

Table of Contents

Preface.....	xi
Introduction.....	xxi
Part I. Inference	
1. Deploying Models.....	3
“It Works on My Machine”.....	4
Model Server.....	6
vLLM.....	8
Hugging Face Text Generation Inference.....	11
Other Model Servers.....	13
Deploying Models to Kubernetes Manually.....	16
Model Server Controller.....	19
KServe.....	20
Ray Serve and KubeRay.....	27
Lessons Learned.....	30
2. Model Data.....	33
Model Data Storage Formats.....	34
Weight-Only Formats.....	35
Self-Contained Formats.....	37
ONNX.....	41
Safetensors.....	42
GGUF and GGML.....	44
Current State and Gaps.....	45
Model Registry.....	46

Hugging Face Model Hub	48
MLflow Model Registry	49
Kubeflow Model Registry	54
OCI Registry	57
Accessing Model Data in Kubernetes	59
Shared Storage with PersistentVolumes	61
OCI Image for Storing Model Data	65
Modelcars	68
OCI Image Volume Mounts	74
Lessons Learned	76

Part II. Production Readiness

3. Kubernetes and GPUs.....	81
GPU Discovery	82
Node Feature Discovery	83
GPU Feature Discovery	84
Kubernetes GPU Device Plug-Ins	86
GPU Workload Scheduling	87
Label-Based Scheduling	88
Resource-Based Scheduling	91
Dynamic Resource Allocation	92
NVIDIA GPU Operator	95
Operator Configuration with ClusterPolicy	97
Sub-GPU Allocation	98
Multi-GPU Inference	104
Data Parallelism	104
Model Parallelism	106
Single-Node Versus Multinode Inference	110
GPU Resource Optimizations	114
Lessons Learned	116
4. Running in Production.....	119
Model and Runtime Tuning	120
Language Model Evaluation	121
Language Model Compression	123
Model Performance Benchmark	125
vLLM Runtime Parameters Tuning	128
Autoscaling	132
Optimize vLLM Startup Time	136
LLM-Aware Routing	139

From API Gateway to AI Gateway	143
Gateway API Inference Extension	145
Disaggregated Serving	148
Lessons Learned	152
5. Model Observability	153
Observability Stack and Configuration	154
Logs	154
Metrics	156
Tracing	158
Model Server Metrics	160
Time To First Token	161
Time Per Output Token or Inter-Token Latency	161
Throughput	162
Latency	162
Request Queue Metrics	163
GPU Usage Monitoring	164
Quality Metrics	165
Responsible AI	167
Explainability	167
Fairness	168
Model Safety: Hallucination and Guardrails	169
Understanding and Detecting Hallucinations	169
Runtime Guardrails	170
Lessons Learned	175

Part III. Tuning

6. Model Customization	179
Introduction to LLM Creation	179
Prompt and Context Engineering	181
When to Use Model Customization	183
Tuning a Model	184
Fine-Tuning	185
Parameter-Efficient Fine-Tuning	186
Low-Rank Adaptation	187
Running Tuning Jobs on Kubernetes	190
Kubeflow Trainer	192
Other Frameworks	200
Lessons Learned	203

7. Job Scheduling Optimization.....	205
Kubernetes Scheduler Optimization	207
Core Kubernetes Scheduler	207
Resource Bin Packing Strategy	208
Dynamic Scheduling with Descheduler	209
Gang Scheduling	212
PyTorch Rendezvous and Gang Scheduling	213
Comparing Gang Scheduling Solutions	213
Topology-Aware Scheduling	220
Comparing Topology-Aware Scheduling Solutions	223
Quota Management and Multitenancy: GPU as a Service	229
Comparing Quota Management and Multitenancy Solutions	229
Network Optimization for Distributed Training	235
Comparing Network Technologies for GPU Communication	238
Using Secondary Network Interfaces in Kubernetes	243
Bridging HPC and Kubernetes: Slurm and Slinky	248
Storage for Training	249
Training Job Security	250
Security Guidelines for Ray	251
Security Guidelines for PyTorch	254
Observability of Training Jobs	254
Metrics Collection for Distributed Training	255
Logging Across Distributed Workers	256
Tracing Distributed Training Operations	256
Lessons Learned	257

Part IV. AI-Driven Apps

8. AI-Driven Applications.....	261
Architectural Patterns	262
Kubernetes Workload Types	262
Chat Applications	263
Backend AI Services	266
Retrieval-Augmented Generation	274
RAG Components	275
Document Ingestion	278
User Query Processing	281
RAG on Kubernetes	283
Agentic Workflows	286
Agentic Frameworks and Runtimes	290
OpenAI's Responses API	291

Agents on Kubernetes	292
Multiagent Systems	295
Ambient Agents	298
Lessons Learned	300
9. Running Agentic Applications in Production.....	301
The Model Context Protocol	303
MCP Security	305
Agent Impersonation (Token Passthrough)	306
Service Account Delegation	307
Delegated Identity via OAuth2 Token Exchange	318
Mutual TLS with SPIFFE/SPIRE (Zero-Trust)	321
Agent-to-Agent Protocol	329
A2A complements MCP	329
A2A in a Nutshell	330
Running A2A on Kubernetes	331
Agent State Management	332
State Storage Patterns	333
Choosing Between Key-Value Stores and Databases	334
Checkpointing for Long-Running Agents	335
Lessons Learned	336
Afterword.....	339
Index.....	343