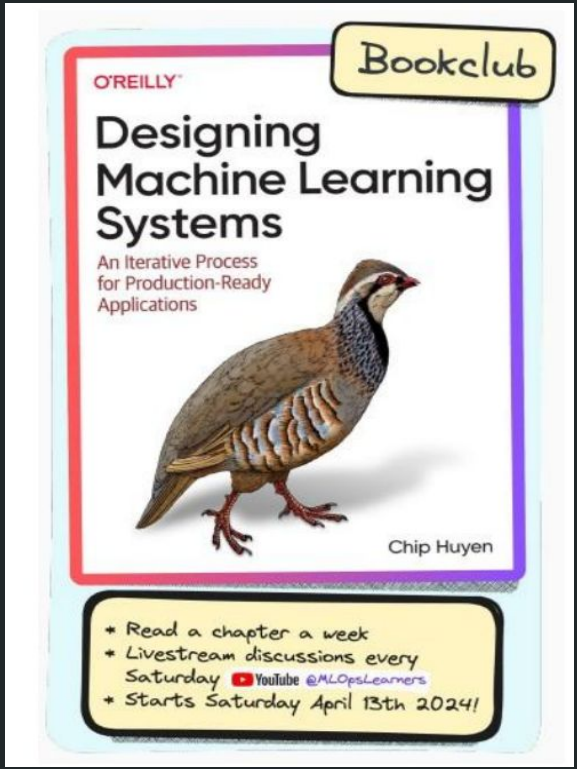


Chapter 10: Infrastructure and Tooling for MLOps

Group 3 (The Rule of Three)

Saturday, June 15, 2024

YouTube

https://www.youtube.com/watch?v=jci0GnWbD3c

Infrastructure and Tooling for MLOps

Components of MLOps:

- Storage and compute considerations
- Development experience
- Efficient resource allocation
- Reliability, observability, automation

