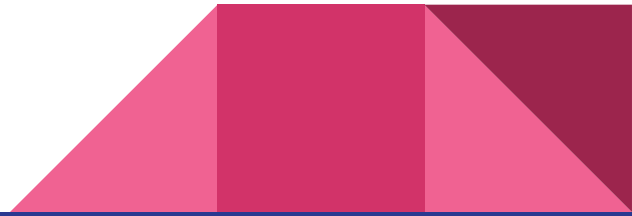
CIS*3750 - System Analysis and Design in Applications

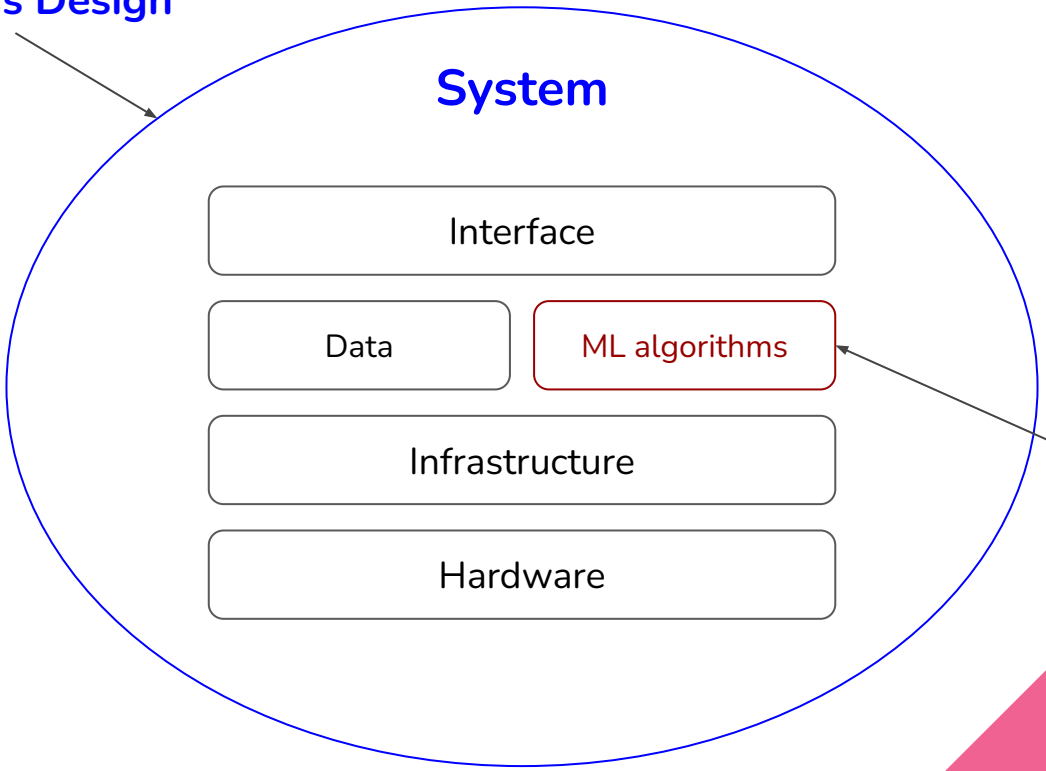
Luiza Antonie, Fall 2025, University of Guelph

Labs - Reminder

- Start next week
- You must attend your assigned lab section
- You will be given your group in the lab
- Lab 1
  - Get to know your team (skills, strengths, weaknesses)
  - Group contract
    - Evaluation rubric



ML Systems Design



Most ML courses/books

What’s machine learning systems design?

The process of defining the **interface, algorithms, data**, **infrastructure**, and **hardware** for a machine learning system to satisfy **specified requirements**.

What's machine learning systems design?

