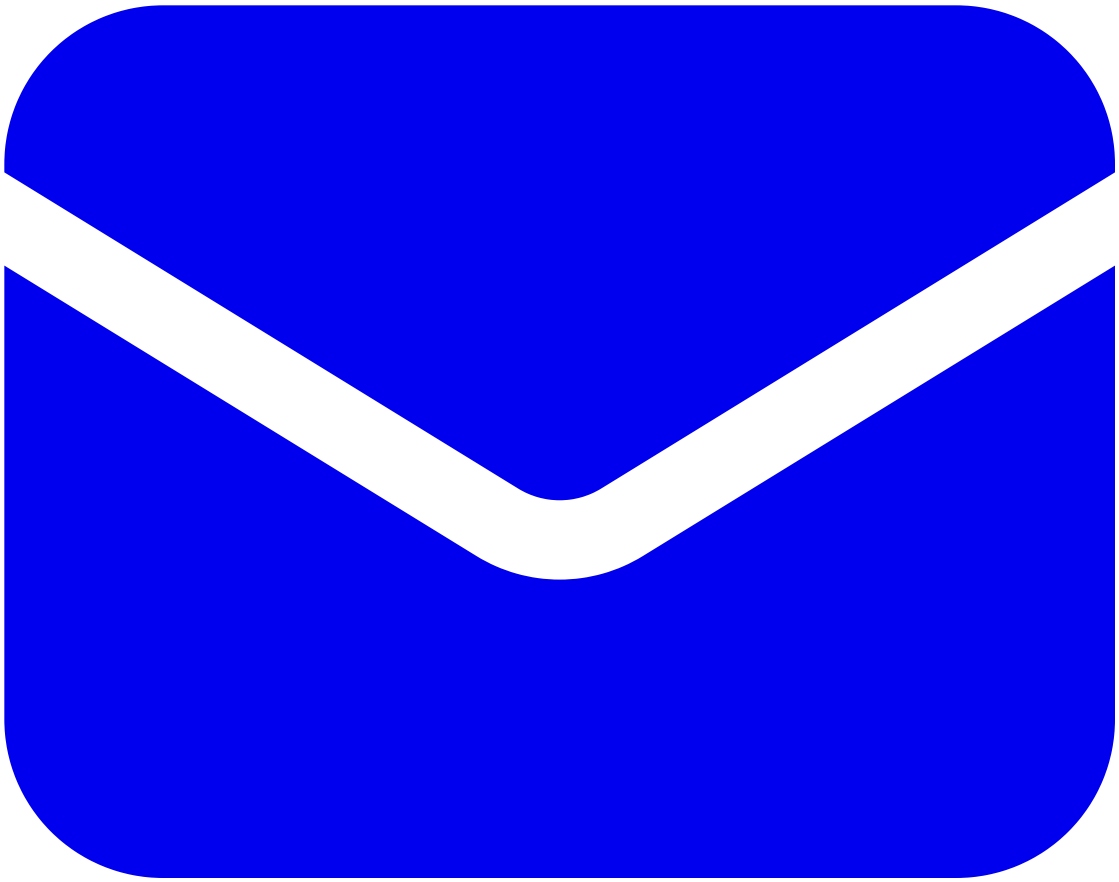
Provenance tracing enables accountable machine learning where every aspect of model behavior can be attributed to specific training decisions. This attribution is the foundation for rights compliance (proving authorized training data), safety assurance (identifying training inputs that contributed to undesirable behavior), and regulatory compliance (demonstrating governed training processes).

Training Governance All 21 steps →

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

Primary Technical Disclosure
○ Depth-Selective Training Governance for Machine Learning Systems
Secondary Technical
○ Training Examples as Proposed Semantic Mutations○ Entropy-Band-Indexed Training Depth Profiles○ Depth-Selective Gradient Routing for Governed Training○ Training-Level Memorization Detection○ Differential Privacy Through Depth-Selective Routing○ Governed Fine-Tuning With Verifiable Provenance○ The Training Loop as a Governed Execution Environment○ Policy-Governed Knowledge Retention and Suppression● Provenance-Traceable Training Dynamics○ Curriculum-Integrated Depth Scheduling○ Affect-Modulated Training Depth○ Training-Inference Governance Integration○ Training Governance for Human-Relatable Agents
Applications (General)
○ Rights-Compliant Model Training Through Depth-Selective Routing○ Regulated Industry Model Governance With Provenance○ Training Governance for Medical AI○ Training Governance for Legal AI○ Training Governance for Financial Model Training○ Training Governance for Defense AI○ Training Governance for Educational AI Models○ Training Governance for Creative AI
Applications (Specific)
○ OpenAI's Training Pipeline Has No Depth-Selective Governance○ Constitutional AI Training Lacks Depth-Selective Control○ Stable Diffusion's Training Has No Provenance Layer○ Midjourney Trains Aesthetics Without Governed Depth○ Scale AI Labels Data Without Governing What Models Learn○ Labelbox Manages Annotation Workflows, Not Learning Dynamics○ Snorkel AI Programs Labels but Does Not Govern Gradient Depth○ Weights & Biases Tracks Experiments, Not Learning Governance○ Determined AI Orchestrates Compute, Not Learning Depth○ MosaicML Optimizes Training Efficiency, Not Learning Governance
Training Governance overview →
AQ
deterministic
autonomy

Last updated: 2026-03-03



-
- nick@qu3ry.net
- 72 28 14 36 01

Invented by Nick Clark | Founding Investors: Devin Wilkie