CHALMERS UNIVERSITY OF TECHNOLOGY | UNIVERSITY OF GOTHENBURG

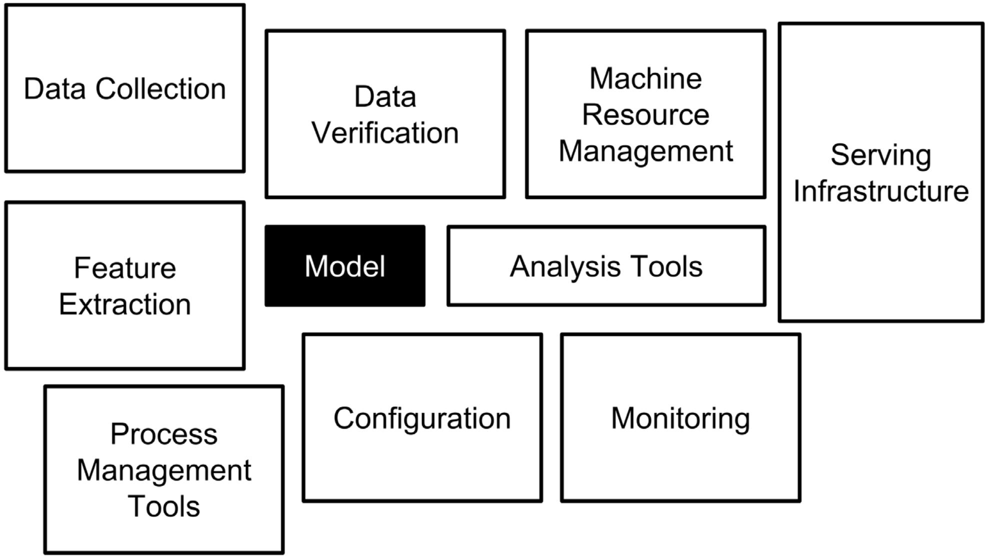
Engineering of ML Systems
Part 1

DIT826

Daniel Strüber

Learning objectives

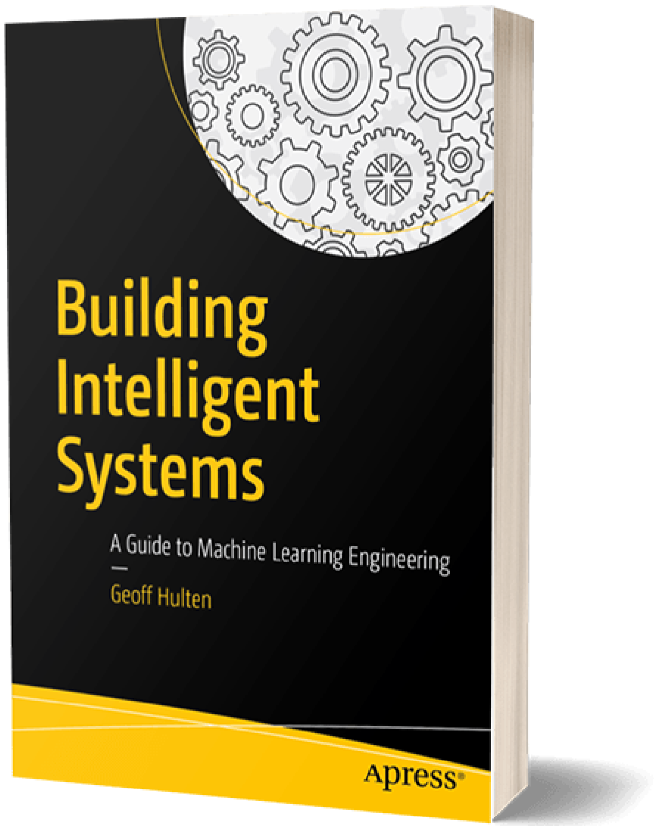
- **Understand** that building ML systems is more than training a model
- **Understand** practices and challenges of ML systems engineering



Book

Building Intelligent Systems: A guide to Machine Learning Engineering, Geoff Hulten

Available as an **e-Book** at GU and Chalmers libraries



Mandatory reading

- D. Sculley et al., Hidden Technical Debt in Machine Learning Systems, https://papers.nips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems.pdf

- S. Amershi et al., Software Engineering for Machine Learning: A Case Study, https://www.microsoft.com/en-us/research/uploads/prod/2019/03/amershi-icse-2019_Software_Engineering_for_Machine_Learning.pdf

