Course Introduction

Sprint 0 - Week 0

INFO 9023 - Machine Learning Systems Design

2024 H1

Thomas Vrancken (t.vrancken@uliege.be)
Matthias Pirlet (matthias.pirlet@uliege.be)

Agenda

What will we talk about today

1. Introduction to the staff
2. Introduction to ML Systems Designs & MLOps
    a. What & Why
3. Course organisation
    a. Goal
    b. Culture / guidelines
    c. Roadmap
    d. Practicals

Introduction to the staff

LIÈGE université

Introduction to the staff

Thomas Vrancken

(Instructor)

t.vrancken@uliege.be

Matthias Pirlet

(Teaching assistant)

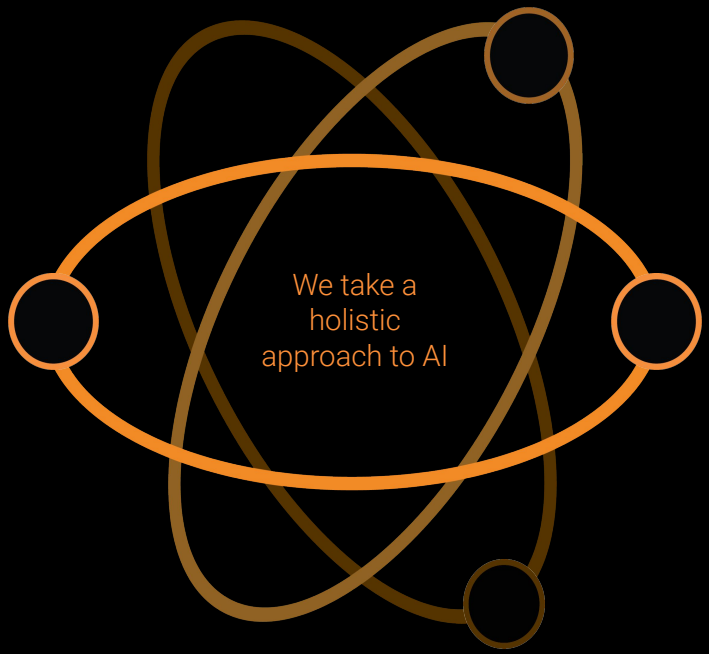
matthias.pirlet@uliege.be

ML6 - your partner for AI

We accompany organisations through the entire AI journey
Use case ideation & assessment
Designing, building and deploying solutions
Managed services for support and maintenance
Further scaling and evolving solutions

We help remove all barriers to technology adoption
Security
Ethics & Regulation
Business case building
Selecting the right tech stack
Facilitating user adoption

We cover all AI domains
Machine Vision
NLP
Structured Data
Reinforcement Learning & Generative AI
MLOps & Engineering best practices



We build bespoke ML solutions
Solution tailored to complex client needs
Agile development, accelerated through the use of boiler plates
Reliable, robust & easily maintainable solutions

We deliver end-to-end solutions
Data labelling
Sourcing of internal and external data
Selecting the right hardware (incl. cameras, sensors, edge devices)
Front-end development
Integration, with traditional machines and robots

We are open source minded
Tech radar for stack selection
Hybrid cloud - on premise; and edge deployment

ML6

ML6 at a glance.

One of the largest and fastest growing AI business & engineering teams in Europe since 2013.

EXCEPTIONAL TALENT & SKILLS

110+ experts spread over **3 different EU** locations.

Talent magnet: 12 new applicants each day

Security, Legal and **Ethical** AI experts

SOME MORE STATS

17% of time in R&D,
250+ publications

Multiple **awards** (AI innovator of the year DataNews 2022, finalist scale up of the year EY 2023, among 8 companies to watch in BE Financial Times, …)

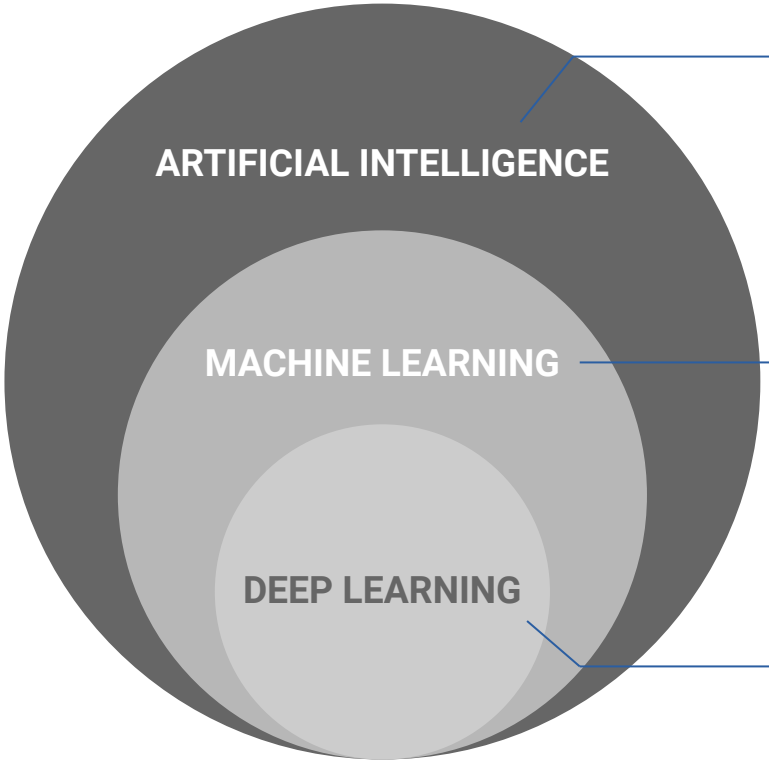
Partner with **AWS**, **Microsoft**, **Google Cloud** & **Cohere**.

ML6

General introduction to ML Systems Design & MLOps

LIÈGE université

AI vs ML vs DL



ARTIFICIAL INTELLIGENCE

Ability of a machine to perform cognitive functions we associate with human minds, such as perceiving, reasoning, learning, problem solving, and even creativity

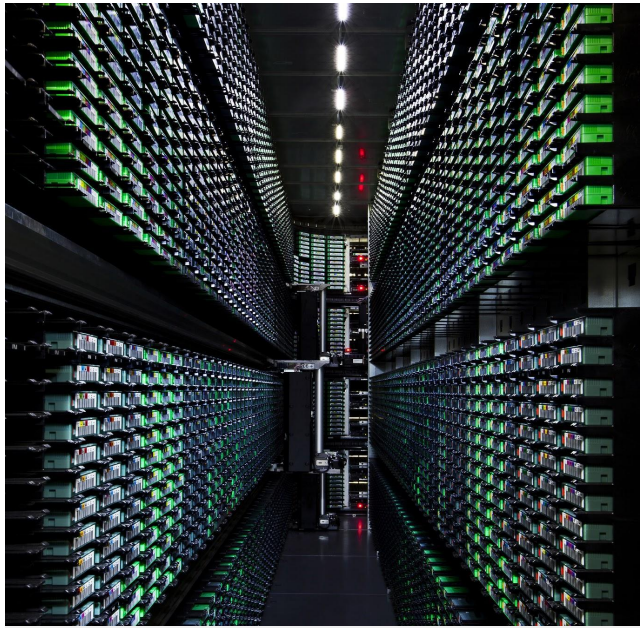
MACHINE LEARNING

AI techniques that give machines the ability to learn from data without being explicitly programmed, i.e. to automatically improve through experience

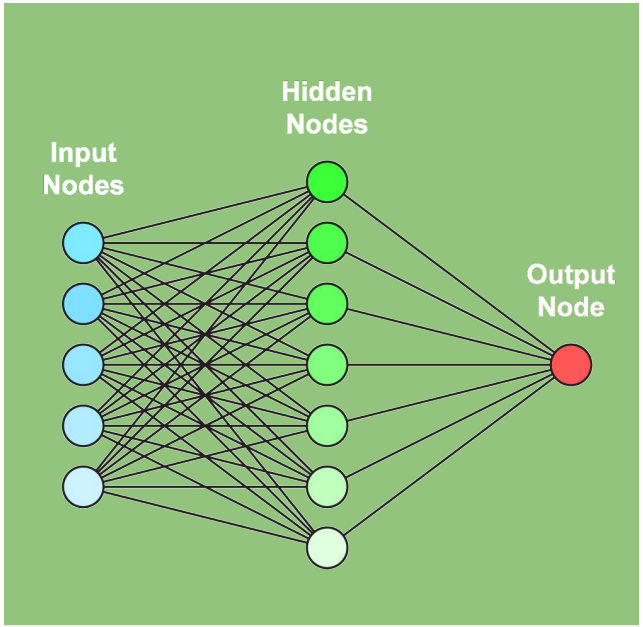
DEEP LEARNING

Type of Machine Learning built upon the concept of interconnected layers known as “neurons” that form a neural network.

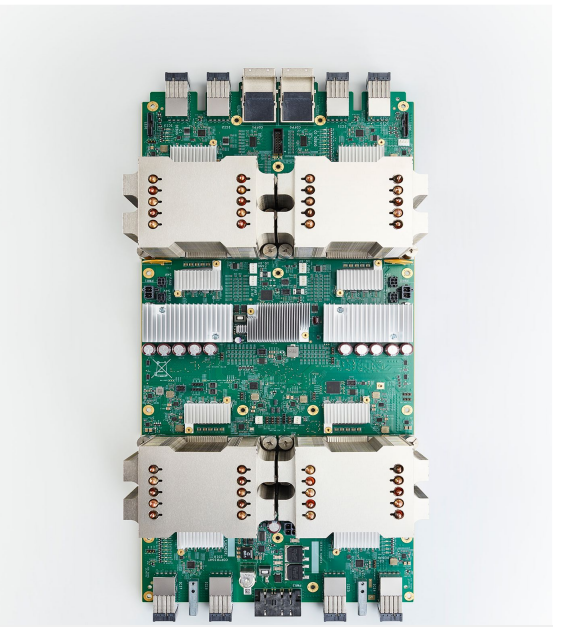
Why now ?



