Position: Accountable Deployment of Agentic AI Demands Layered, System-Level Interpretability

Forty-Third International Conference on Machine Learning

Judy Zhu* | Dhari Gandhi* | Ahmad Rezaie Mianroodi | Dhanesh Ramachandram | Sedef Akinli Kocak | Shaina Raza

*These authors contributed equally to this work.

vectorinstitute.github.io/ATLIS

VECTOR INSTITUTE | INSTITUT VECTEUR

DALHOUSIE UNIVERSITY

ICML International Conference On Machine Learning

THE PROBLEM

Model-Centric Interpretability Fails for Agentic Systems

Agentic AI systems plan, invoke tools, update memory, and coordinate over multiple steps. But current methods (i.e. SHAP, saliency maps, circuit analysis, etc.) explain individual model outputs, not how failures emerge from component interactions over time.

1

Co-Design over Reaction

Interpretability must co-evolve with agentic capabilities, not be retrofitted after deployment failures.

2

Layered Decomposition

Agentic opacity occurs at distinct layers: behavioral, mechanistic, coordination, and safety, each needing tailored methods.

3

Lifecycle Integration

Interpretability must operate across the full deployment lifecycle, not as a one-time audit.

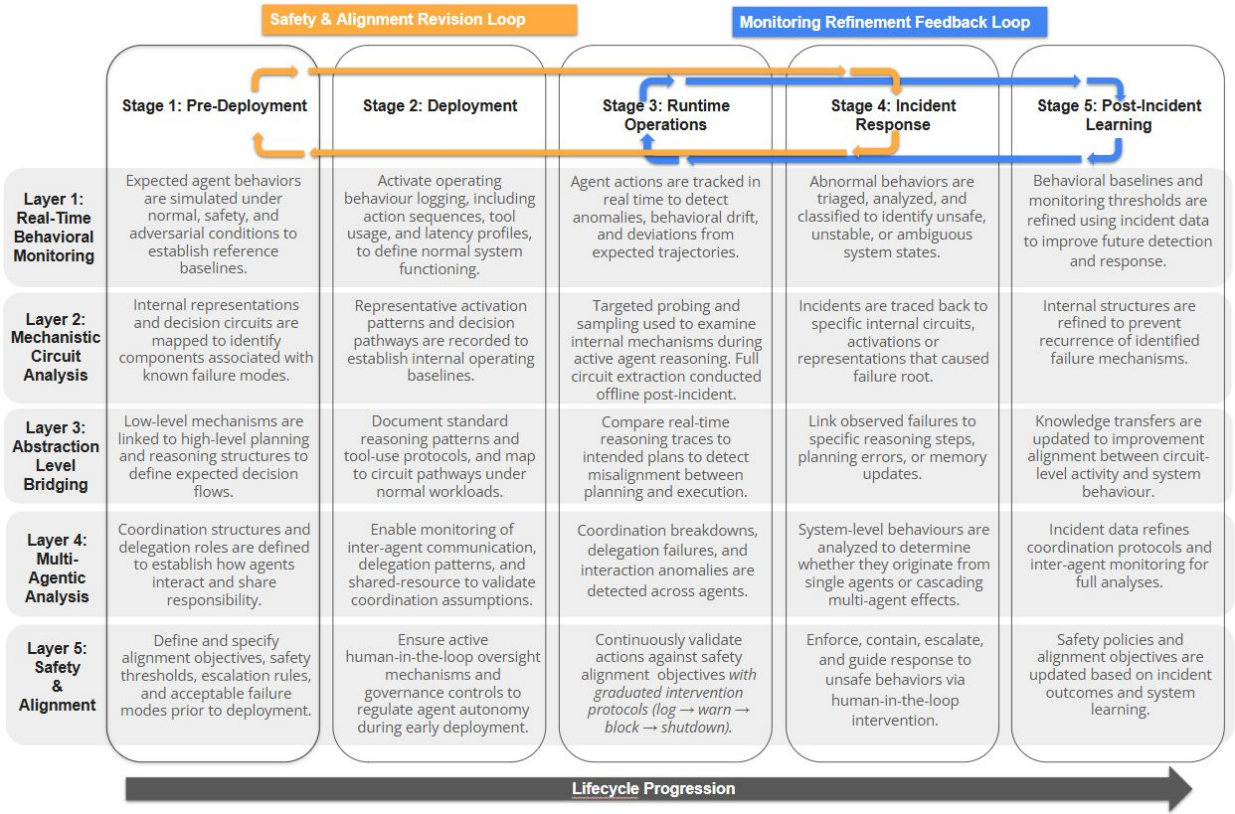
VECTOR INSTITUTE | INSTITUT VECTEUR

DALHOUSIE UNIVERSITY

ICML International Conference On Machine Learning

THE SOLUTION

Framework: Agentic Trajectory & Layered Interpretability Stack (ATLIS)



5 interpretability layers × 5 deployment stages.

This enables lightweight monitoring with risk-aware escalation to deeper analysis.

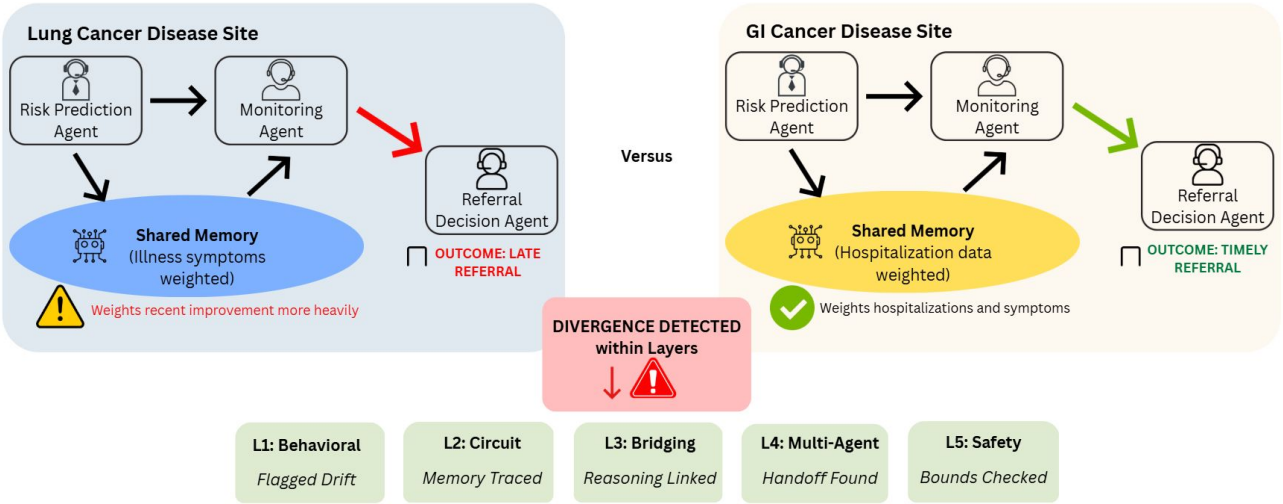
VECTOR INSTITUTE | INSTITUT VECTEUR

DALHOUSIE UNIVERSITY

ICML International Conference On Machine Learning

GROUNDING IN PRACTICE

Case Study: Palliative Care Referral Divergence



Why ATLIS Catches It

L1 Behavioral drift flagged

L2–4 Root cause traced to memory weighting & handoff

L5 Safety bounds checked, clinician escalated

Model-centric methods miss this. The divergence arises from longitudinal component interactions, not a single output.

Researchers

Formalize modular interpretability stacks; build shared benchmarks for agentic failures

Practitioners

Embed traceability at module boundaries; instrument planning, memory, & tool calls

Regulators

Mandate system-level evidence: tracing, lifecycle monitoring, decision provenance