LLMOps

A comprehensive cheat sheet for preparing as an AI Engineer specializing in LLMOps.

Contents

1. Core Technical Skills Development

Programming

Python:

- Essential for ML/AI frameworks, scripting, and automation.
- Refine skills in NumPy, Pandas, PyTorch, and PySpark for data manipulation, modeling, and distributed training.
- **PySpark**: Learn efficient big data processing, including DataFrame APIs and Spark MLlib.
- Build pipelines with PySpark to process large datasets.
- Fine-tune PyTorch-based models.

Bash/Shell scripting:

- For managing cloud environments and CI/CD pipelines.

Cloud Computing (Databricks & Azure)

Gain hands-on experience in:

- **Databricks**: Setting up clusters, using MLFlow for tracking, and training large models.
- **Azure**: Using Azure Machine Learning for model training/deployment, storage solutions, and Kubernetes Service (AKS).
- Learn how to integrate Databricks and Azure for ML pipelines.

Resources:

- Databricks Academy (free learning resources).
- Microsoft Learn platform for Azure ML certifications.

Machine Learning & Deep Learning

Core Concepts:

- Backpropagation, gradient descent, overfitting/underfitting, activation functions.

Libraries/Frameworks:

- PyTorch & TensorFlow (focus on fine-tuning and deploying large models).
- Hugging Face Transformers (model fine-tuning, tokenizers, datasets, pipelines).

Optimization:

- Mixed precision training (NVIDIA Apex).
- Efficient fine-tuning techniques like LoRA, adapters, and prompt tuning.

Large Language Models (LLMs):

- Study LLM architectures like **LLama**, **Mistral**, and related frameworks (**LLamaIndex**, **Langchain**).

- Practice fine-tuning LLMs on domain-specific data using Hugging Face Transformers or PyTorch.
- Explore advanced techniques like **RAG (Retrieval-Augmented Generation)** and **embedding search** for text generation tasks.

Resources:

- Hugging Face Transformers Course.
- LangChain documentation for chaining LLM workflows.
- LLamaIndex tutorials for document-based querying.

Data Engineering

Preprocessing:

- Handling large datasets (Apache Spark, Dask).
- Tokenization and dataset augmentation.

Storage:

- Vector databases: Pinecone, Weaviate, Milvus.
- Efficient storage for embeddings: Redis, Faiss.

- Streaming:

- Kafka, RabbitMQ for real-time data pipelines.

DevOps and MLOps

Containerization and Orchestration:

- Docker (containerizing models).
- Kubernetes (managing LLMs at scale).

Infrastructure as Code (IaC):

- Terraform, AWS CloudFormation.

Continuous Integration/Continuous Deployment:

- GitHub Actions, Jenkins, or GitLab CI/CD.
- Model CI/CD tools: MLflow, Kubeflow.

Monitoring and Logging:

- Prometheus, Grafana.
- Model monitoring: WhyLabs, Weights & Biases, or Neptune.ai.

Automated Pipelines:

- CI/CD for ML pipelines with GitHub Actions or Azure DevOps.
- Automate workflows using tools like Kubeflow and MLflow.

Model Deployment:

- Deploy models using **FastAPI**, **Triton Inference Server**, or **Azure Functions**.
- Optimize inference using **ONNX**, **TensorRT**, or **DeepSpeed**.
- Full Stack Deep Learning (covers MLOps in-depth).
- Kubeflow documentation and tutorials.

2. LLM-Specific Operations

Fine-Tuning & Training

Transfer Learning:

- Pretrained models → Domain-specific fine-tuning.

Parameter Efficient Methods:

- Prefix-tuning, LoRA (Low-Rank Adaptation), BitFit.

Frameworks:

- Hugging Face's Trainer API.
- Accelerate for multi-GPU and TPU setups.

Model Deployment

Inference Optimization:

- ONNX, TensorRT, DeepSpeed-Inference.
- Hugging Face's Optimum library.

Serving Models:

- FastAPI, Flask for lightweight APIs.
- TorchServe, Triton Inference Server for large-scale deployments.

Edge Deployment:

- Quantization (int8/int4) with PyTorch or TensorFlow Lite.

Scaling & Parallelism

Techniques:

- **Model parallelism:** Partition models across GPUs.
- **Data parallelism:** Replicate models and distribute data.
- **Pipeline parallelism:** Split the model into layers for distributed inference/training.

Tools: DeepSpeed, Hugging Face Accelerate, Ray for distributed training/inference.

3. Key Tools for LLMOps

Model Management

- **Weights & Biases (W&B), MLflow:** Model versioning, tracking experiments.
- **Hugging Face Hub:** Storing and sharing pretrained models.

Data Management

- **LangChain:** For chaining LLMs with external tools/data.
- **RAG (Retrieval-Augmented Generation):** Combining embeddings and external knowledge bases.

Vector Search

- **Tools:** Pinecone, Weaviate, Qdrant, or Faiss for embedding lookups.
- **Implementations:** Dense embeddings for semantic search (e.g., sentence-transformers).