Large Datasets



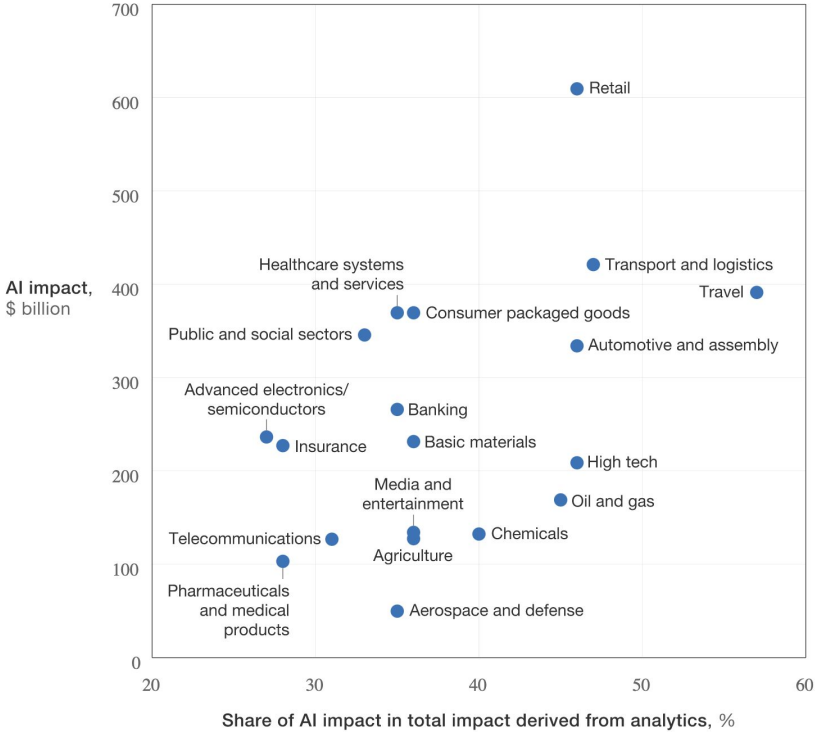
Better Models



Lots of Computation

AI is bringing a revolution in many different industries

Artificial intelligence (AI) has the potential to create value across sectors.



AI value creation by 2030

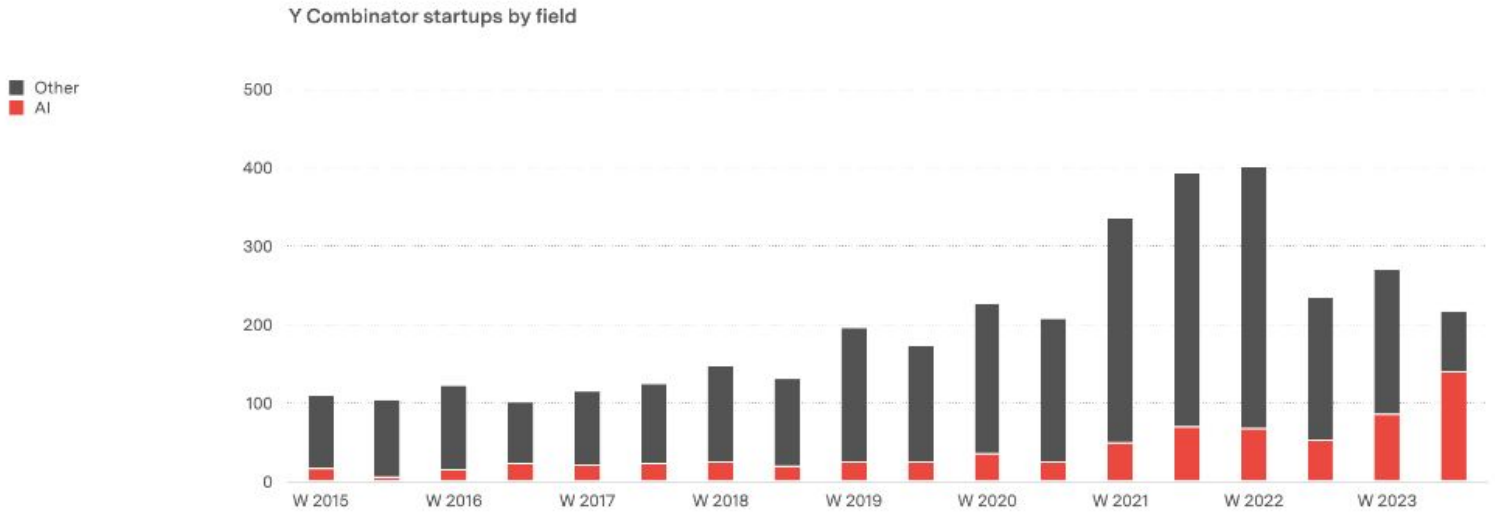
13 trillion USD

Most of it will be outside the consumer internet industry

CS 329S: Machine Learning Systems Design - Stanford (Chip Huey)
https://stanford-cs329s.github.io/syllabus.html

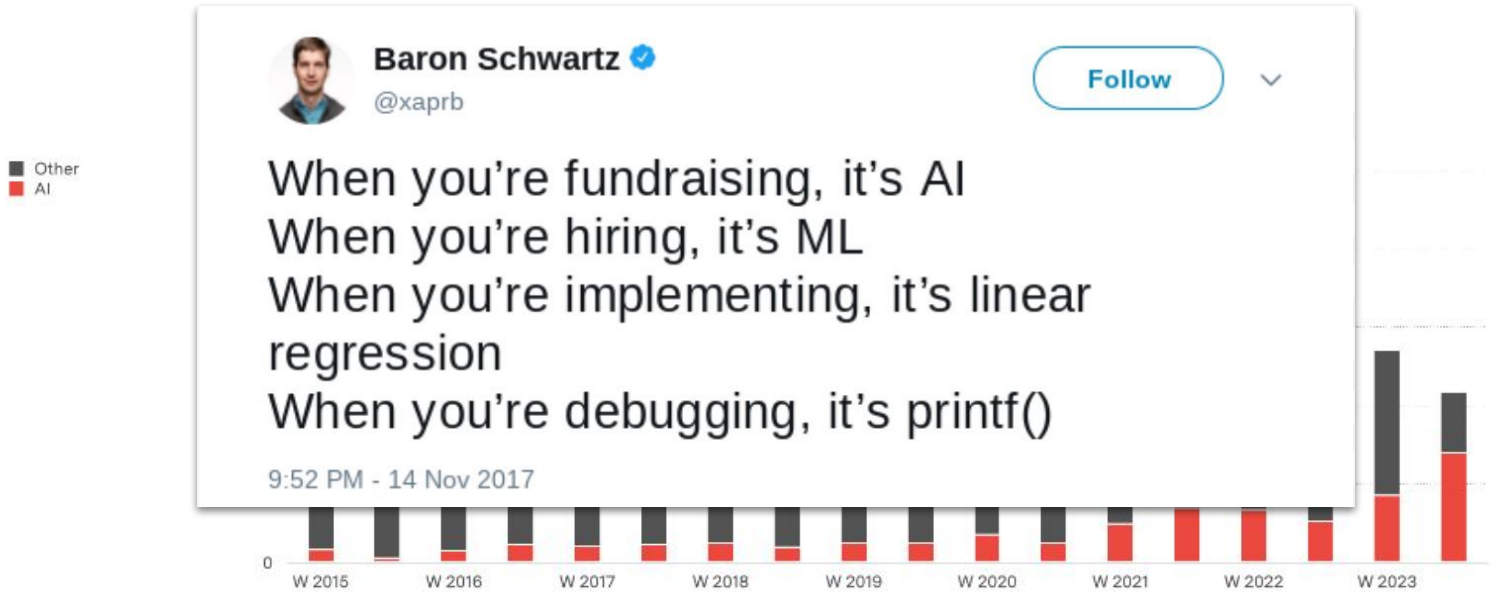
McKinsey&Company | **Source:** McKinsey Global Institute analysis

Investment in AI ventures is skyrocketing.



Source: Y-Combinator // Shared in: https://www.ben-evans.com/presentations

Investment in AI ventures is skyrocketing.



Source: Y-Combinator // Shared in: https://www.ben-evans.com/presentations

AI is everywhere!

A few example of ML applications.

Facial recognition

Product recommendation

Email spam filtering

Autocomplete

Finance predictions

Healthcare imaging

Weather forecast

...

Why do we need ML Systems Design?

Building a ML application means implementing much more than just your ML model.

INFO 9023 - Machine Learning Systems Design

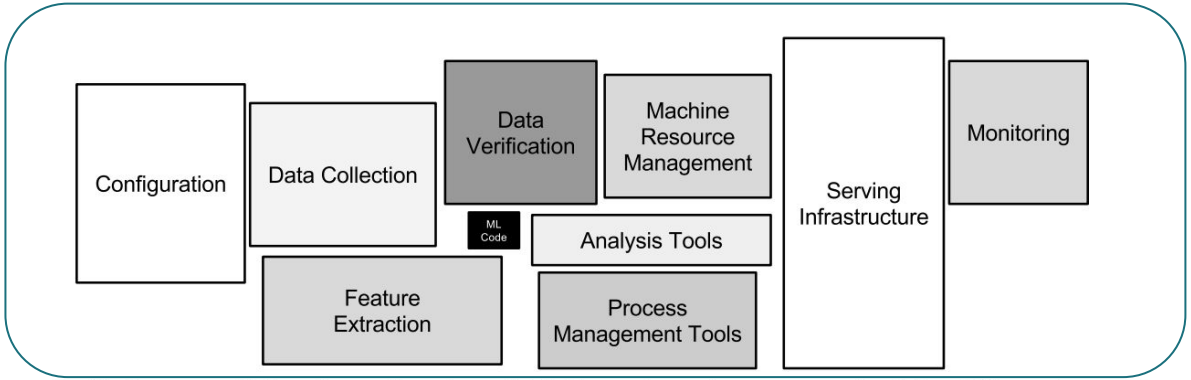
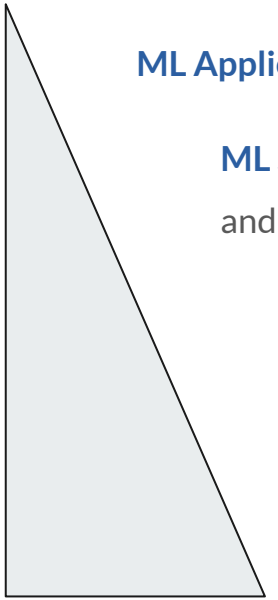


Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

Sculley, D. et al. (2015). Hidden technical debt in machine learning systems. https://papers.nips.cc/paper_files/paper/2015/hash/86df7dcfd896fcaf2674f757a2463eba-Abstract.html

Important definitions



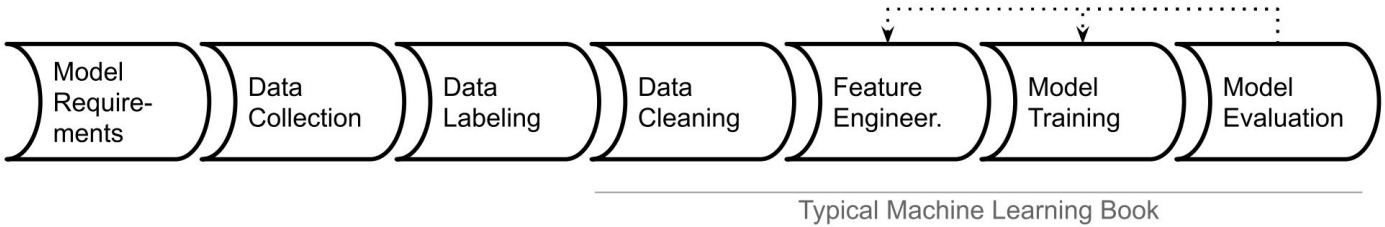
ML Application: The final solution or program powered by a Machine Learning model.