2 Contributors

Meet our team - pt.1

Srishti Sehgal

@princessevilsrishti

Staff Engineer @ Okta

Brian H. Hough

@BrianHHough

AWS DevTools Hero

Meet our team - pt.2

Sargun Nagpal

@sargun-nagpal

Hritik Jain

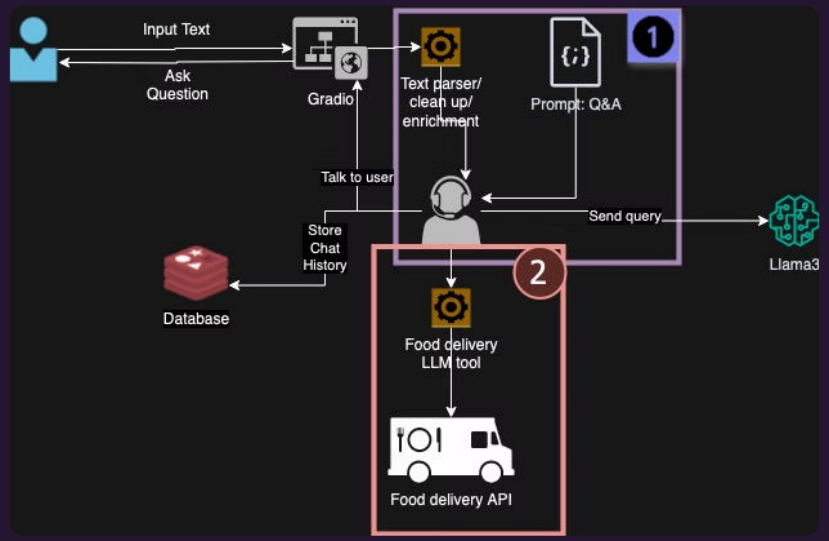
@thehritikjain

Lexie

@lexie1956

Shagun Gupta

@shagunxgupta



Building an Agent…

Built an agent using Langchain. I set up

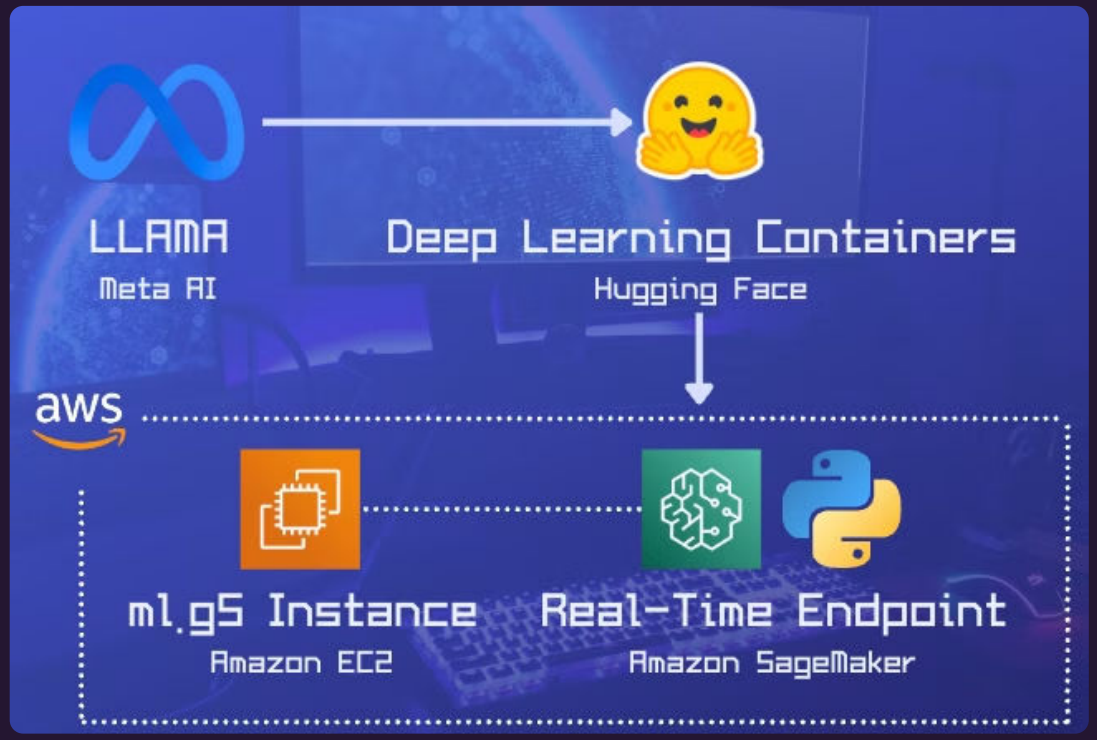
- a Gradio front-end,
- a Redis DB to store chat history,
- Ollama (Llama3) (no API key required!)
- a food delivery API and
- a corresponding custom LLM tool to use it

While the barrier to entry for building with AI has lowered, creating enterprise-grade products and improving such systems is deceptively difficult.

