Day 9: Testing & Evaluation

ENSURING QUALITY THROUGH RIGOROUS ASSESSMENT METHODS

Testing & Evaluation Overview

Importance of Testing and Evaluation

Validation of AI Behavior

Testing ensures AI agents behave correctly under diverse conditions including edge cases and high-load scenarios.

Performance Metrics Evaluation

Evaluation measures AI performance with key metrics like accuracy, latency, precision, recall, and user satisfaction.

Detecting Issues and Ensuring Trust

Testing detects bias, compliance gaps, and performance degradation to maintain safety and build user trust.

Continuous Monitoring and Feedback

Structured frameworks use offline and online testing, monitoring, and feedback loops to sustain AI performance.

Built-in Evaluation Framework

Vertex AI Evaluation Tools and Methods

Built-in Performance Metrics

Vertex AI offers essential metrics like accuracy, precision, recall, latency, and throughput to evaluate AI models effectively.

Offline Evaluation

Offline testing uses historical datasets to validate AI models before deployment, ensuring baseline accuracy and reliability.

Online A/B Testing

Online A/B testing compares different model versions in production to measure real-world performance and user satisfaction.

Continuous Monitoring

Continuous monitoring detects data drift, anomalies, and performance degradation to maintain model reliability over time.

Logging and Telemetry

Observability for AI Agents

Logging for AI Systems

Logging captures detailed records of user queries, responses, errors, and system events for analysis.

Telemetry and Performance Metrics

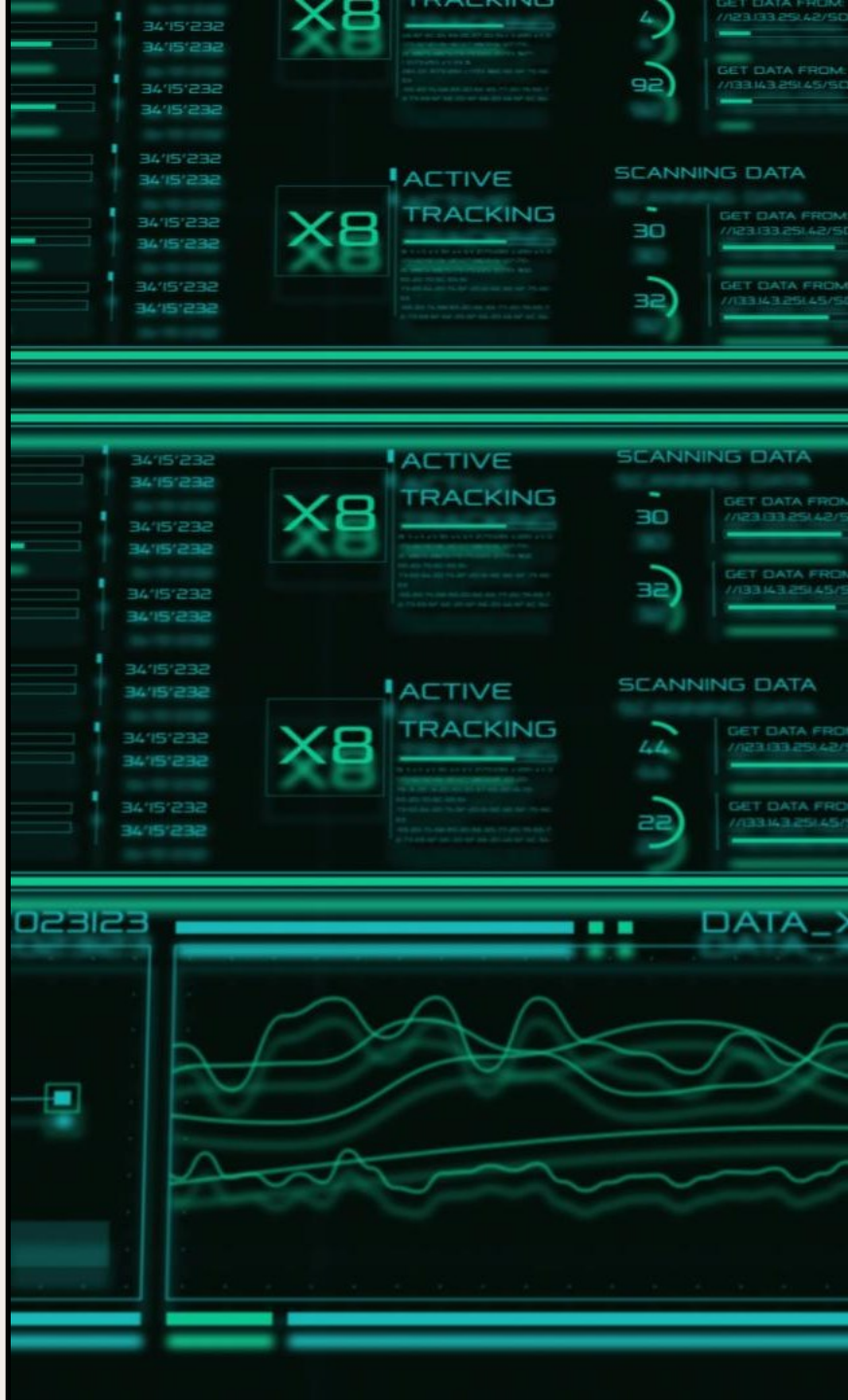
Telemetry collects metrics like latency, resource use, and API call frequency for performance monitoring.