ML System: All the components responsible for the implementation and management of the data and models powering an ML application.

ML Systems Design: The act of designing the architecture and implementing an ML System.

MLOps: Set of practices that aim at implementing and maintaining ML systems in production reliably and efficiently.

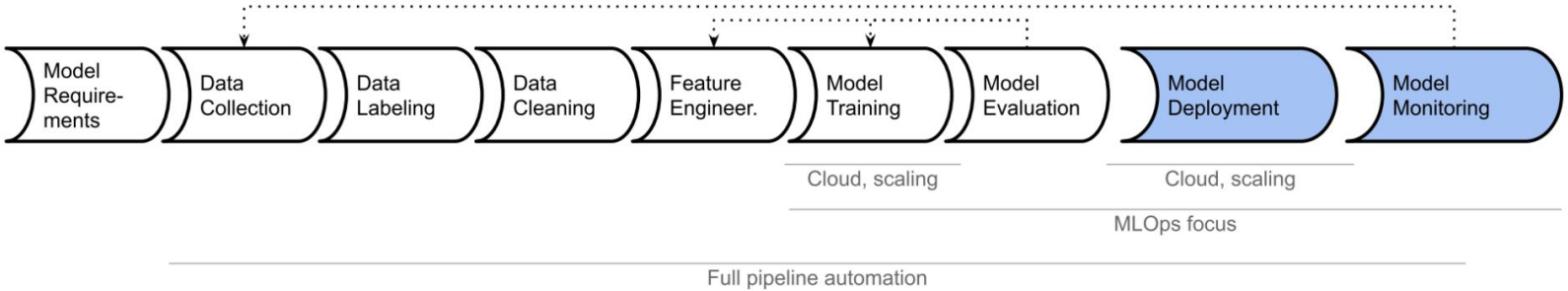
Traditional data science



Output:

- Jupyter Notebook
- Single model working on static dataset

MLOps - Fully automated pipeline



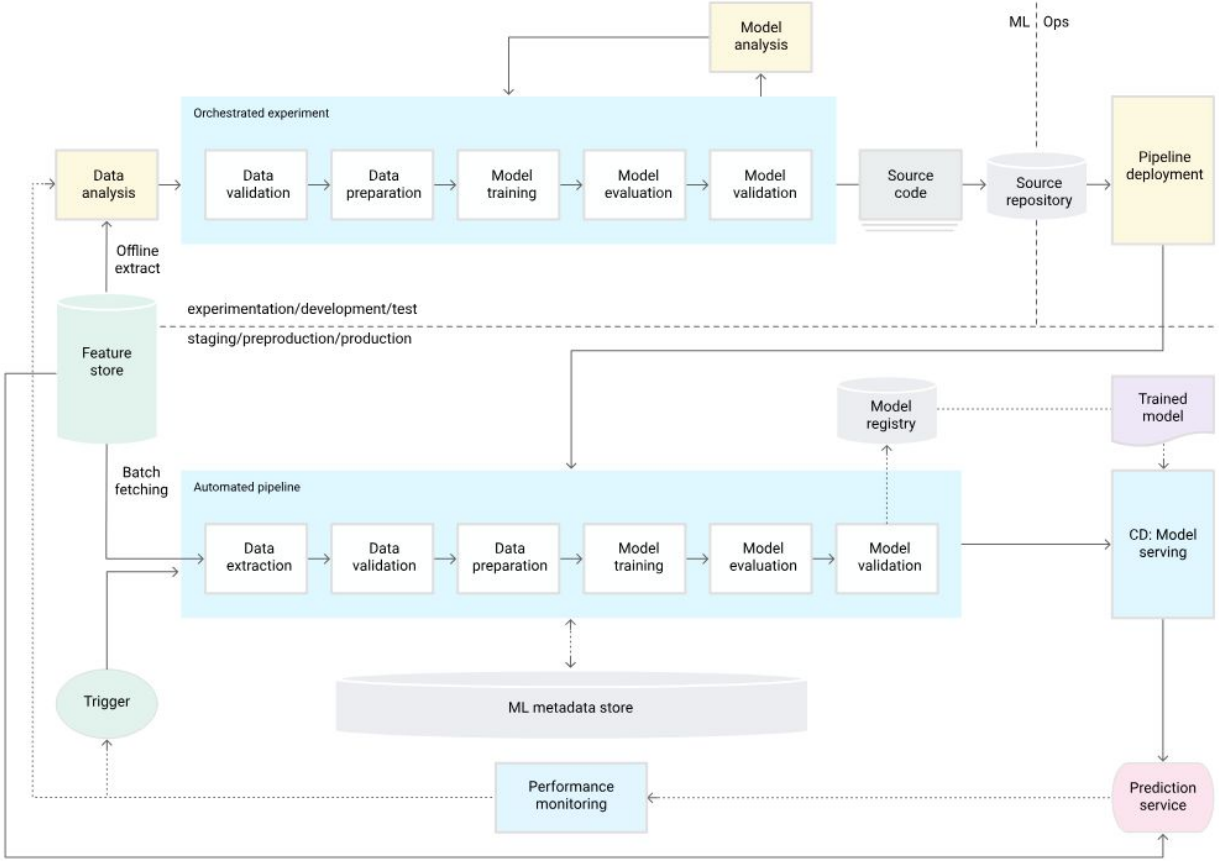
Output:

- Deployed model (e.g. API in the Cloud)
- Monitor live model performance
- Directly connected to data source
- Fully automated pipeline to train and deploy new models
- ...

Machine Learning in Production / AI Engineering - Carnegie Mellon University (Christian Kästner)
https://github.com/ckaestne/seai/tree/F2022/lectures/02_systems

Key concepts of ML Systems Design

Typical architecture of an ML system



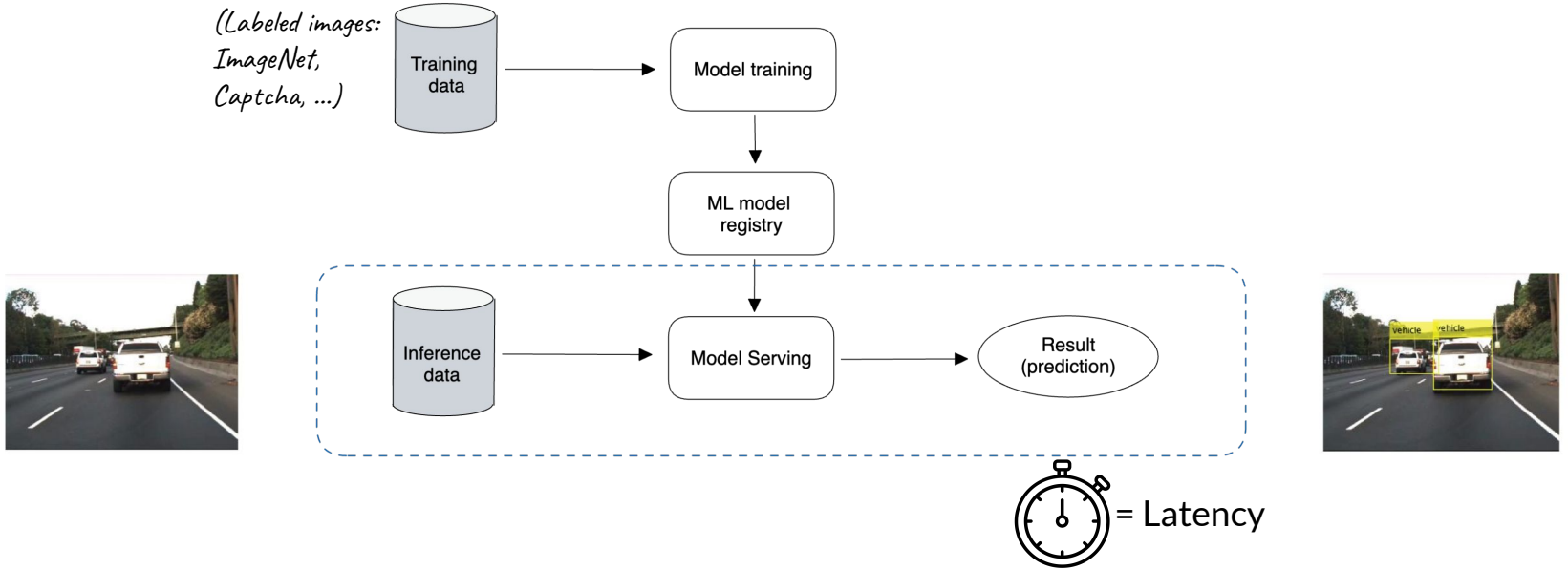
https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

Key concept: Data preparation

It all starts with data. How to go through all these steps efficiently and effectively.

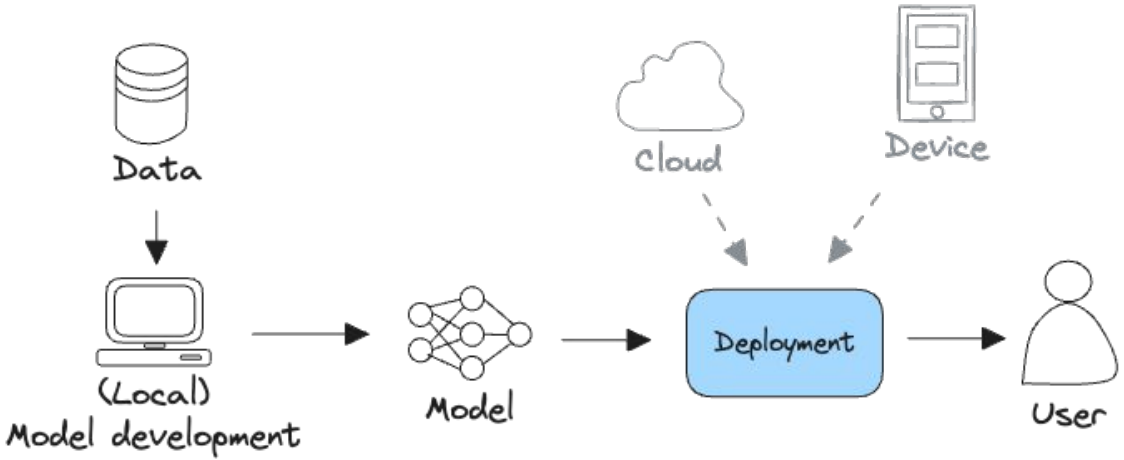
Key concept: ML model serving

How to efficiently serve ML model to client.



Key concept: ML model deployment

How to efficiently deploy your model for serving.