System Architecture Considerations

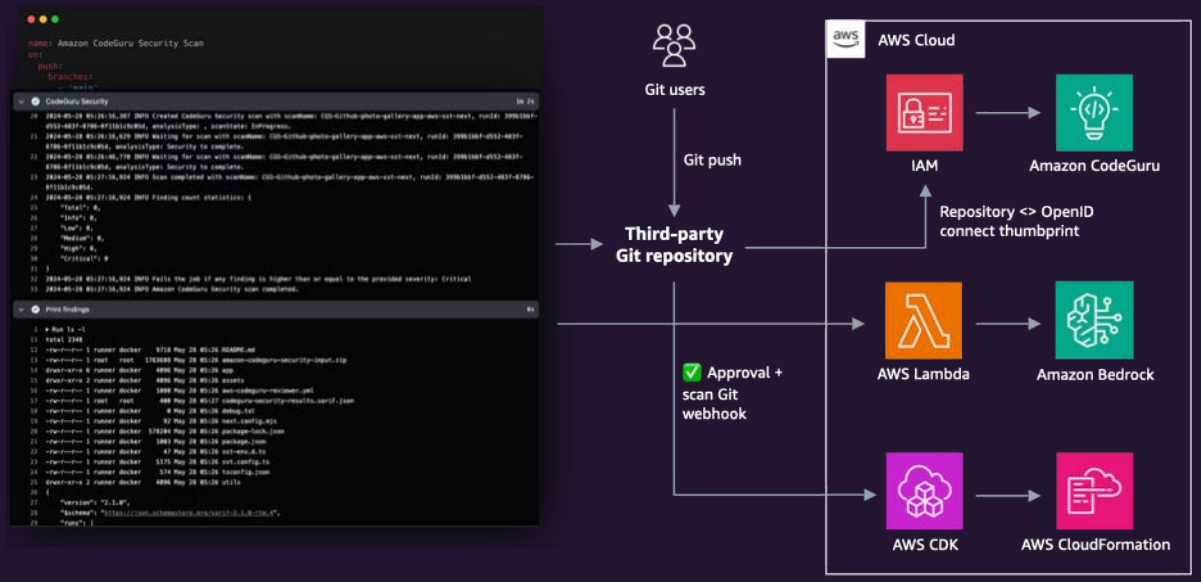
- prompting
  - n-shot prompts, in-context learning, chain-of-thought
  - adding serialization formatting for structure
  - focus on single-task prompts
  - be resourceful but also take a step back
  - consider some other prompting fundamentals e.g. prefilling
  - have automatic migration evaluations/tests!
- user experience
  - product management, developing user journeys
  - guardrails to protect the product/the user
- memory is cool
  - can save cost and eliminate generation latency by removing the need to recompute responses for the same input
- monitoring, performance and quality
  - costs...error management... did I mention costs!?
  - reliability e.g 99.9% uptime? other SLAs?
  - what does good look like? what's a baseline for this system?
- security
- ethical and harmless content generation
- development
  - minimize dev-prod skew

Self-Hosting Models



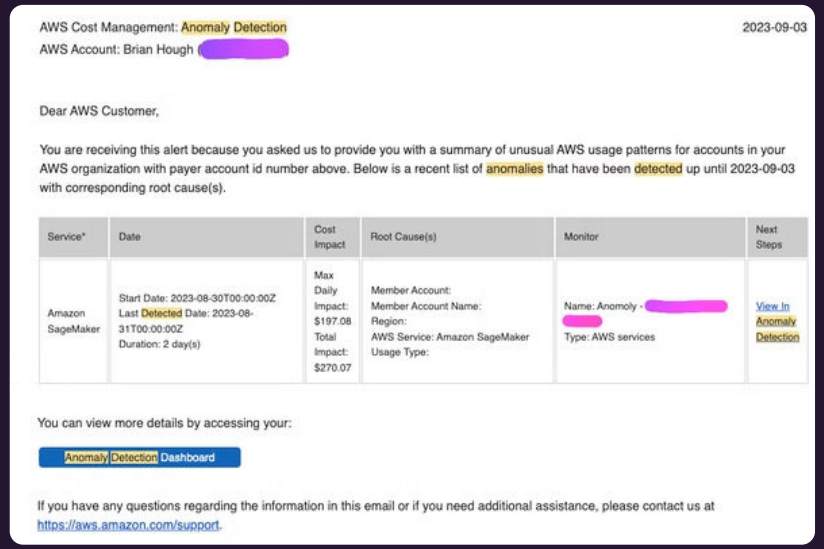
- Model source
  - Hugging Face Deep Learning Containers
- API Endpoint
  - Import and deploy to a real-time endpoint
  - Amazon SageMaker
  - Instances can get expensive if running 24/7!!
- Scalability
  - Async Inference

Serverlessly Utilized Models



- Plug into your workflow
- Increasingly larger and larger token allowances
- Usually much more affordable
- Examples:
  - Amazon Bedrock
  - OpenAI
  - NVIDIA NIM
  - Perplexity

Self-host vs. Serverless



AWS Cost Management: Anomaly Detection
AWS Account: Brian Hough [redacted]

2023-09-03

Dear AWS Customer,

You are receiving this alert because you asked us to provide you with a summary of unusual AWS usage patterns for accounts in your AWS organization with payer account id number above. Below is a recent list of anomalies that have been detected up until 2023-09-03 with corresponding root cause(s).