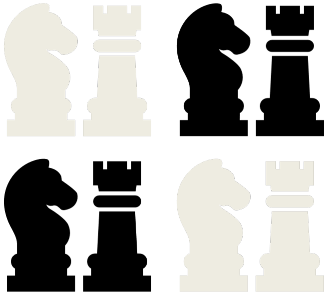
- M. Zinkevich, Rules of Machine Learning: Best Practices for ML Engineering, http://martin.zinkevich.org/rules_of_ml/rules_of_ml.pdf

Machine Learning

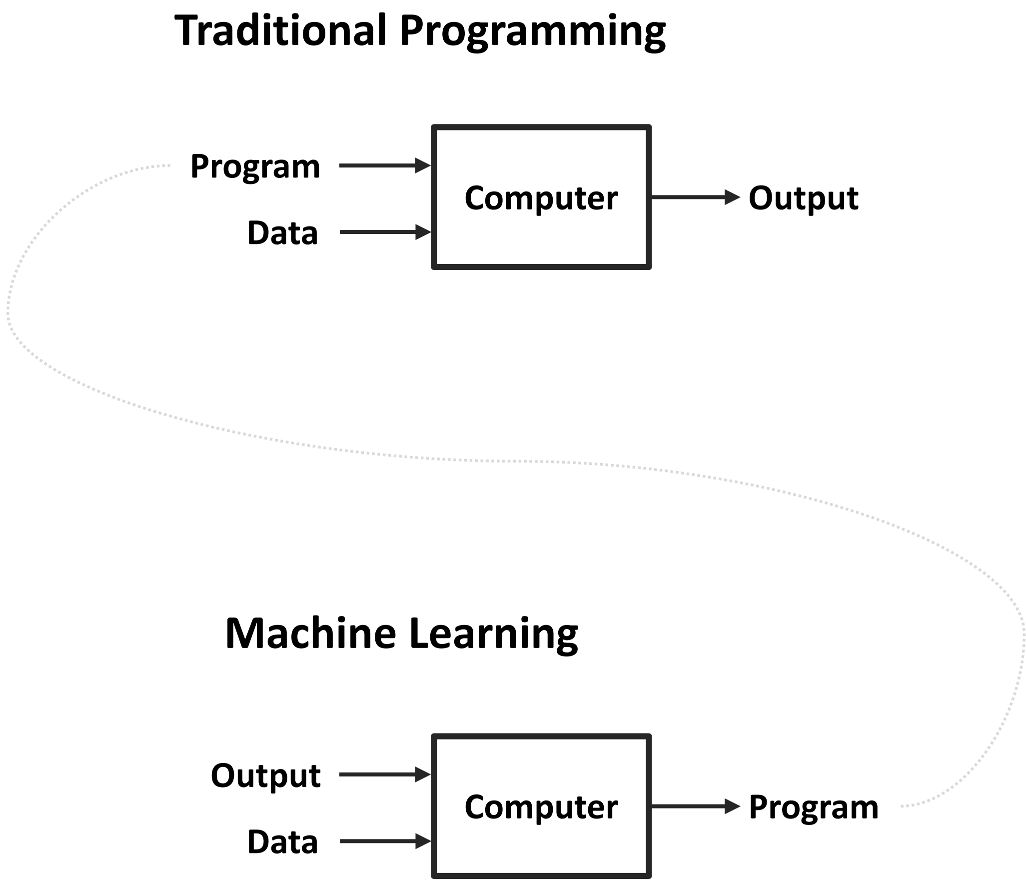
- Is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns, and make decisions with minimal human intervention.

- "A computer program is said to learn from **experience E** with respect to some class of **tasks T** and **performance** measure **P**, if its **performance** at **tasks** in **T**, as measured by **P**, improves with **experience E**" Tom Mitchell

- Example
  - Task T, What is the task?
  - Experience E, What is the Experience?
  - Performance P, What is the Performance?
    - *If P increases with E → the machine is learning!*



Programming vs. ML



Software Engineering

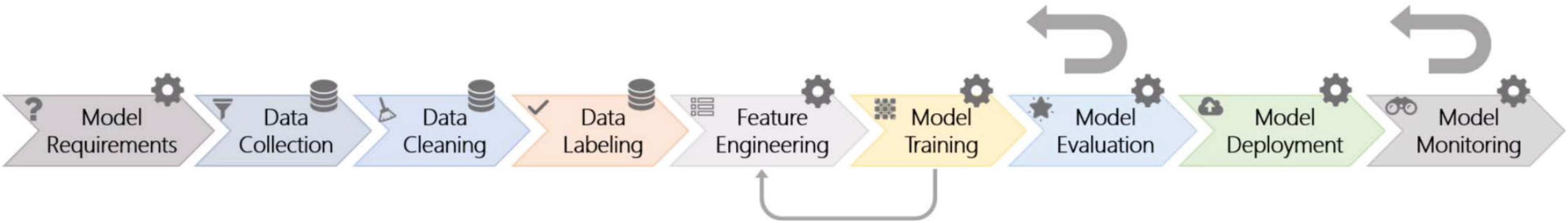
- A systematic approach for software development.