Key concept: Containerisation

Containers encapsulate an application as a **single executable package** that contains all the information to **run it on any hardware**:

- Application code
- configuration files
- libraries
- dependencies

Abstracts the application from its **host operating system.**

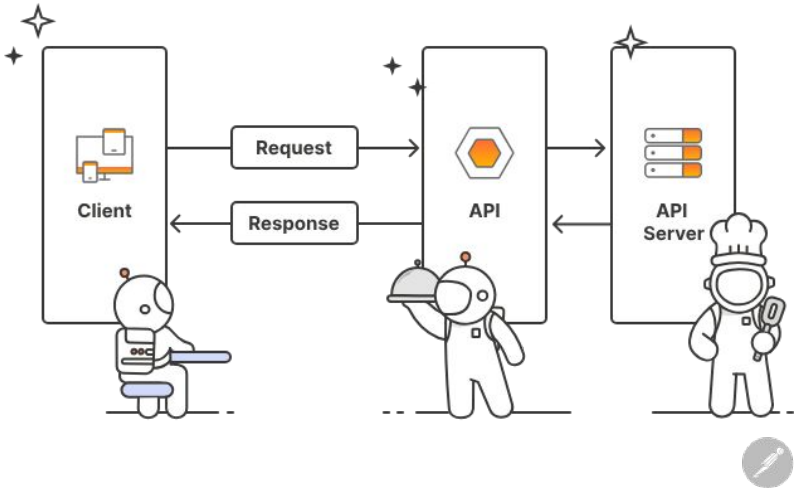
Containers can be easily transported from a desktop computer to a virtual machine (VM) or from a Linux to a Windows operating system, and they will run consistently on virtualized infrastructures or on traditional “bare metal” servers, either on-premise or in the cloud.

LIÈGE université

https://www.ibm.com/topics/containerization

Key concept: APIs

Allow other services to call your model or application.



An **Application Programming Interface (API)** is a set of protocols that enable different software components to communicate and transfer data.

Developers use APIs to bridge the gaps between small, discrete chunks of code in order to create applications that are powerful, resilient, secure, and able to meet user needs.

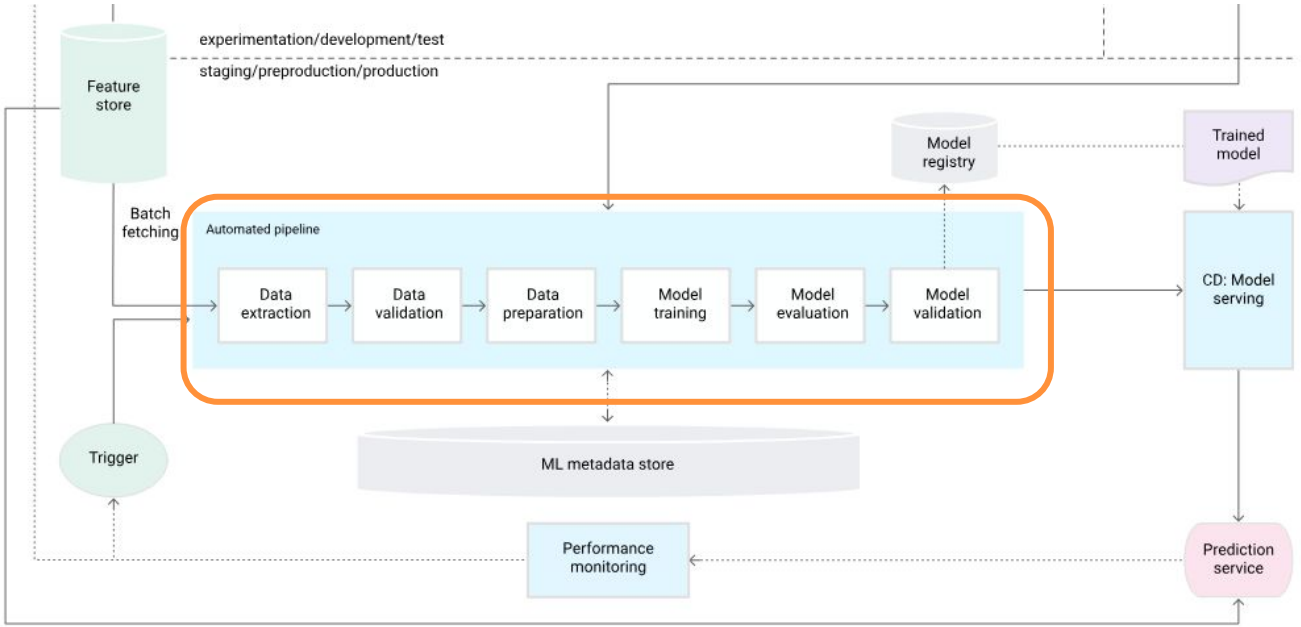
https://www.postman.com/what-is-an-api/

Key concept: Cloud infrastructure

Cloud infrastructure allow for data storage, compute allocation, training and deploying model, monitoring, …

Key concept: ML Pipeline

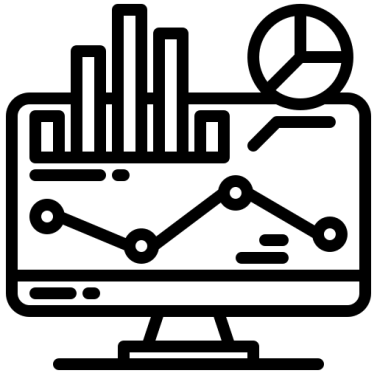
Orchestrates components to prepare data, train, evaluate and deploy ML models (among other things)



https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

Key concept: Monitoring

Ensuring that models in production are performing well.



Resource level (performance and usage of resources used by the model serving)

- How much is it being used by users?
- Are the CPU, RAM, network usage, and disk space as expected?
- What are the Cloud costs?
- Are requests being processed at the expected rate?
- What is the system uptime? Some maintenance contract depend on it.

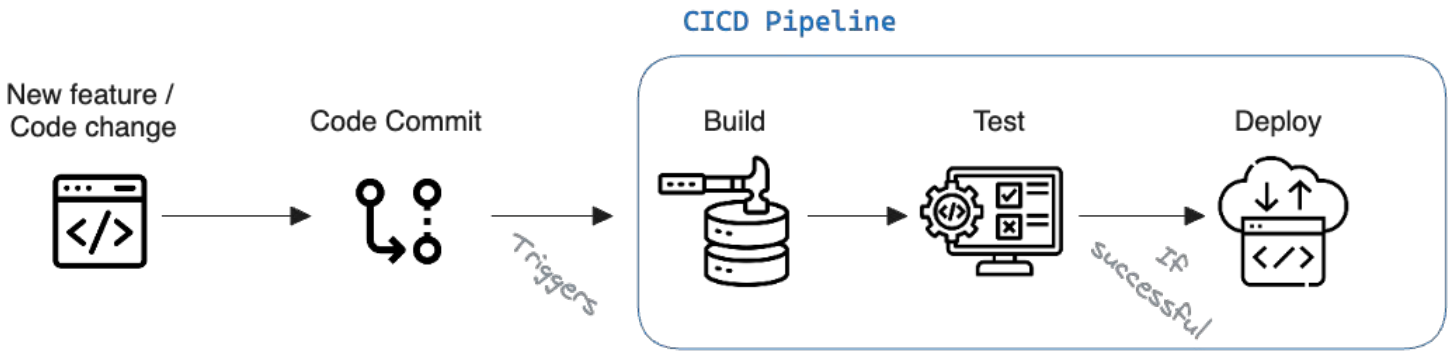
Performance level (performance/accuracy of the model over time)

- Is the model still doing accurate predictions with the new data coming in?
- Is the data distribution changing?
- Is the target variable changing?
- Are concepts around the model changing?

Key concept: CICD

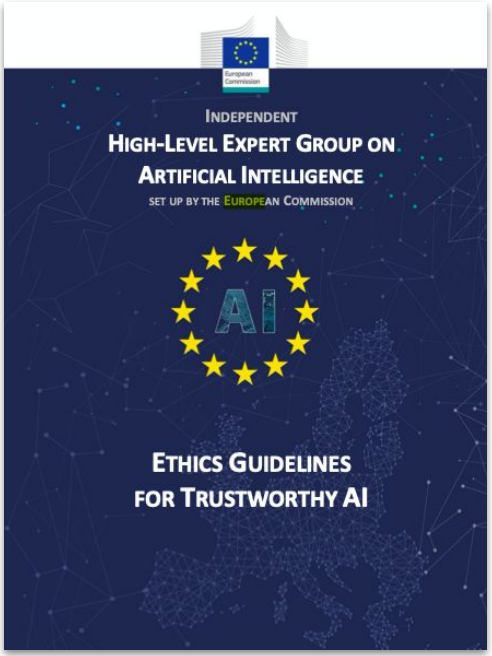
Allows you to continuously work on your application and efficiently deploy new changes to it.

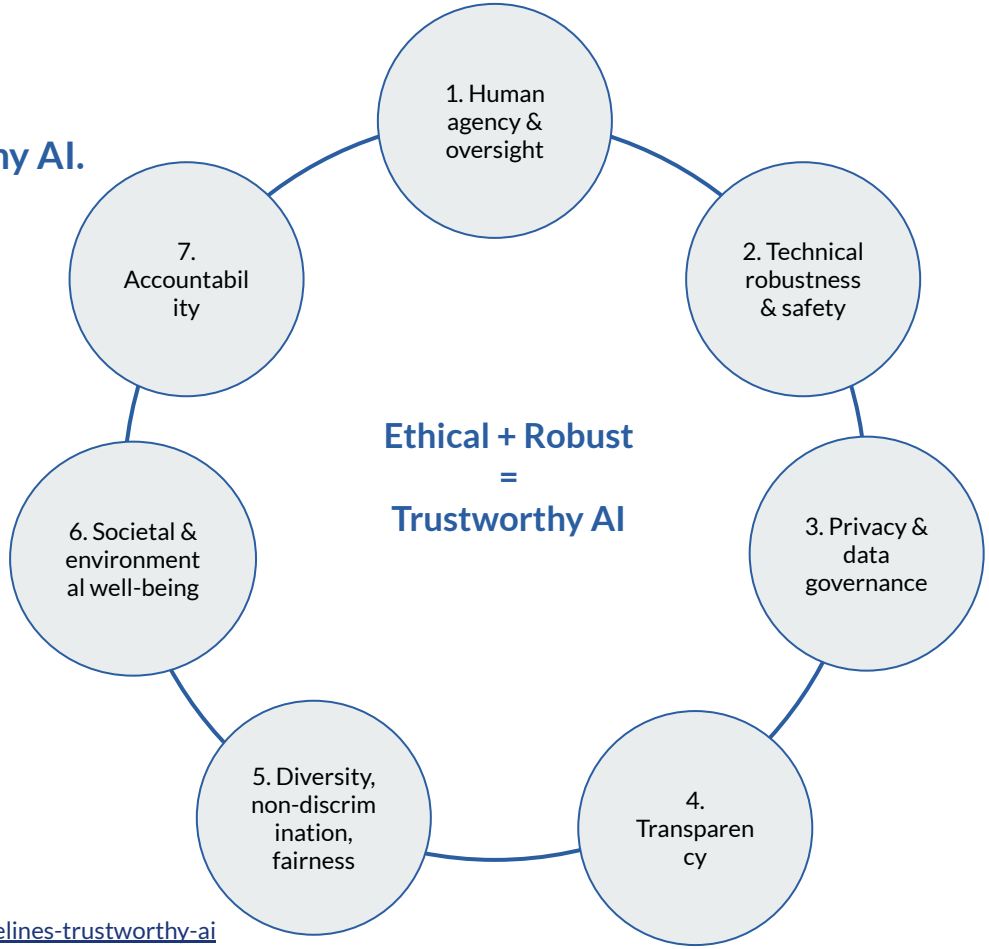
Continuous Integration and Continuous Delivery.



