Effective MLOps
Model Development

Introduction

ML is great, but many ML projects today still fail

Scoping

Project not suitable for ML
Value doesn't justify cost
Lacking trust or adoption

Scaling

Difficulty going from single to multiple models
Lack of testing

Development

Messy/manual processes
Key person dependencies
Communication overheads

Deployment

Performance requirements
Operational costs

Data

Cost/quality of labelling
Feature Engineering
Data availability

Model/Data Drift

Dealing with changes

Effective MLOps addresses these problems

- **Opportunity Identification**
- **Training Operationalization**
- **Model Development**
- **Model Deployment**
- **Data Labelling and Management**
- **Production Monitoring**

Focus of this course

Model Development

Focus of This Course

Model Development

Build An End-To-End Prototype

June 28, 8AM PT/ 5PM CET

Optimize Your Solution

July 5, 8AM PT/ 5PM CET

Model Evaluation

July 12, 8AM PT/ 5PM CET

July 19, 8AM PT/ 5PM CET - Final Project Presentations

W&B Developer ML Ops Platform

W&B PLATFORM

Experiment tracking	Dataset versioning	Model evaluation	Model optimization	Collaborative analysis
Experiment	Artifacts	Tables	Sweeps	Reports

FRAMEWORK AGNOSTIC

PyTorch · Lightning · TensorFlow · Keras · Hugging Face · fast.ai · Scikit · XGBoost

ENVIRONMENT AGNOSTIC

Cloud hosted + On-premises

AWS · Azure · Google Cloud · Kubernetes

Our Goals For This Course

Accelerate
Your model development via principled workflows

Best practices
for collaboration

Enterprise-grade
reproducibility and governance through data and model lineage tracking

Better Models
through better insights from experiments

Improve Productivity
with automation

Never Lose Track
of your work with experiment management

How To Engage With the Course

Join our discord **#course** channel:
wandb.me/discord

Form a **team** or work as individual

Choose a **project** and a **dataset** to work on

Complete assignments

Get W&B course **certificate**

Effective MLOps
Model Development

Lesson 1 - Building An End-To-End Prototype

Agenda - Building an End-to-End Prototype

Understand the Business Context

Frame the Data Science Problem

Explore & Understand Your Data

Establish Baseline Metrics & Models

Communicate Your Results

Tables

Artifacts

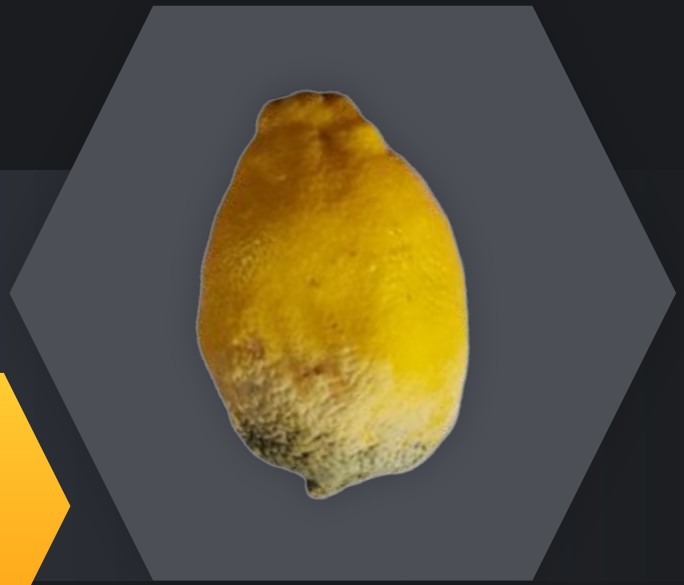
Experiments

Reports

Case Study - Lemons Turning Sour

Successful lemonade franchise running into problems due to quality of lemons being shipped directly to stands

Case Study - Lemons Turning Sour



No central quality control but supplier offered to send photos of lemons before shipping them

They shared with us their labeled dataset of lemon photos

They will replace a shipment if we notify them within 5 minutes of receiving the photos with legitimate concerns

W&B Tables

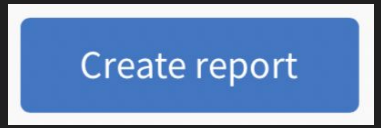
- Visualize and analyze model predictions
- Centralize exploratory data analysis
- Quickly spot check rows from your dataset

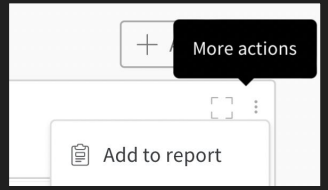
```
# Quickly log your first table

wandb.log({“table”: my_dataframe})
```

