Full-stack Machine Learning

Alexander D’hoore

Full-stack Machine Learning

- Introduction to MLOps
- Comparison to DevOps

- Experiment Tracking
- **Lab 1**

- Data Engineering
  - => **you are here**
- **Lab 2**

- Model Deployment
  - Cloud, edge
  - Prototyping
- **Lab 3**

- Automation
  - Agents, pipelines
  - Hyperparameters
- **Lab 4**

Introduction to Labeling Tools

What are Labeling Tools?

- **Definition**: Labeling tools are software / platforms designed to assist in the **annotation of data**, which is crucial for supervised machine learning models.

- **Purpose**: They help in assigning **meaningful labels to raw data**, enabling machine learning algorithms to learn and make predictions accurately.

- **Importance in MLOps**: Ensures the creation of **high-quality labeled datasets**, which is a critical step in the MLOps pipeline.

Why is Data Labeling Important?

- **Training Accuracy**: High-quality labeled data improves the accuracy and performance of machine learning models.
- **Data Consistency**: Ensures consistent and precise labels across the dataset.
- **Efficiency**: Streamlines the process of preparing data for machine learning, saving time and resources.
- **Scalability**: Facilitates handling large volumes of data required for training robust models.

Types of Data Labeling Tools

- **Manual Labeling Tools**: Require **human annotators** to label data manually.
- **Automated Labeling Tools**: Use AI and machine learning to **automatically label** data with minimal human intervention.
- **Hybrid Tools**: Combine manual and automated processes to balance accuracy and efficiency.

Popular Data Labeling Tools (1)

- **Label Studio**
  - Features: **Open-source** data labeling tool
  - Supports text, images, audio, video, and time series
  - Use Cases: Computer vision, NLP, and more.

- **Labelbox**
  - Very popular
  - Features: Collaborative platform, supports various data types, integrates with ML workflows.
  - Use Cases: Computer vision, NLP, and more.

Popular Data Labeling Tools (2)

- **SuperAnnotate**
  - Features: Advanced annotation tools, **AI-assisted labeling**.
  - Use Cases: Image and video annotation, medical imaging.

- **Scale AI**
  - Features: High-quality annotations, API integration, supports diverse data formats.
  - Use Cases: Autonomous vehicles, e-commerce, robotics.

Data Versioning

What is Data Versioning?

Data versioning involves **tracking changes** to datasets,

ensuring **reproducibility and traceability**

in machine learning workflows.

Why use Data Versioning?

- **Reproducibility**: Enables reproducing experiments with the same dataset versions
- **Data Lineage**: Provides insights into how datasets evolve over time and are used in different experiments
- **Collaboration**: Facilitates sharing and collaboration on datasets among team members

The old way: Databases

- Production data is often stored in databases
  - SQL/relational databases, like PostgreSQL, SQL Server…
  - NoSQL databases like MongoDB, Redis…
- Data is **modified in-place**
  - Old data is changed, removed, updated
  - What was the database like yesterday?
  - Which data was the model from last week trained on?
- We need something better!

Use what we know: Git

- As software developers we know Git
- Could we use Git to store our data?

- **Yes**: technically it might work

- **No**: Git doesn't handle large data well
  - Git was made for source code
  - Becomes slow with larger datasets

Existing Tools: DVC (Data Version Control)

- DVC is an open-source tool for managing machine learning projects and data versioning.

- Enables **versioning large datasets** efficiently by storing metadata and using Git for version control.
  - **Git stores the metadata** to track versions
  - But the data itself is stored outside Git

- Integrates with existing ML workflows and cloud storage providers.
  - We can use this together with MLflow, ClearML…

Existing Tools: LakeFS

- LakeFS is a **versioned data lake** for managing large-scale datasets with built-in version control.
- Provides a Git-like interface for versioning data objects and managing metadata.
  - It doesn’t actually build on Git
  - But it gives you commands that look like Git
- Offers data consistency and integrity guarantees for collaborative data workflows.
  - Can replace the production database (DVC can’t)

Existing Tools: ClearML (again)

- ClearML extends its capabilities to include **data versioning and management** alongside experiment and model tracking.
- Allows **versioning and tracking of datasets** used in machine learning experiments.
- Provides integration with various data storage solutions and cloud providers.
  - Amazon AWS, Google GCP, Microsoft Azure

vives University of Applied Sciences

Lab 2: Data Engineering

Lab 2: Data Engineering

- We will learn **ClearML Datasets**

- Download exercises from Github
- Exercises will be **added 1 by 1**

https://github.com/AlexanderDhoore/240603-mlops-workshop