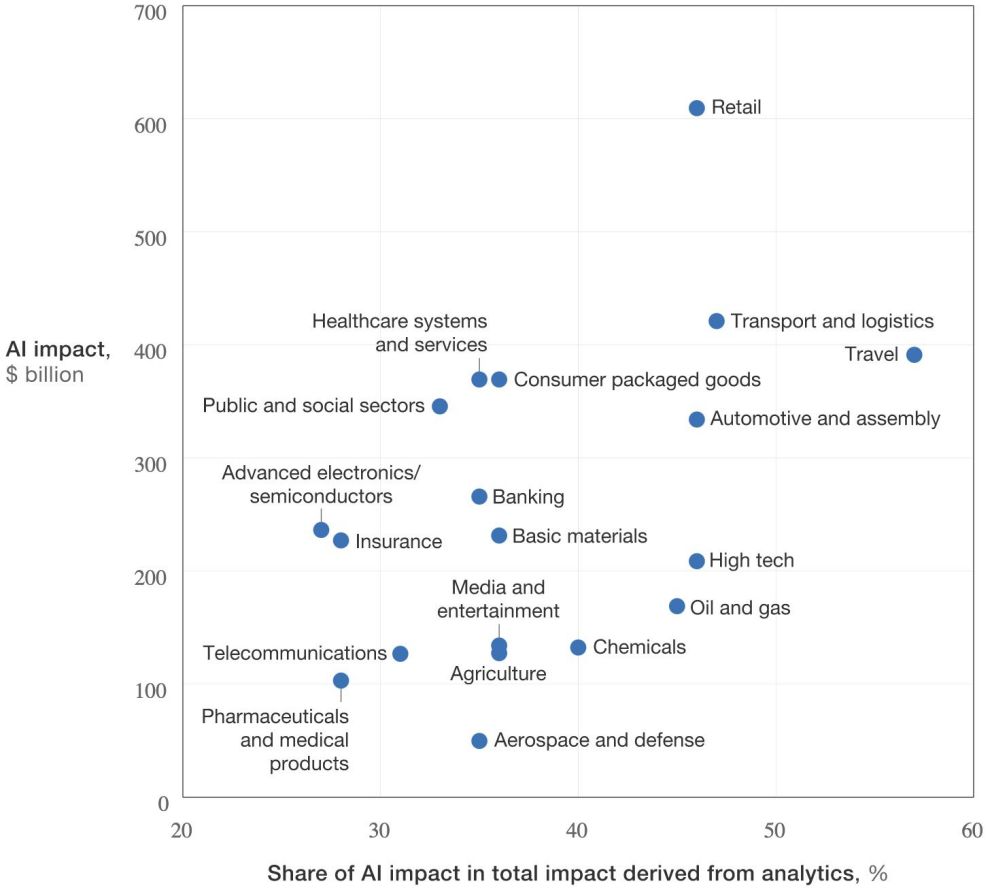
The process of defining the **interface, algorithms, data**, **infrastructure**, and **hardware** for a machine learning system to satisfy **specified requirements**.

reliable, scalable, maintainable, adaptable

The questions this class will help answer …

- You've trained a model, now what?
- What are different components of an ML system?
- How to do data engineering?
- How to engineer features?
- How to evaluate your models, both offline and online?
- What's the difference between online prediction and batch prediction?
- How to continually monitor and deploy changes to ML systems?
- …

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



AI value creation by 2030

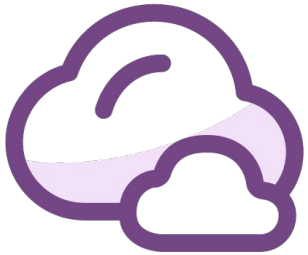
13 trillion USD

Most of it will be outside the consumer internet industry

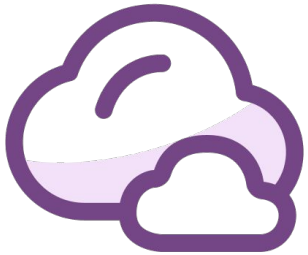
We need more people from non-CS background in AI!

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

ML research vs. ML production



When you think of ML in research, what words come to mind?



When you think of ML in production, what words come to mind?