- An iterative process that includes different activities, such as:
  - planning, risk analysis, requirement engineering, software design, coding, versioning, testing, integration, deployment, maintenance, etc.

Software Engineering For ML

- A systematic way to create and integrate ML into software products and services.

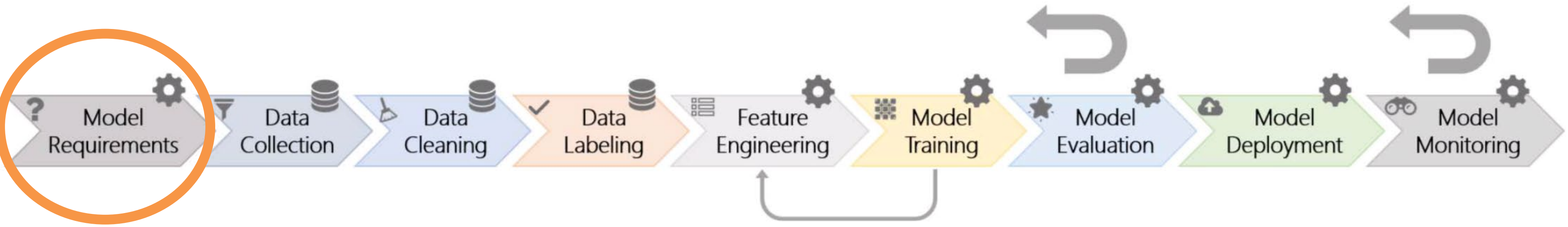
Workflow is needed



- Not a linear process

S. Amershi et al., Software Engineering for Machine Learning: A Case Study, ICSE-SEIP, 2019

The ML Workflow



- Model Requirements:
  - Problem and Goals?
  - Models?

Problems and when to use ML

- Syndrome: *"We need ML because it's trendy"**
- How often you think you need to update your system?
  - If *n* is small, then using ML is probably not right.

- Example:

  NewBalance = OldBalance – WithdrawalAmount

*Rule #1: Don't be afraid to launch a product without machine learning, in "M. Zinkevich, Rules of Machine Learning"

Requirements: Problem

- Big problems
  - ~100 million songs
  - ~130 million books
  - ~1.5 billion websites
- Time changing problems
  - New technologies
  - Human faces – masks, face tattoos?
  - UX
- Open ended problems
  - ~6k tweets per second
  - ~60k new web pages per day
  - ~3 billion active Facebook users
- Intrinsically hard problems
  - Weather prediction
  - Complex Open-ended games
    - First chess program 1957
    - Deep Blue 1997 beat Kasparov
    - AlphaGo