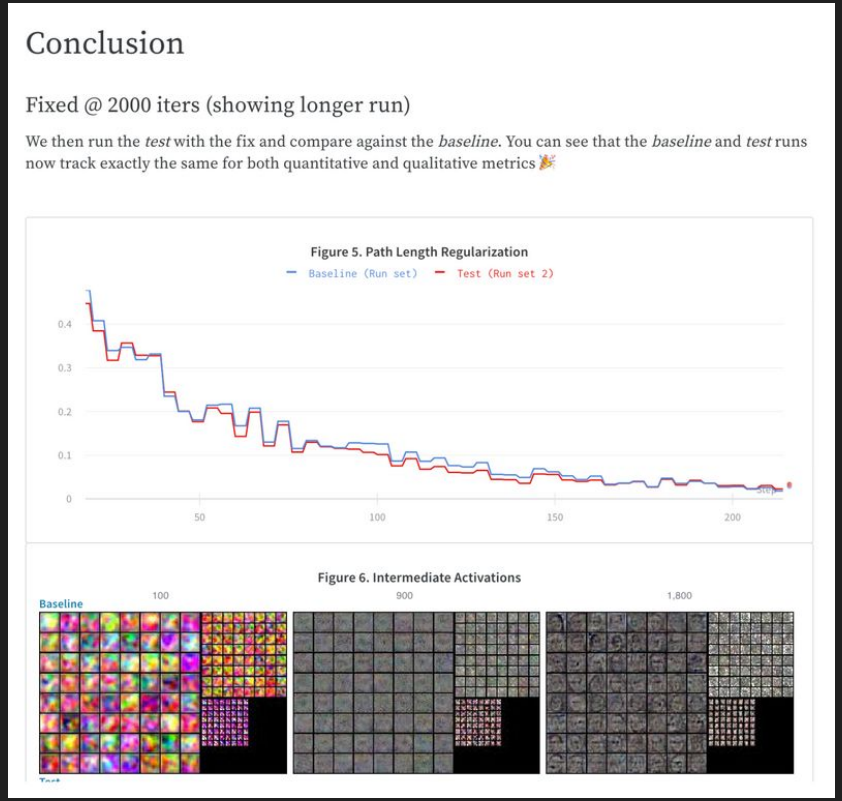
raw	annotated	color_mask	% human	% motor vehicle ↓
			0	0.6238
			0	0.4169
			0.0002767	0.4047
			0.005658	0.3953
			0.01103	0.3882

W&B Reports

- Collaborative analysis in live dashboards
- Share with your coworkers
- Make live comments, describe your findings, and take snapshots of your work
- Export as a LaTeX zip file or convert the file to PDF



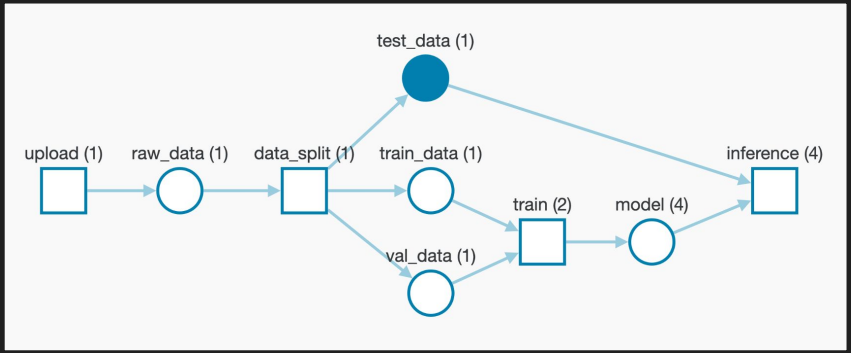




W&B Artifacts

- Lightweight dataset and model versioning with deduplication
- Save every step of your pipeline
- Model tracking and model lineage
- Effortless observability
- Data Access controls

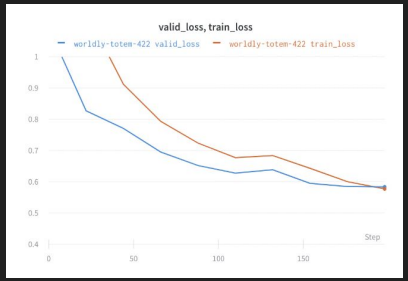
```
# Log an artifact
artifact = wandb.Artifact('mnist',
type='dataset')
artifact.add_dir('mnist/')
wandb.log_artifact(artifact)

# Use artifact in your pipeline
artifact = run.use_artifact(mnist:v1')
artifact_dir = artifact.download()
```

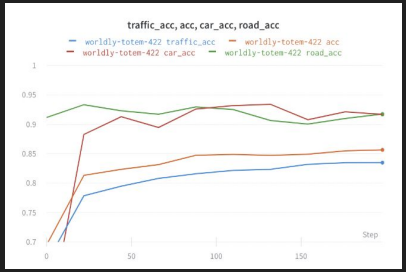


W&B Experiments

- A system of record for your model training
- Visualize and compare every experiment
- Quickly find and re-run previous model checkpoints
- Monitor your compute
- Debug performance in real time

```
# Integrate with any Python script
import wandb

# 1. Start a W&B run
wandb.init(project='gpt3')

# 2. Save model inputs and hyperparams
config = wandb.config
config.learning_rate = 0.01

# Model training here

# 3. Log metrics over time to visualize performance
wandb.log({"loss": loss})
```





Assignment 1

1. **Pick** a problem and dataset
2. **Log** dataset as an Artifact
3. **Visualize** data with a Table
4. **Develop** a simple baseline and log it as an Experiment
5. **Share** your baseline result via a Report in *#course* discord channel (wandb.me/discord)

Dataset Suggestions

- ❑ Anything you're already working on
- ❑ Previous Kaggle competition kaggle.com/competitions
- ❑ Lemon-dataset github.com/softwaremill/lemon-dataset