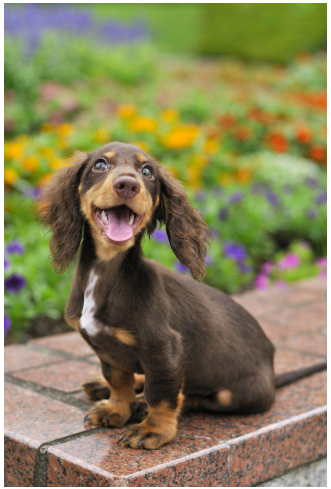
ML research vs. ML production

	Research	**Production**
Objectives	Model performance*	Different stakeholders have different objectives

"*" It's actively being worked. See Utility is in the Eye of the User: A Critique of NLP Leaderboards (Ethayarajh and Jurafsky, EMNLP 2020)

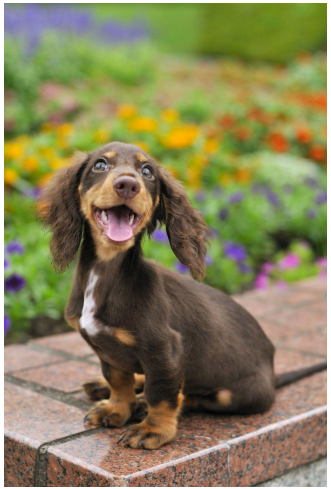
Stakeholder objectives

ML team
highest accuracy

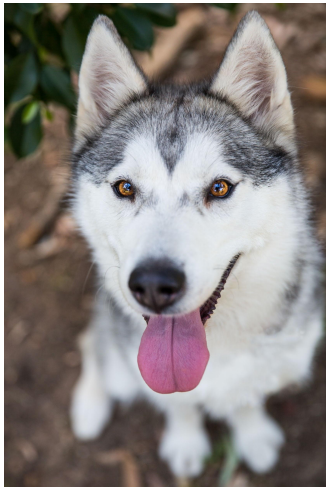


Stakeholder objectives

ML team
highest accuracy

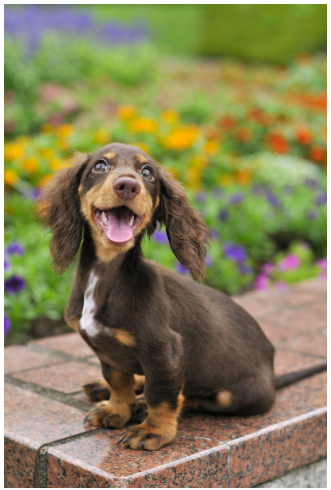


Sales
sells more ads

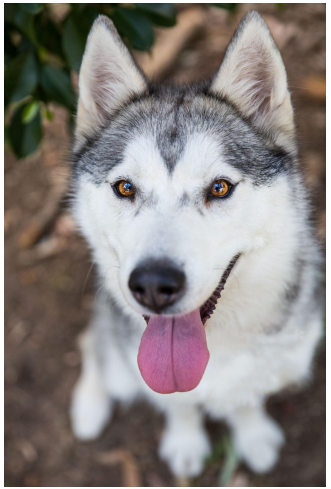


Stakeholder objectives

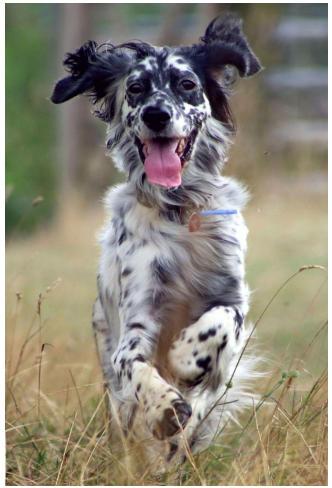
ML team
highest accuracy



Sales
sells more ads

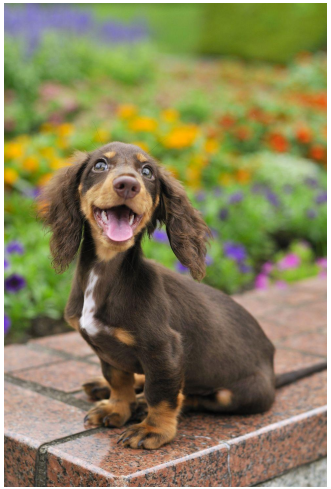