Key concept: Ethical AI

Guidelines & legislation on building trustworthy AI.

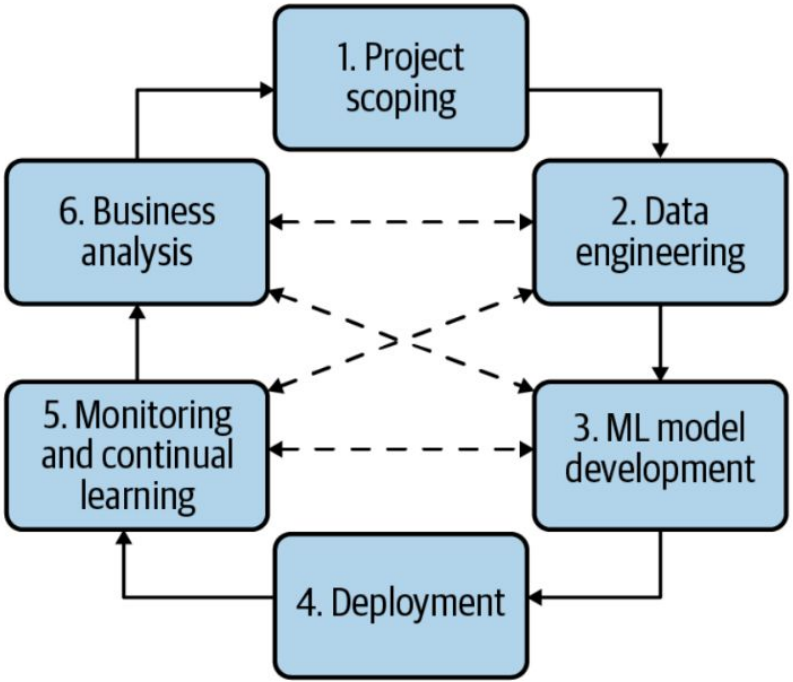




https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai

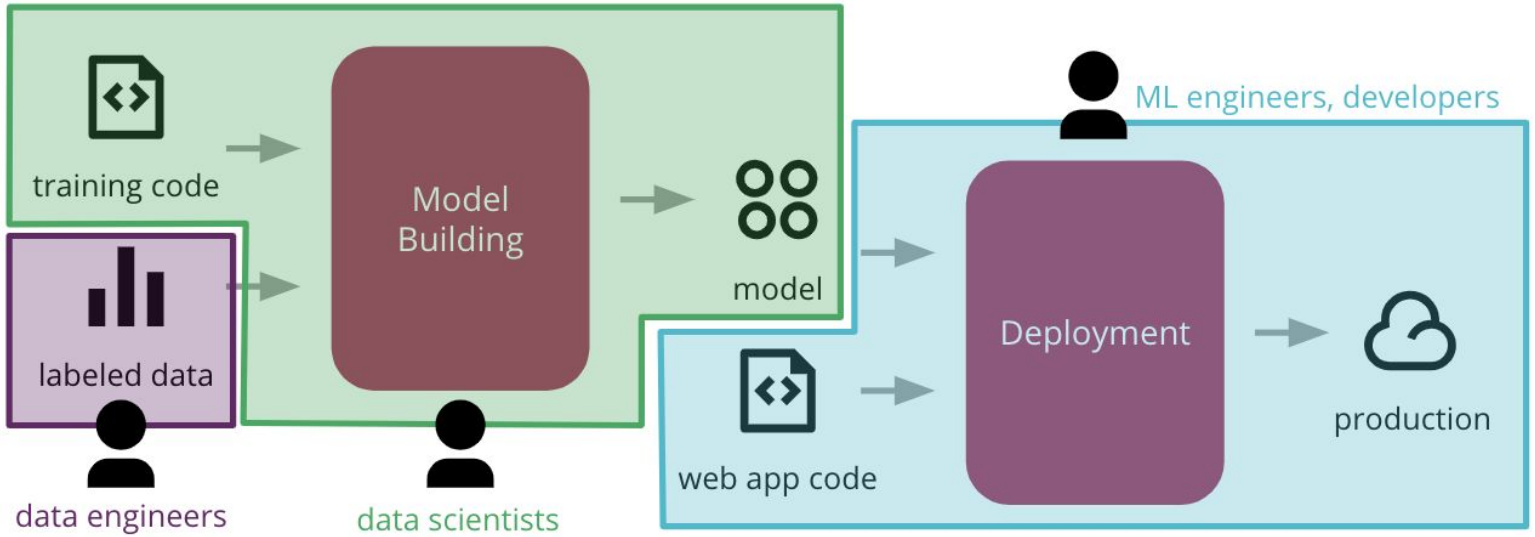
Roles & organisation of ML projects

Typical ML project lifecycle



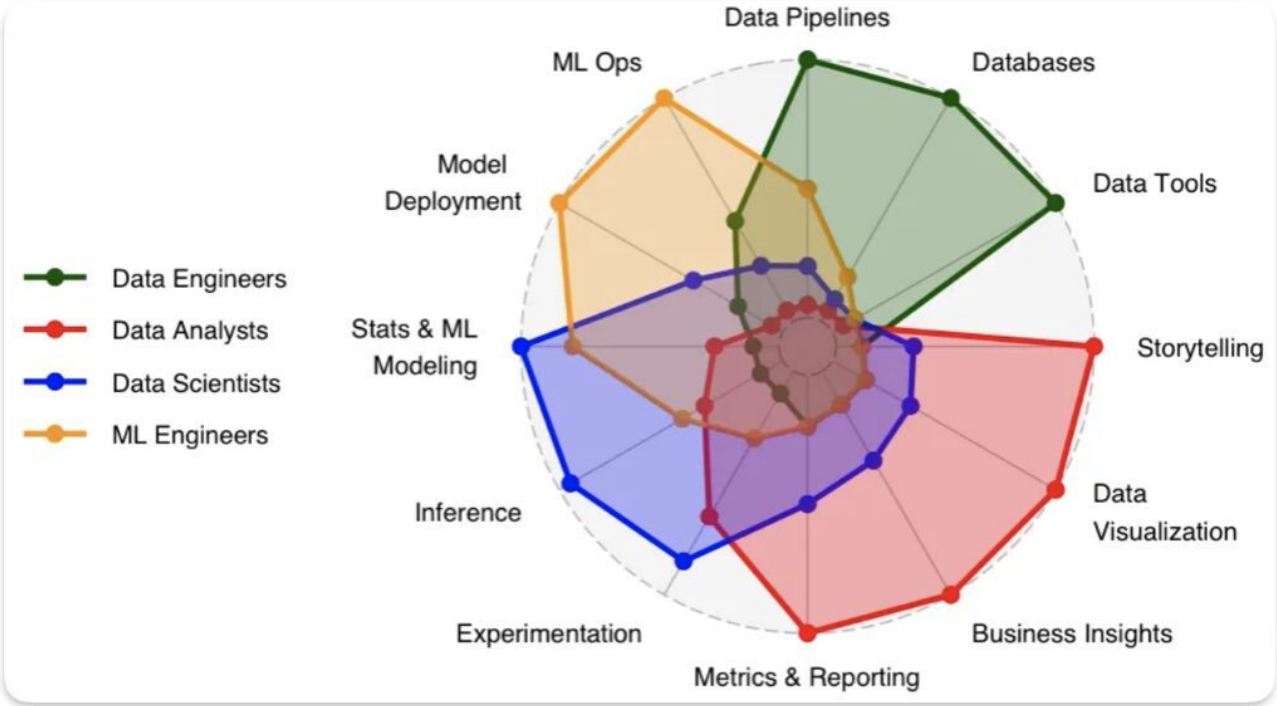
CS 329S: Machine Learning Systems Design - Stanford (Chip Huey)
https://stanford-cs329s.github.io/syllabus.html

Roles around a ML system implementation.



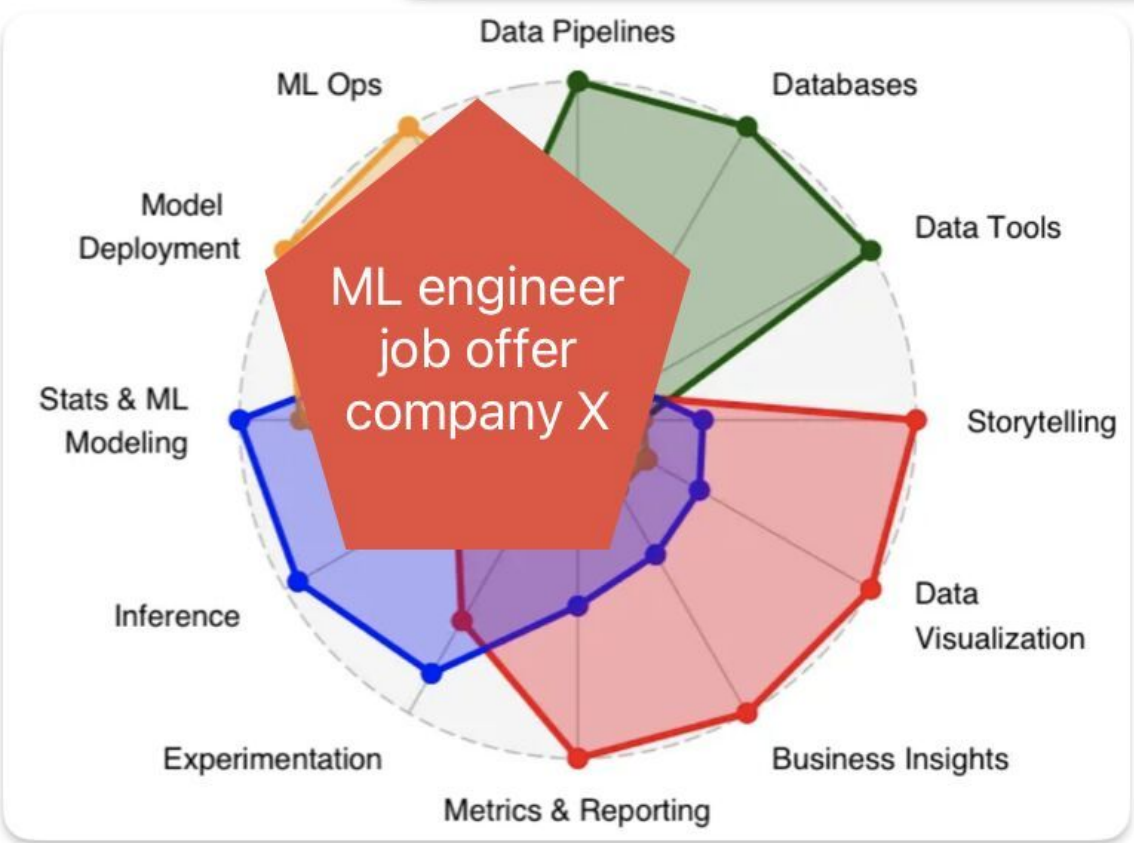
https://martinfowler.com/articles/cd4ml.html