Proactive Monitoring Tools

Integrated tools enable dashboards, alerts, and real-time insights for troubleshooting AI agents.

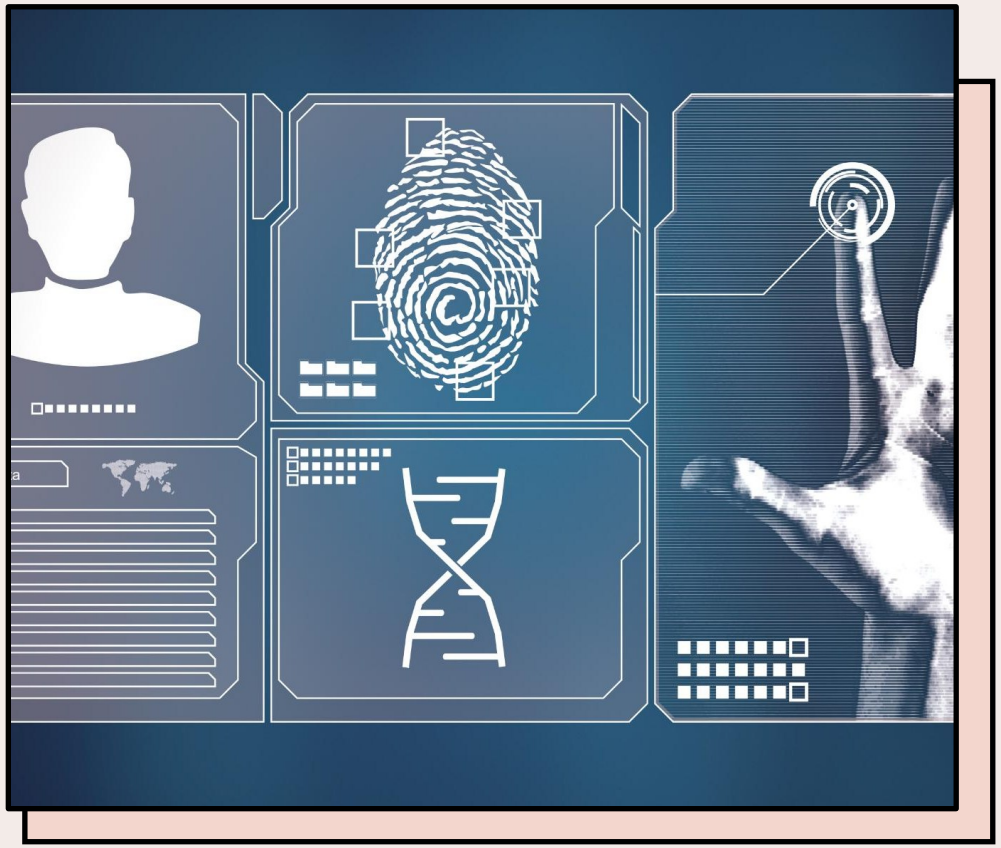
Continuous Improvement

Logs and telemetry data help identify root causes of issues and guide model retraining and adjustments.



Safety Checks and Debugging

Ensuring Responsible AI and Effective Debugging



AI Safety Checks

Safety checks detect and mitigate risks like biased outputs, harmful content, and ensure compliance with regulations.

Debugging Techniques

Debugging identifies and resolves AI performance issues through prompt analysis, error tracing, and configuration fixes.

Explainability Tools

Explainability frameworks like SHAP and LIME provide insights into model decisions for root cause analysis.

Responsible AI Operation

Combining safety checks and debugging ensures ethical AI, maintaining user trust and regulatory compliance.