Training-Inference Governance Integration

by Nick Clark | Published March 27, 2026 | PDF

Governance applied during training must be consistent with governance applied during inference. A model trained under specific content depth restrictions should enforce compatible restrictions during inference. Training-inference governance integration ensures this consistency by deriving inference governance constraints from training governance records, creating a unified governance lifecycle from training through deployment.

What It Is

Training-inference governance integration links the governance policies applied during training to the governance policies enforced during inference. Content that was restricted to shallow integration during training should not be deeply relied upon during inference. Content classes excluded from training should not appear in inference outputs as if the model had learned them.

Why It Matters

Disconnected training and inference governance creates inconsistencies. A model might be trained to avoid memorizing specific content but then be deployed without corresponding inference restrictions, allowing it to reconstruct memorized patterns through creative prompting. Integrated governance prevents such inconsistencies by ensuring inference constraints match training constraints.

How It Works

The training lineage includes a governance manifest that summarizes the training policies applied: which content classes were admitted at which depths, what memorization thresholds were enforced, and what content was excluded. The inference governance system reads this manifest and applies compatible constraints: content trained at shallow depth is restricted from deep inference reliance; excluded content classes trigger inference-time guardrails.

What It Enables

Training-inference integration enables end-to-end governance where training decisions propagate to deployment constraints automatically. Models carry their governance manifest with them, and any deployment environment can configure appropriate inference governance based on how the model was actually trained. This creates consistent, auditable governance across the entire model lifecycle.

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

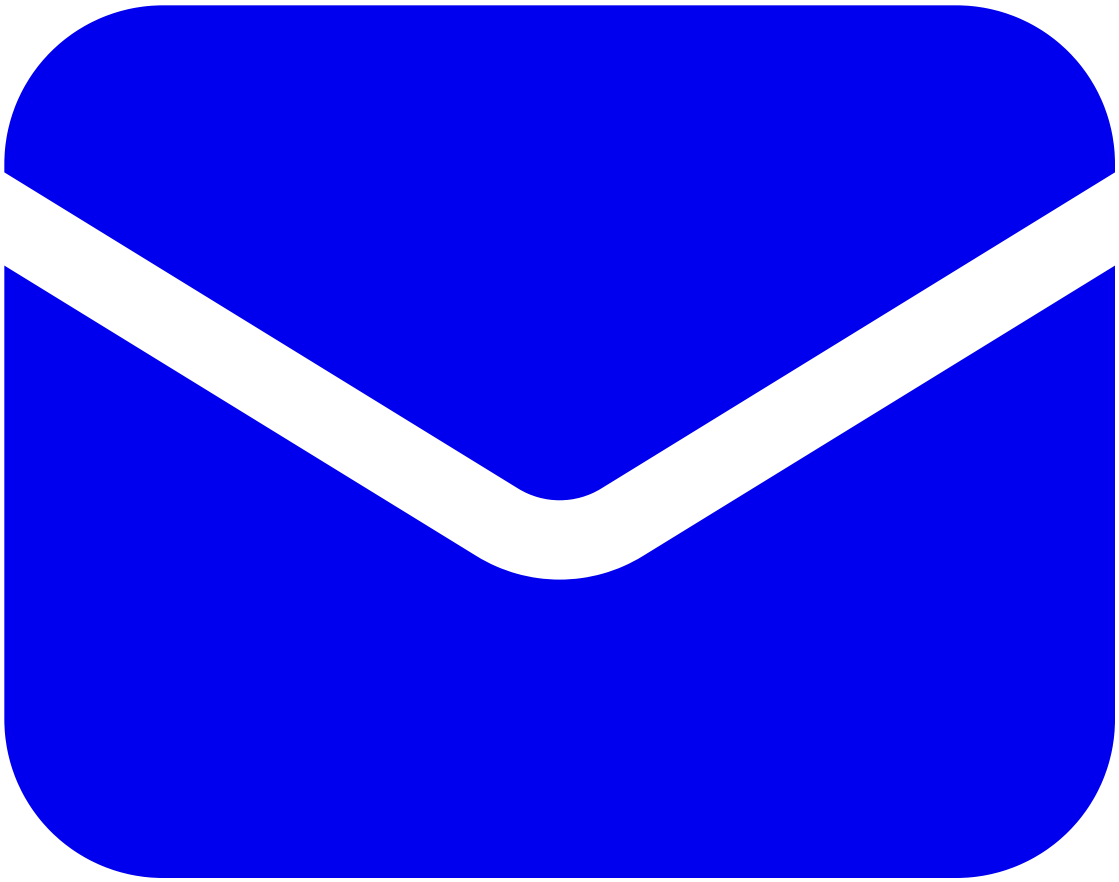
Legal

Subject to one or more pending U.S. and international patent applications, see Patents for the current list and status. No license, express or implied, is granted. Any use requires a separate written agreement—see Licensing. Patent applications referenced on this site are pending. Claim scope, if any, is subject to examination and may issue in altered form or not at all. See Legal for terms and conditions.

Adaptive Query™ is a trademark of Nicholas Clark. U.S. federal registration is pending. federal registration. AQ™ , AQ Inside™ , Adaptive Index™ , Adaptive Network™ , Semantic Agent™ , @AQ™ , AQID™ , and Adaptive Coin™ are used as trademarks in connection with the Adaptive Query platform and brand. Other names may be trademarks of their respective owners.

Platform operated by Adaptive Query LLC, which provides patent and trademark licensing services. Copyright © 2025-2026 Nicholas Clark. All rights reserved.

Last updated: 2026-03-03



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

Invented by Nick Clark | Founding Investors: Devin Wilkie