Different set of skills per roles



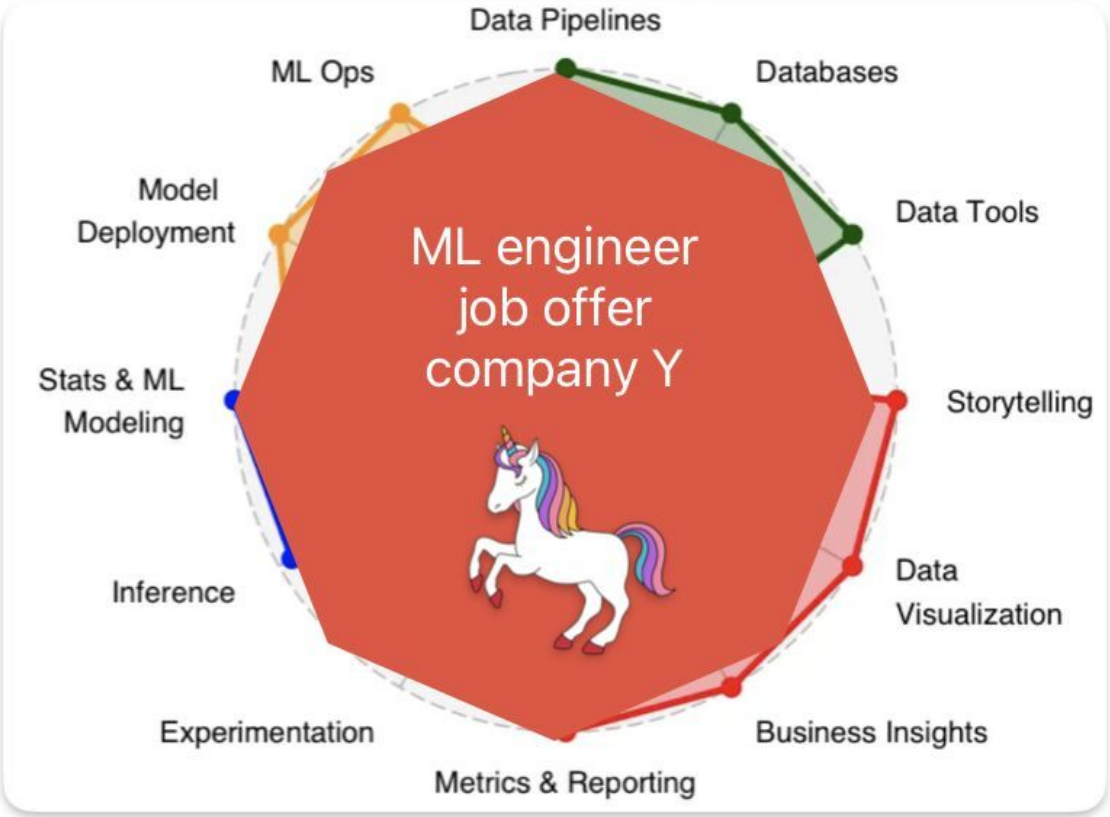
https://www.linkedin.com/posts/maria-vechtomova_data-ai-mlops-activity-7125095266403143680-9X-U/

In reality it's a bit blurry



https://www.linkedin.com/posts/maria-vechtomova_data-ai-mlops-activity-7125095266403143680-9X-U/

In reality it's a bit blurry



https://www.linkedin.com/posts/maria-vechtomova_data-ai-mlops-activity-7125095266403143680-9X-U/

ML Engineering skills are in high demand

Teams can adopt different MLOps maturity levels

Level	Highlights	Technology
Level 0 **No MLOps**	• Difficult to manage full ML model lifecycle • Teams are disparate and releases are painful • "black boxes," little feedback during/post deployment	• Manual training, builds and deployments • Manual testing of model and application • No centralized tracking of model performance
Level 1 **DevOps but no MLOps**	• Releases are less painful than No MLOps • Limited feedback on how well a model performs in production • Difficult to trace/reproduce results	• Automated builds • Automated tests for application cod
Level 2 **Automated Training**	• Training environment is fully managed and traceable • Easy to reproduce model • Releases are manual, but low friction	• Automated model training • Centralized tracking of model training performance • Model management
Level 3 **Automated Deployment**	• Releases are low friction and automatic • Full traceability from deployment back to original data • Entire environment managed: dev > test > production	• Integrated A/B testing of model performance • Automated tests for all code • Centralized tracking of model training performance
Level 4 **Full MLOps**	• Full system automated and easily monitored • Automated feedback collection and retraining • Close to zero-downtime	• Automated model training and testing • Verbose, centralized metrics from deployed model

https://learn.microsoft.com/en-us/azure/architecture/ai-ml/guide/mlops-maturity-model#level-0-no-mlops

Study on demanded skills for MLOps engineers.

Looking at 310 job offers on MLOps in Q4 2023.

Top 3 highest demanded skills:

https://marvelousmlops.substack.com/p/what-do-teams-really-want-from-an?utm_source=profile&utm_medium=reader2

Going from standard ML Engineer to MLOps master…

Real-life example of a MLOps system

Linkedin case study

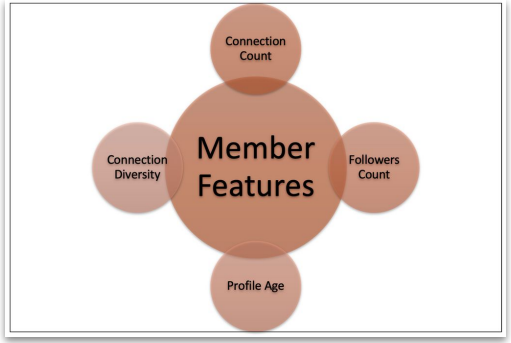
Linkedin integrates many ML applications

Viral spam content detection

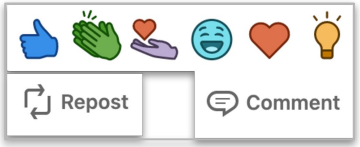
Detecting spam content…

… Using boosted tree algorithm on the following features:

Post features



Member features



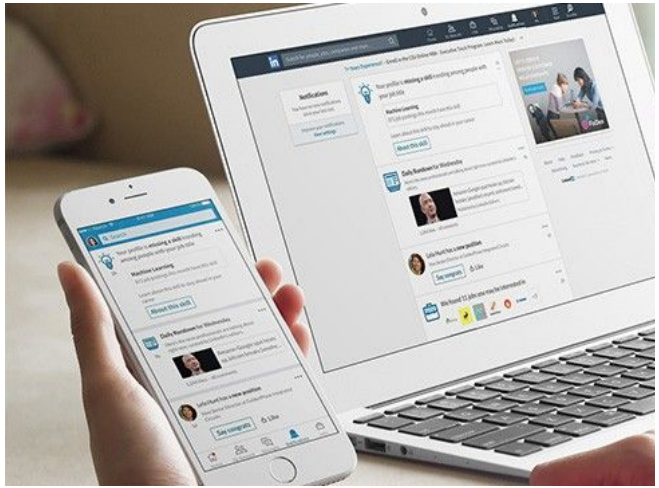
Engagement features

https://www.linkedin.com/blog/engineering/trust-and-safety/viral-spam-content-detection-at-linkedin

Linkedin integrates many ML applications

Personalised LinkedIn News Feed

Select personalised content for users…



… Using boosted tree algorithm on the following features:

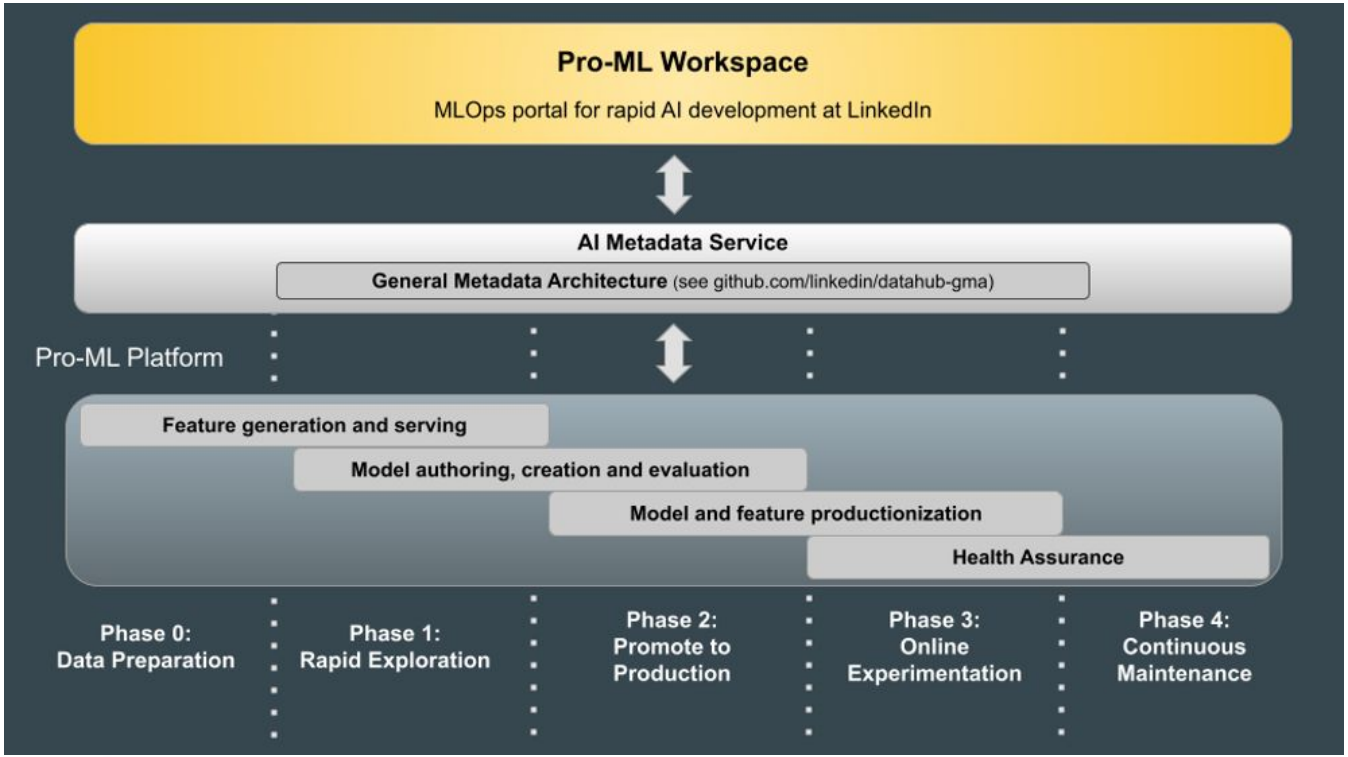
Identity: Who are you? Where do you work? What are your skills? Who are you connected with?