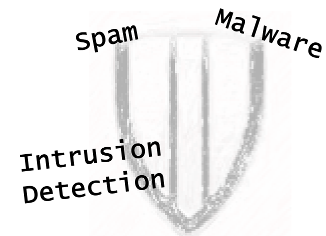
Success of ML and AI

Source: Geoff Hulten, *Building Intelligent Systems*

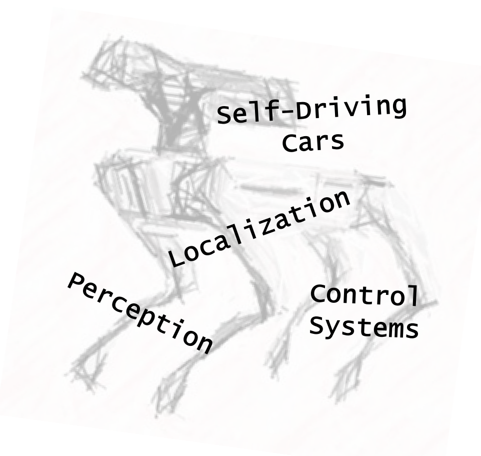
Web Search

Finance

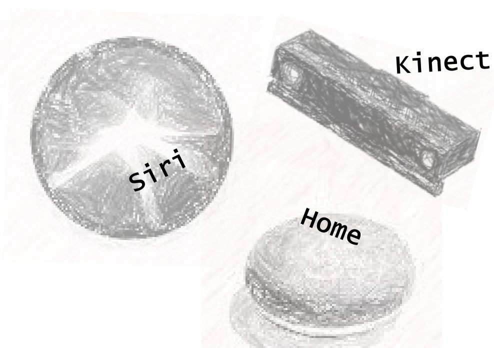
Marketing & E-commerce



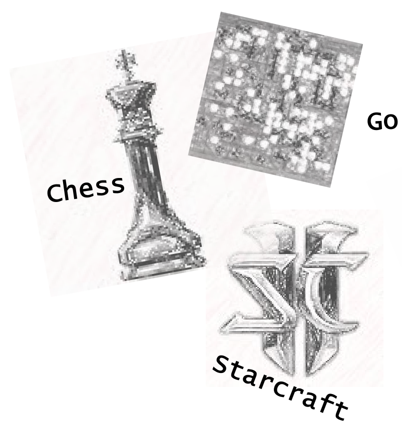
Abuse / Security



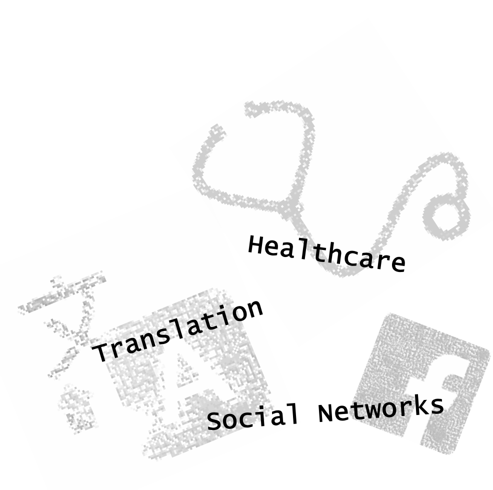
Robotics



Digital Assistants

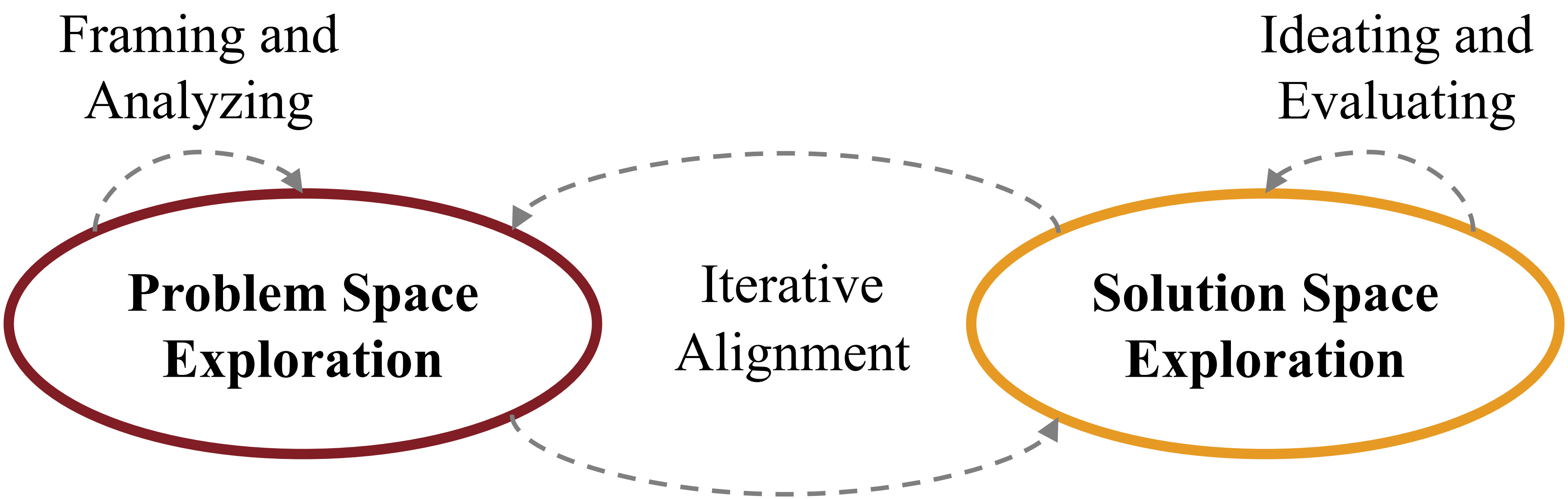


Games



Many Others

A Problem-Solving Technique: *Design Thinking*



- Lindberg et al. *Design thinking: A fruitful concept for IT development?* In *Design thinking*. Springer, 3–18, 2011
- Nigel Cross. *Design thinking: Understanding how designers think and work*. Berg, 2011

Defining the Goals

- A successful goal should:
  1. Communicate the desired outcome: *what?*
  2. Be achievable: *how?*
  3. Be measurable: *does it work?*

- Spam/Not Spam

Model Type

Source: Geoff Hulten, *Building Intelligent Systems*

- What models are most appropriate for the given problem? (classification, clustering, etc.)