Service*	Date	Cost Impact	Root Cause(s)	Monitor	Next Steps
Amazon SageMaker	Start Date: 2023-08-30T00:00:00Z Last Detected Date: 2023-08-31T00:00:00Z Duration: 2 day(s)	Max Daily Impact: $197.08 Total Impact: $270.07	Member Account: Member Account Name: Region: AWS Service: Amazon SageMaker Usage Type:	Name: Anomoly - [redacted] Type: AWS services	View In Anomaly Detection

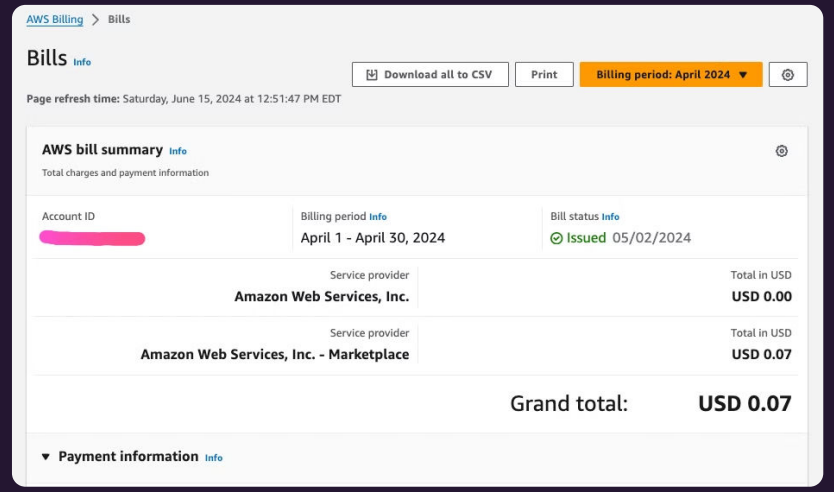
You can view more details by accessing your:

Anomaly Detection Dashboard

If you have any questions regarding the information in this email or if you need additional assistance, please contact us at https://aws.amazon.com/support.

Amazon SageMaker

$270/day (Llama2 + g5.12xlarge)



AWS Billing > Bills

Bills Info

Download all to CSV | Print | Billing period: April 2024

Page refresh time: Saturday, June 15, 2024 at 12:51:47 PM EDT

AWS bill summary Info

Total charges and payment information

Account ID [redacted]

Billing period Info: April 1 - April 30, 2024

Bill status Info: Issued 05/02/2024

Service provider: Amazon Web Services, Inc. — Total in USD: USD 0.00

Service provider: Amazon Web Services, Inc. - Marketplace — Total in USD: USD 0.07

Grand total: USD 0.07

Payment information Info

Amazon Bedrock

$0.07/month (Claude + Claude 3 Haiku)

Book Summary

Storage and Compute: The Foundation

Public Cloud

Benefits:

- elastic
- on-demand access
- well-suited for the bursty nature of ML workloads
- good for scaling/management
- pricey but schemes are available

Private Data Centers

As companies grow, the cost of cloud usage can become significant, prompting a move towards private data centers. This "cloud repatriation" can provide more control and potentially lower costs, but requires substantial upfront investment in hardware and engineering efforts.

Multi-Cloud Strategies

Some organizations adopt a multi-cloud approach, leveraging the best technologies and cost-effective offerings from multiple cloud providers. However, maintaining a cohesive multi-cloud infrastructure can be extremely challenging, often leading to unintentional complexity.

Developing for MLOps: Bridging the Gap

Standardizing Dev Environments

Consistent, standardized development environments are crucial for productivity and troubleshooting. Cloud-based dev environments can help enforce tool and package standardization while allowing developers to use their preferred IDEs.

Containers for Deployment

Container technologies, such as Docker and Kubernetes, play a vital role in bridging the gap between development and production. Containerization ensures consistent environments and streamlines the deployment process.

1

2

3

Notebook Support

Smooth notebook integration is essential for the exploratory nature of ML development. Tools like Papermill, Commuter, and Nbdev can enhance the notebook experience, enabling better version control, sharing, and collaboration.