Evaluation

Tools:

- Perplexity metrics, BLEU/ROUGE scores.
- Human feedback integration (RLHF pipelines).

4. Cloud & Infrastructure

Cloud Platforms

- AWS: SageMaker, Lambda, ECS, EKS.
- GCP: Vertex AI.
- Azure: Azure Machine Learning.

Key Concepts

- Cost optimization (spot instances, auto-scaling).
- Networking (VPC, Subnets, Load Balancers).
- Security:
    - Secrets management (AWS Secrets Manager, HashiCorp Vault).
    - Role-based access control (IAM).

Serverless Computing

- Lambda, API Gateway for lightweight deployments.
- Batch processing with AWS Batch or Google Cloud Functions.

5. Applied Experience

Key Research Topics

- Efficient LLM fine-tuning (parameter-efficient training methods).
- RLHF (Reinforcement Learning with Human Feedback).
- Optimizing large-scale inference for minimal latency.
- Prompt engineering and prompt optimization.

Big Data & Databases

Data Pipelines

- Construct pipelines using tools like Apache Airflow, Databricks Delta Lake, or Azure Data Factory.
- Learn how to preprocess data in **Parquet**, **JSON**, and other serialization formats for LLM training.

Database Expertise

- SQL/NoSQL: Gain proficiency in managing relational databases (PostgreSQL/MySQL) and NoSQL systems (MongoDB, Cassandra).
- Vector Databases: Explore tools like **Pinecone**, **Weaviate**, or **Faiss** for embedding-based search.

Practice:

- Build a pipeline to load, preprocess, and train models on a large dataset.
- Integrate vector search for semantic queries.

Project Ideas:

Generative AI for Summarization:

- Use LLama or Mistral to fine-tune a text summarization model on domain-specific datasets.
- Deploy on Azure or Databricks for production-scale inference.

Chatbot with RAG:

- Combine LangChain with vector search (Pinecone/Weaviate) to create a chatbot with real-time knowledge retrieval.

Monitoring Dashboard:

- Implement a performance monitoring tool using Prometheus and Grafana to track model performance in production.

End-to-End MLOps Pipeline:

- Build a CI/CD pipeline for fine-tuning an LLM, storing model artifacts in MLFlow, and deploying via Docker containers on Azure Kubernetes.

OpenAI API & Fine-Tuning

- Integrating GPT models via OpenAI's API.
- Fine-tuning OpenAI models:
    - Data preparation guidelines (JSONL format).
    - Hyperparameter adjustments.

Deploy a fine-tuned model as an API.

Create a multi-modal system (e.g., vision + language).

Courses:

- Hugging Face Transformers course.
- Full Stack Deep Learning Bootcamp.
- ML Ops Zoomcamp (free).

Debugging Tips

- Model convergence issues: Check learning rates, gradient clipping.
- Deployment latency: Optimize model size, batch inference.
- Tokenizer mismatch: Verify tokenizer and model alignment.

6. Mindset

Stay curious: LLMOps is evolving rapidly.

- Hands-on practice is critical, experiment with real-world workflows.
- Keep up with research, Explore academic papers on cutting-edge techniques in generative models and NLP evaluation. **Papers with Code**, **ArXiv NLP**, and **OpenAI's blog**.
- Stay updated on **LLM performance optimization** (e.g., quantization, low-rank adapters).

Soft Skills and Communication

- Prepare to document ML workflows and findings for non-technical stakeholders.
- Develop a framework for presenting model results in alignment with business goals.
- Practice collaboration using Agile methodologies, Git version control, and team tools (e.g., Jira, Trello).

Certification Plan

- **Databricks**: Data Engineer Associate or Machine Learning Professional certification.
- **Azure**: Microsoft Azure AI Engineer Associate.
- **MLOps**: TensorFlow AI Engineering Professional Certificate (Coursera).

7. Weekly Roadmap

Week 1-2: Foundations

- Review LLM architectures (LLama, Mistral) and train small-scale models using Hugging Face.
- Complete basic tutorials on Databricks and Azure.

Week 3-4: Hands-On with MLOps & Pipelines

- Set up automated ML pipelines (e.g., Kubeflow, MLflow).
- Build and deploy a small-scale LLM-based app on Azure.

Week 5-6: Advanced LLMOps

- Fine-tune models on large datasets.
- Experiment with scaling and latency reduction techniques.
- Implement a monitoring dashboard for inference.

Week 7-8: Final Prep

- Work on interview-specific projects, such as a chatbot or summarizer.
- Polish presentation and communication skills for technical and non-technical stakeholders.

Resources

Books:

- *Deep Learning with Python* by François Chollet.
- *Designing Machine Learning Systems* by Chip Huyen.

Courses:

- Hugging Face Transformers Course.
- Databricks Academy: Large-Scale ML.

Practice Platforms:

- Kaggle (competitions focusing on NLP).
- GitHub for collaborative projects.