Book Title	Number of Pages	Year Published	Genre	HasWord(Robot)	Author ID	Best Seller
Gone With The Wind	1037	1936	Historical Romance	0	1001	1
For Whom the Bell Tolls	480	1940	War Drama	0	1010	1
I, Robot	253	1980	Science Fiction	1	1020	1
One Hundred Goodbyes	100	2018	Science Fiction	0	1030	0

Linear Models
Decision trees
Ensembles of models
Neural networks
Support vector machines
Etc.

Decision Tree

F(X)
0
1
0
1

```
def F(BookTitle, NumberOfPages, YearPublished, Genre, HasWord(Robot), AuthorID):

    if YearPublished > 1990:
        if Genre == "Science Fiction":
            return 1
        else:
            return 0
    elif AuthorID == 1010:
        return 1
    else:
        return 0
```

Linear Model

F(X)
1
0
0
0

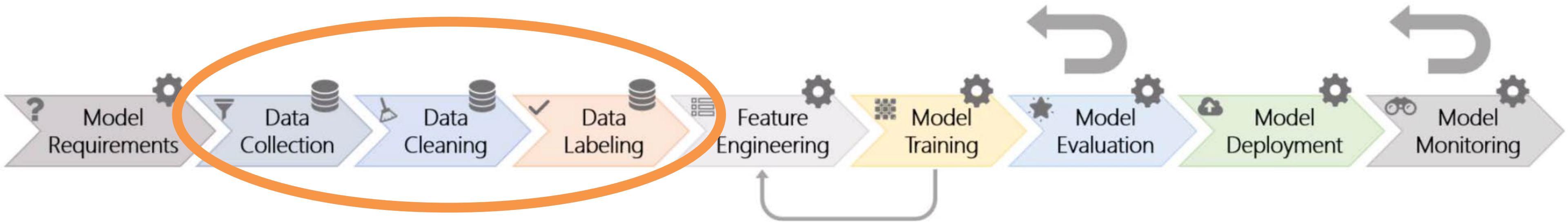
```
def F(BookTitle, NumberOfPages, YearPublished, Genre, HasWord(Robot), AuthorID):

    sum = 0.5 * NumberOfPages + 0.75 * YearPublished + 0.1 * AuthorID

    return 1 if sum > 2000 else 0
```

Questions?

The ML Workflow



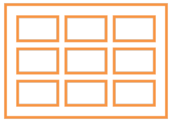
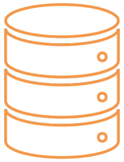
- Working with data:
  - Collection
  - Cleaning
  - Labeling

Data pipeline
Definition

- Process that takes input data through a series of transformation stages, producing data as output.
- Both the input and output data can be fetched and stored in different locations, such as a database, a stream, a file, etc.

Data collection

- Prepare the scripts to fetch the raw data

- Store the data
  - CSV
  
  - DB
  
  - Cloud
  

Data collection

- Sources
  - Existing data sources in the company
  - Existing open source data
  - Collect new data

Data cleaning

- The data are often unstructured and can be quiet difficult to work with.

- Dealing with noise e.g., inaccurate and incomplete data.

Data labeling

- Ground truth

- Sometimes readily available
  (sale price of house collected from a website)

Pitfalls of working with data

Broken confidence intervals

- 95% chance of being within an interval means that there is a 5% chance of being outside the interval.

- *Example*: Self-driving cars
  - 100 times → 5 mistakes

- TIP: Ask yourself few questions:
  - Is this right?
  - How sure are we?
  - Is there another interpretation?
  - How can we know which is correct?

Pitfalls of working with data

Noisy data

- Every large data set will have noise.
- Noisy data will inject errors into things created from these data.

- *Mitigation*:
  - Need for validation (There is lecture on this soon!)

Pitfalls of working with data

Biased data

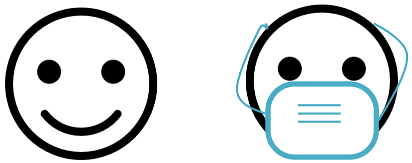
- Bias happens when data is collected in ways that are systematically different from the way the data is used.
- Bias can make data less useful.

- *Mitigation*:
  – Get more relevant telemetry or training data that contain context.

Pitfalls of working with data

Out-of-date data

- Things do change and collected data might not be representative anymore.
- *Example*: face recognition



- *Mitigation*
  - Train and deploy new models.

Questions?