Product
fastest inference

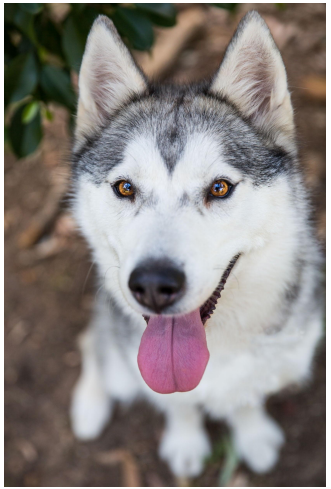


Stakeholder objectives

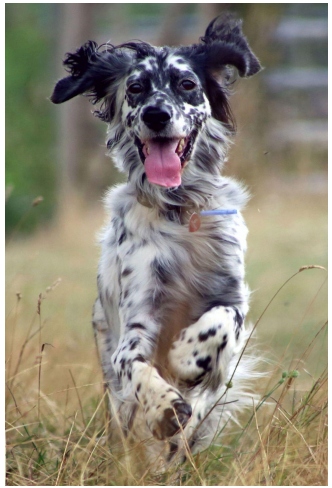
ML team
highest accuracy



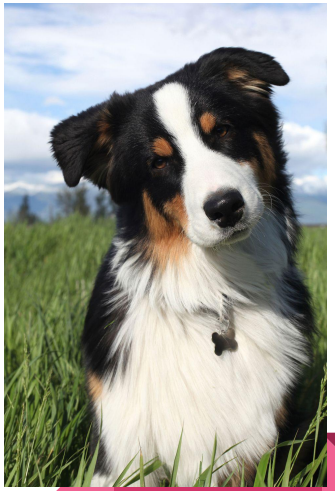
Sales
sells more ads



Product
fastest inference



Manager
maximizes profit
= laying off ML teams



Leaderboard-style ML

- More comprehensive utility function
  - Model performance (e.g. accuracy)
  - Latency
  - Prediction cost
  - Interpretability
  - Robustness
  - Ease of use (e.g. OSS tools, community support)
  - Hardware requirements
- Adaptive to different use cases
  - Instead of a leaderboard for each dataset/task, the leaderboard adapts to each company's needs
- Dynamic datasets
  - Realistic distribution shifts with different types of shifts