Managing Resources for ML Workflows

1 Cron, Schedulers, and Orchestrators

Cron handles simple, repetitive tasks, while schedulers manage more complex, interdependent workflows. Orchestrators, such as Kubernetes, focus on provisioning and managing the underlying compute resources to run these workflows.

2 Data Science Workflow Management

Popular workflow management tools for data science include Airflow, Prefect, Argo, Kubeflow, and Metaflow. Each offers unique features and trade-offs, catering to different needs in terms of flexibility, ease of use, and integration with cloud platforms.

The ML Platform: Enabling Scalable Deployments

Model Hosting Services

Model hosting services, such as AWS SageMaker, GCP Vertex AI, and Azure ML, provide the infrastructure to deploy and serve machine learning models in production. These services handle the complexities of scaling, versioning, and monitoring model deployments.

Model Stores

Model stores go beyond just storing model artifacts, also capturing metadata like model definitions, parameters, dependencies, and experiment details. This comprehensive model versioning and provenance is crucial for maintaining and debugging deployed models.

Feature Stores

Feature stores address the challenges of feature management, computation, and consistency across training and inference. They enable teams to discover, share, and reuse features, while ensuring data quality and reducing technical debt.

Build Versus Buy: Weighing the Options

Advantages of Building

Building custom infrastructure can provide more control, flexibility, and cost-optimization, especially as an organization scales. However, this requires significant upfront investment in engineering resources and ongoing maintenance.

Benefits of Buying

Leveraging managed services from cloud providers or third-party vendors can accelerate time-to-market and reduce the operational burden. This is particularly beneficial for early-stage companies or teams with limited engineering resources.

Balancing Tradeoffs

The decision to build or buy ML infrastructure should be based on factors like the organization's scale, technical expertise, and long-term strategic goals. A hybrid approach, utilizing both custom and off-the-shelf solutions, is often the most pragmatic solution.

Conclusion: Unlocking the Full Potential of MLOps

Cloud Infrastructure

Leveraging the flexibility and scalability of public cloud platforms is often the starting point for building robust ML infrastructure.

Containerization

Container technologies like Docker and Kubernetes play a crucial role in bridging the gap between development and production.

Workflow Management

Effective workflow management tools are essential for orchestrating complex, interdependent ML tasks and ensuring reliable execution.

ML Platform

A well-designed ML platform, with components like model hosting, model stores, and feature stores, enables scalable and maintainable model deployments.

Embracing the Future of MLOps

1

Evolving Landscape

The MLOps landscape is rapidly evolving, with new tools, platforms, and best practices emerging to address the growing complexity of machine learning deployments.

2

Continuous Improvement

As organizations continue to invest in and refine their MLOps capabilities, they will unlock greater efficiency, scalability, and reliability in their machine learning operations.

3

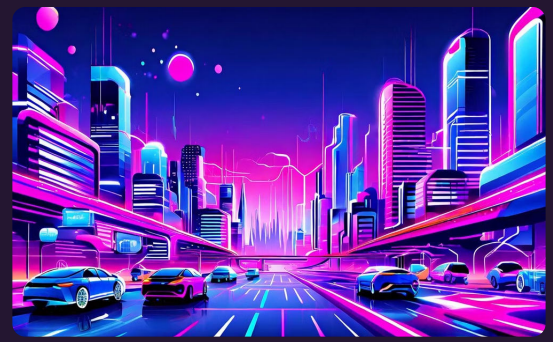
Unlocking Innovation

By establishing a robust MLOps foundation, companies can focus on driving innovation and delivering impactful machine learning solutions that transform their businesses and industries.

The Future is Bright for MLOps

Collaborative Innovation

As MLOps practices mature, data science and engineering teams will work together more seamlessly, fostering a culture of collaboration and accelerating the development of transformative machine learning solutions.



Intelligent Infrastructure

The advancements in MLOps will enable the creation of intelligent, self-optimizing infrastructure that can adapt to the evolving needs of machine learning workloads, driving greater efficiency and scalability.

Celebrating Success

As organizations master the art of MLOps, they will be able to rapidly deploy and maintain high-performing machine learning models, unlocking new levels of innovation and business transformation.