Content: How many times was the update viewed? How many times was it “liked”? What is the update about? How old is it? What language is it written in? What companies, people, or topics are mentioned in the update?

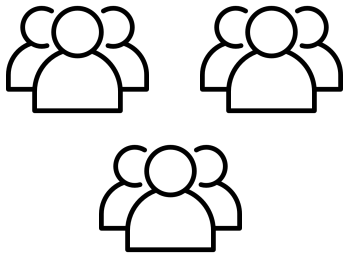
Behavior: What have you liked and shared in the past? Who do you interact with most frequently? Where do you spend the most time in your news feed?

https://www.linkedin.com/blog/engineering/feed/a-look-behind-the-ai-that-powers-linkedins-feed-sifting-through

Linkedin's Productivity Machine Learning (Pro-ML) platform.



Teams of data scientists

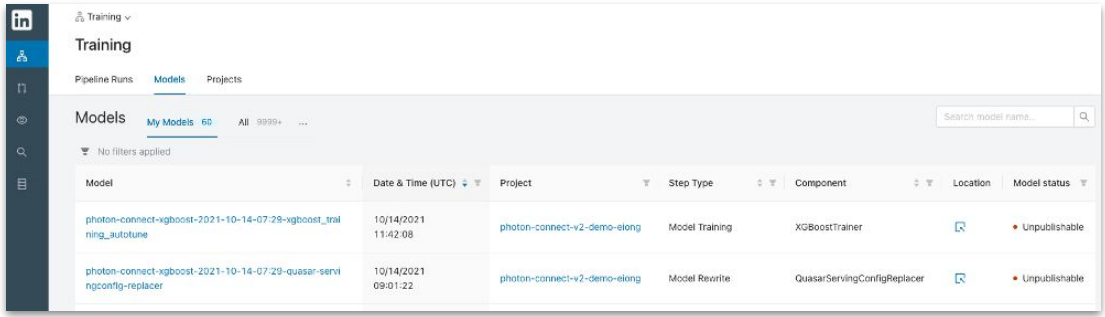


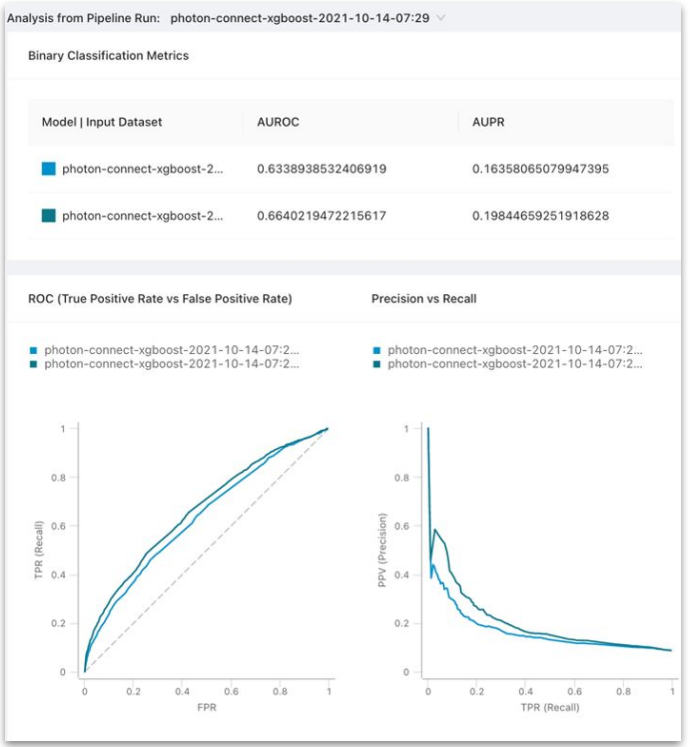
https://www.linkedin.com/blog/engineering/machine-learning/one-stop-mlops-portal-at-linkedin

Linkedin Pro-ML platform.

Step: Model authoring, creation, and evaluation

Model tracking and experimentation platform
Similar to **MLFlow** or **Weights & Biases** (which we will cover in this course).

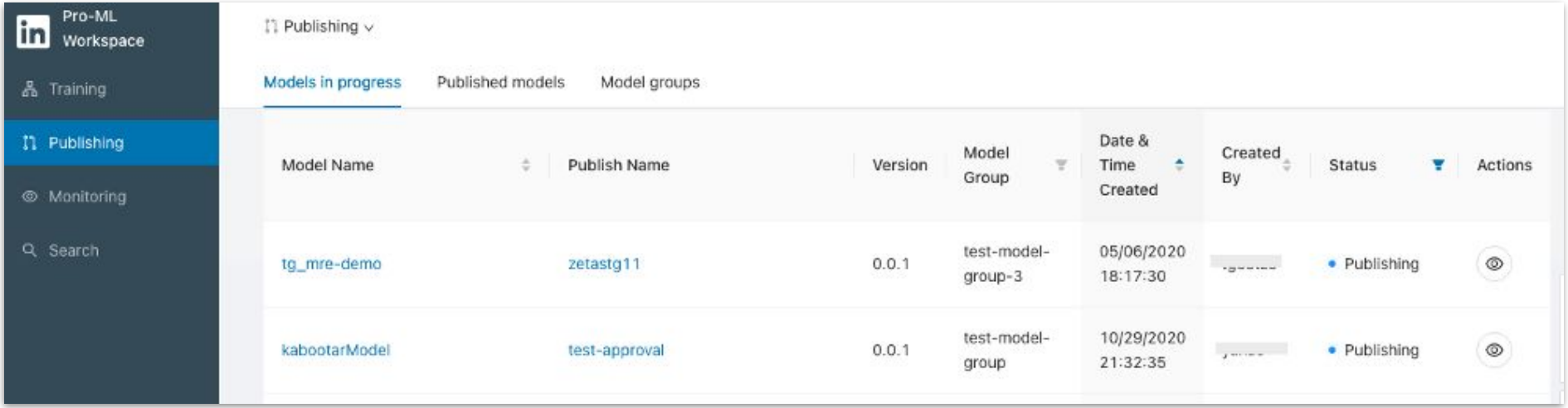




https://www.linkedin.com/blog/engineering/machine-learning/one-stop-mlops-portal-at-linkedin

Linkedin Pro-ML platform.

Step: Model productionisation



Workflows to **publish** or **deprecate** models.

https://www.linkedin.com/blog/engineering/machine-learning/one-stop-mlops-portal-at-linkedin

Linkedin Pro-ML platform.

Step: “Health insurance” (aka monitoring)

https://www.linkedin.com/blog/engineering/machine-learning/one-stop-mlops-portal-at-linkedin

Course organisation

Objective for this course.

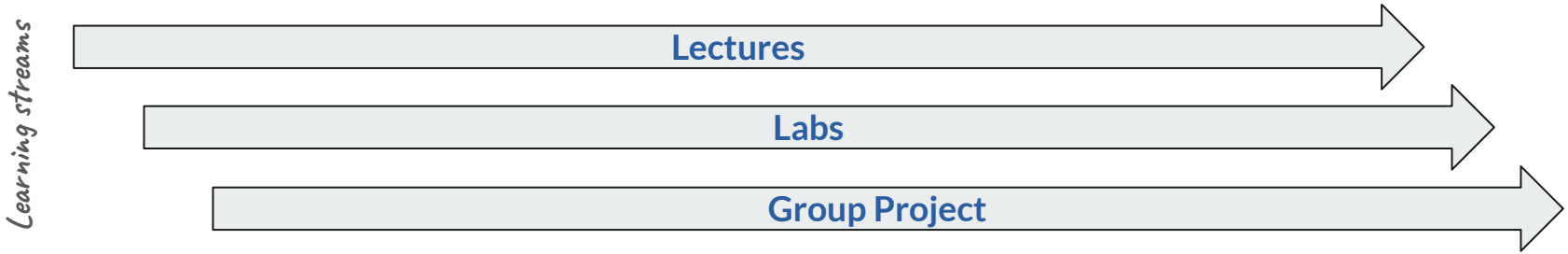
We want to enable **you** with practical skills to go make **positive impact with ML** 🚀

We'll cover key **concepts** of MLSD during our classes. We will also host Labs to show how to use key **tools** to develop ML applications.

We're happy if you learn **useful things** and can go **apply your own ideas.**

Structure of the course

Learning streams and pillars



Pillars

Relevant

Focused on core concepts of building ML applications. Tailored choice of current best practices.

Practical

Concrete Labs, resources, real life examples, time to experiment, support line, …

Engaging

Interactive class session. Healthy tempo (break out exercises, QA, …).

