Cloud Contraptions
TRAINING & CONSULTING

To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

LLM Observability and Cost Engineering

Class Duration

7 hours of live training delivered over 1-2 days to accommodate your scheduling needs.

Student Prerequisites

- Professional software development experience
- Familiarity with LLM API usage and basic application monitoring concepts

Target Audience

Software engineers, platform engineers, and MLOps practitioners responsible for running LLM-powered applications in production. Particularly relevant for teams whose LLM costs are growing unexpectedly, who have poor visibility into what their LLM application is actually doing, or who need to demonstrate cost control to leadership. Teams that also need to measure output quality can pair this with *Evaluating AI Coding Assistants and LLM Apps*.

Description

LLM applications are opaque by default and expensive at scale. This course covers the techniques and tooling to make them observable and cost-controlled. We cover distributed tracing for LLM calls (LangSmith, Langfuse, Braintrust, and OpenTelemetry integration), token counting and budget enforcement, prompt and output caching strategies, prompt versioning and A/B testing, and cost attribution dashboards. Participants instrument a realistic LLM application end-to-end during the labs.

Learning Outcomes

- Instrument an LLM application with distributed tracing across all model calls.
- Calculate and track per-request and per-user token costs accurately.
- Implement prompt caching (exact and semantic) to reduce cost and latency.
- Version prompts and deploy changes with measurable impact tracking.

To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

- Set token budget limits and cost alerts at the application level.
- Build a cost attribution dashboard segmented by feature, user cohort, and model.
- Diagnose performance and cost anomalies from trace data.

Training Materials

Comprehensive courseware is distributed online at the start of class. All students receive a downloadable MP4 recording of the training.

Software Requirements

Python 3.12+ or Node.js 22+, a Langfuse or LangSmith account (free tier), API keys for at least one frontier model, and Git.

Training Topics

LLM Observability Fundamentals

- Why standard APM tools fall short for LLM apps
- Key dimensions: latency, token usage, cost, quality scores
- Trace-first design for LLM applications

Distributed Tracing for LLM Calls

- Instrumenting model API calls with spans and attributes
- Tracing multi-step and agentic pipelines
- Langfuse, LangSmith, and Braintrust integration
- OpenTelemetry for LLM applications (semantic conventions)

Token Counting and Budget Enforcement

- Counting tokens before and after requests
- Per-request, per-session, and per-user budget tracking
- Soft and hard budget limits with graceful degradation
- Token estimation for pre-request validation

Caching Strategies

- Exact prompt caching (provider-native and custom)
- Semantic caching with embedding similarity
- Cache key design for parameterized prompts
- Cache invalidation on prompt version changes

To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

Prompt Versioning

- Treating prompts as versioned artifacts
- Prompt registry: storage, retrieval, and rollback
- A/B testing prompt variants with statistical significance
- Shipping prompt changes safely in production

Cost Attribution

- Mapping token costs to product features and user cohorts
- Cost allocation for shared infrastructure
- Building a cost attribution dashboard
- Forecasting LLM cost at scale

Anomaly Detection and Alerting

- Cost spike detection and alerting
- Latency regression alerting
- Quality score drift detection
- Incident response for LLM production issues

Performance Optimization

- Model downtiering for cost-sensitive paths
- Prompt compression and context trimming
- Batching for throughput-sensitive workloads