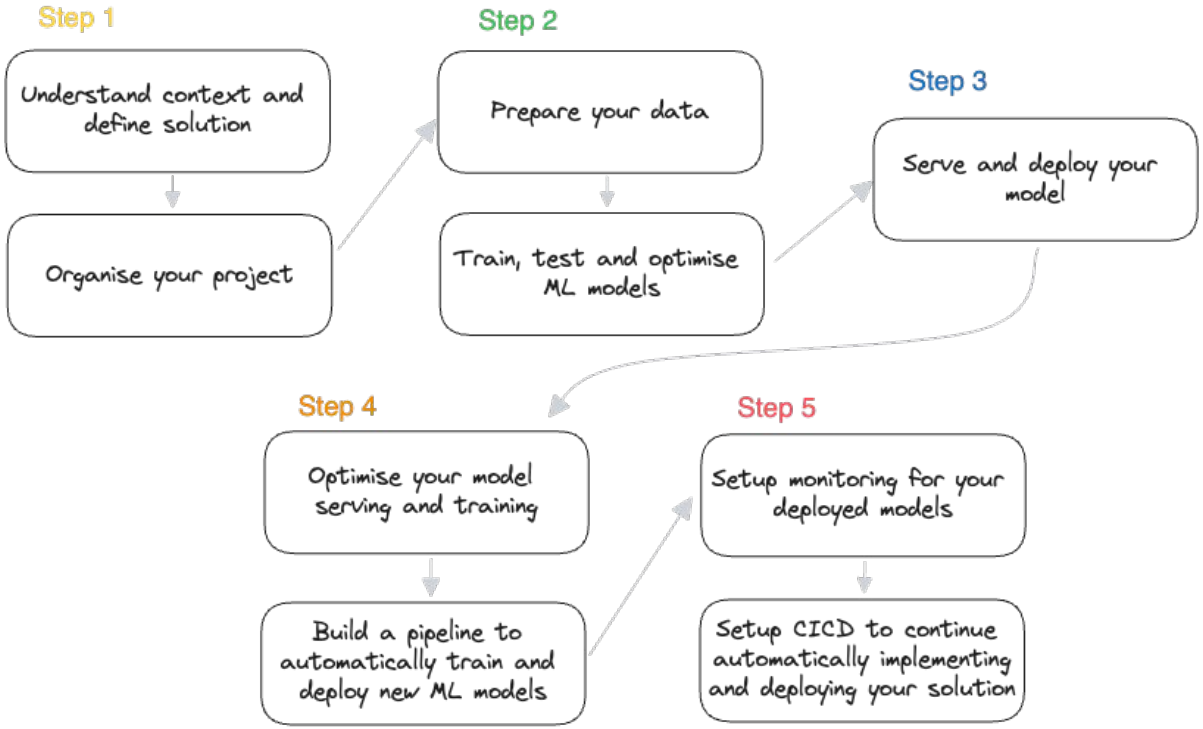
… lots of memes.

We’re making history

This is the first version of this class

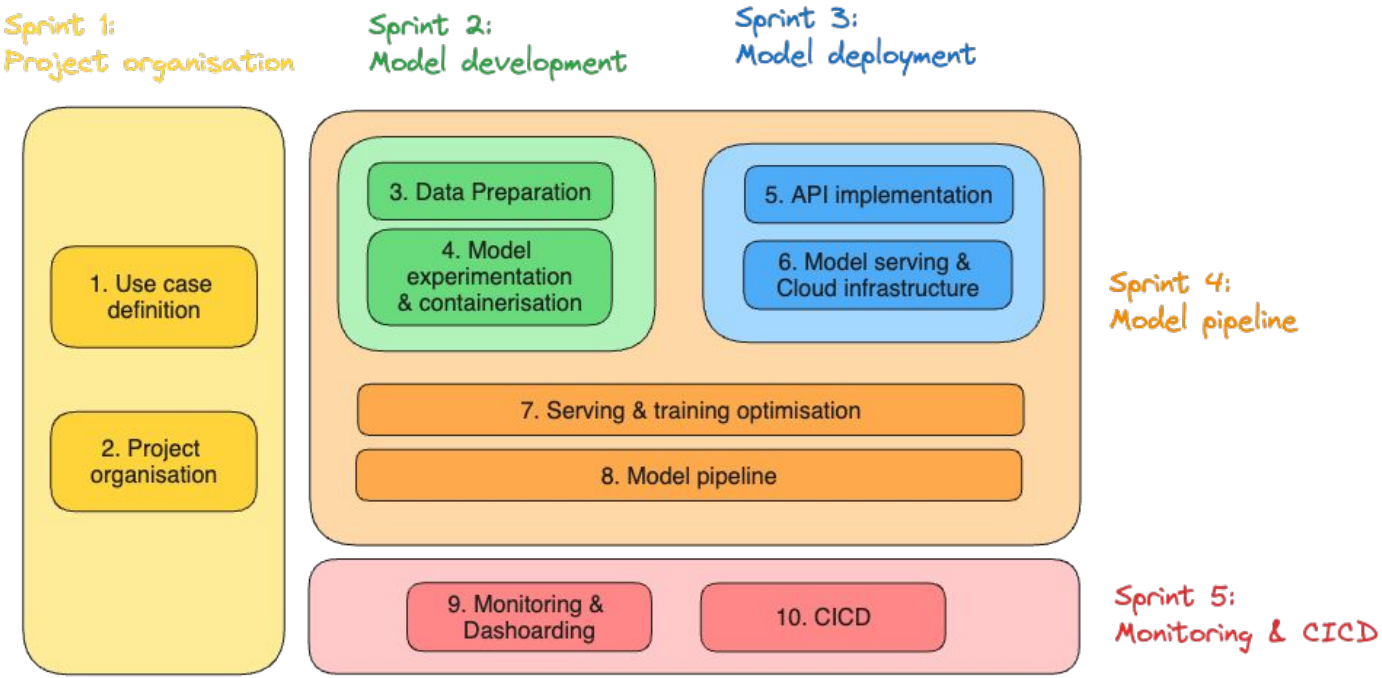
- Quick feedback cycles
- Open communication
- Enthusiasm for trying new things 🧪
- Active support from teaching staff

Another view on typical steps in an ML project



Course outline

Overview of sprints & classes



Overall organisation & communication

Class organisation

- We meet every Monday from **9:00** to **12:30** in B28 R.75 (0/75) [Liège Sart-Tilman - Polytech].
- Typically you'll have about 2h of lecture + labs. Remaining of the time can be spent working on your project.

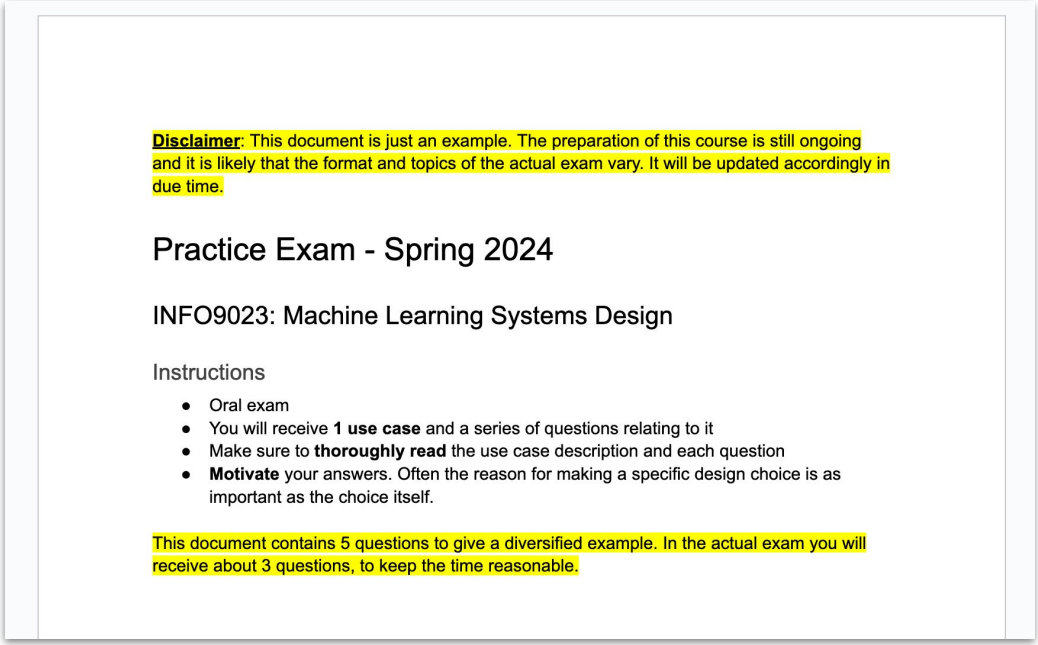
Useful links:

- All info on the Github page: https://github.com/ThomasVrancken/info9023-mlops
  - Project info
  - Sample exam
  - Lecture & labs (before the class)
- Discord: https://discord.gg/AVbAdNGR
- Open office hours on **Monday afternoons** (office Number I 77 B in Montefiore)

Grading

Exam & Project

1. Oral exam (30% of the final grade)
    - Find practice exam on Github
2. Project (70% of the final grade)



Disclaimer: This document is just an example. The preparation of this course is still ongoing and it is likely that the format and topics of the actual exam vary. It will be updated accordingly in due time.

Practice Exam - Spring 2024

INFO9023: Machine Learning Systems Design

Instructions

- Oral exam
- You will receive **1 use case** and a series of questions relating to it
- Make sure to **thoroughly read** the use case description and each question
- **Motivate** your answers. Often the reason for making a specific design choice is as important as the choice itself.

This document contains 5 questions to give a diversified example. In the actual exam you will receive about 3 questions, to keep the time reasonable.

Project

All information is on Github.

Organisation

Build one ML system throughout the course. The application is picked by yourself.

- **Teams:** 3 - 5 students
  - Try to form group by next week!
  - Let the teaching staff know if you don't have a group and you'll be assigned one
- **Structure**
  - The building blocks to be implemented in the project follow the course's **5 sprints**.
- **Handovers**
  - There will be **3 milestone meetings** where you can present your results
  - **Code submission** - make sure to document anything you want the teaching staff to read
- **Support**
  - Often lectures/labs will be shorter than the time slot for this course. You can spend the extra time working with your team. Teaching staff will be in the room to provide support.
  - Open office hours on Monday afternoon in office Number I 77 B in Montefiore
  - Feel free to reach out by email if you have any question/struggle
- **You're in the driving seat!**
  - Many building blocks are optional. You are free to choose the overall design and tools used for your project. Experiment and ask questions if you have any.

Project

Guiding principles

- Learn, learn and learn!
- Find a fun project to work on - ideally with a real world usage
- Come up with your own design and toolstack
- Motivate your design choices

- … And pick a cool name for it

Resources

Similar courses

- University of Bari
  - Paper: "Teaching MLOps in Higher Education through Project-Based Learning." arXiv preprint arXiv:2302.01048 (2023)
  - Lanubile, Filippo, Silverio Martínez-Fernández, and Luigi Quaranta
- Stanford University
  - CS 329S: Machine Learning Systems Design (link)
  - Chip Huyen
- Carnegie-Mellon University
  - Machine Learning in Production / AI Engineering (link)
  - Christian Kästner

Interesting resources

- Machine Learning Engineering for Production (MLOps) Specialization (Coursera, Andrew Ng)
  - GitHub, Youtube
- Made with ML (link)
- Marvelous MLOps (link)

Books

- Designing Machine Learning Systems: An Iterative Process for Production-Ready Applications (Chip Huyen)
- Building Machine Learning Powered Applications: Going from Idea to Product (Emmanuel Ameisen)
- Introducing MLOps (Mark Treveil, Nicolas Omont, Clément Stenac et al.)
- Machine Learning Design Patterns (Valliappa Lakshmanan, Sara Robinson, Michael Munn)