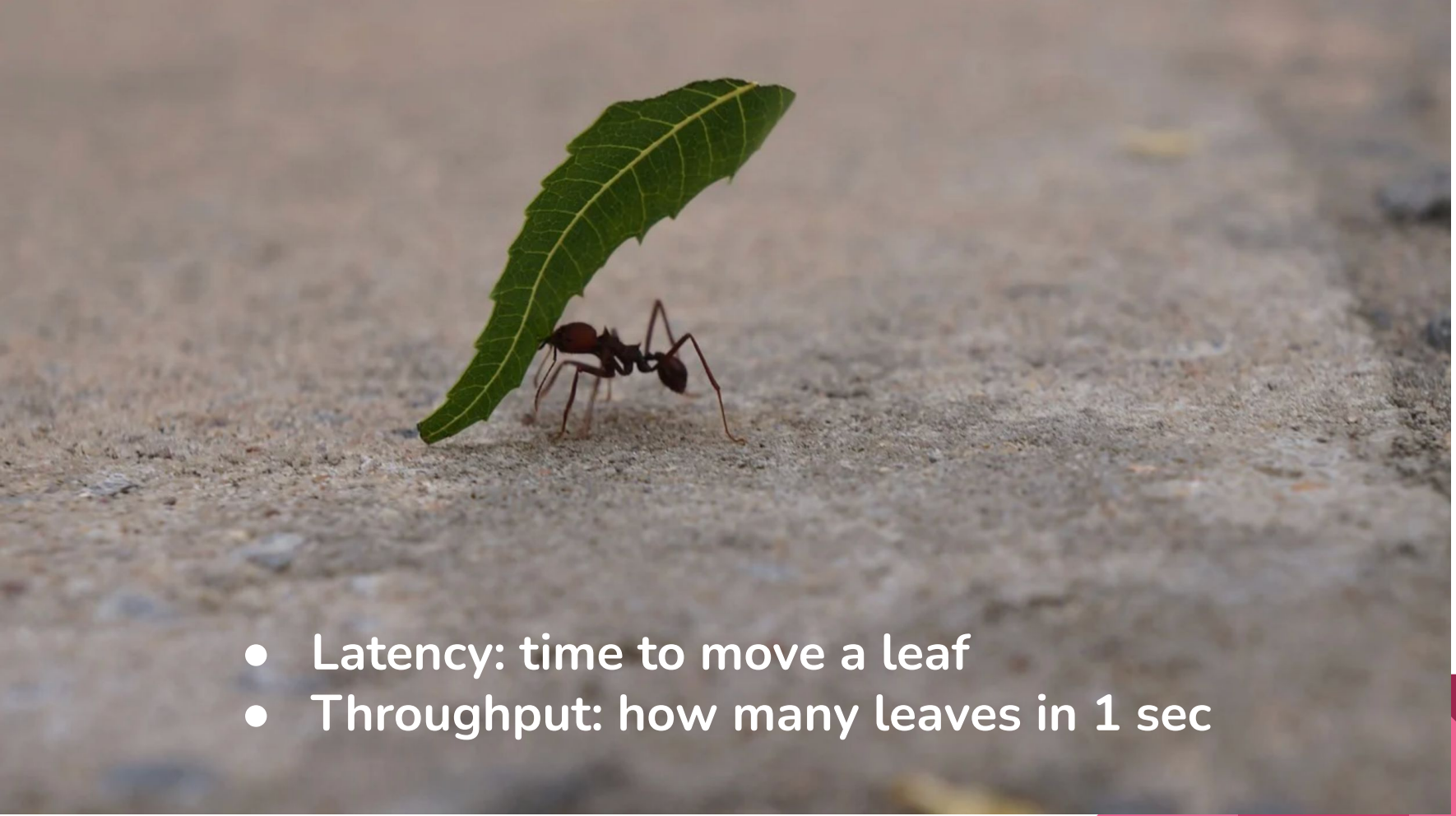
Computational priority

	Research	**Production**
Objectives	Model performance	Different stakeholders have different objectives
Computational priority	Fast training, high throughput	Fast inference, low latency

generating predictions

Latency matters

- 100ms delay can hurt conversion rates by 7% (Akamai study ‘17)
- 30% increase in latency costs 0.5% conversion rate (Booking.com ‘19)
- 53% phone users will leave a page that takes >3s to load (Google ‘16)



- Latency: time to move a leaf
- Throughput: how many leaves in 1 sec

- Real-time: low latency = high throughput
- Batched: high latency, high throughput

Slide based on Designing Machine Learning Systems textbook and https://cs329s.stanford.edu course

ML in research vs. in production

	Research	**Production**
Objectives	Model performance	Different stakeholders have different objectives
Computational priority	Fast training, high throughput	Fast inference, low latency
Data	Static	Constantly shifting

Data

Research	Production
• Clean • Static • Mostly historical data	• Messy • Constantly shifting • Historical + streaming data • Biased, and you don't know how biased • Privacy + regulatory concerns

THE COGNITIVE CODER

By Armand Ruiz, Contributor, InfoWorld | SEP 26, 2017 7:22 AM PDT

The 80/20 data science dilemma

Most data scientists spend only 20 percent of their time on actual data analysis and 80 percent of their time finding, cleaning, and reorganizing huge amounts of data, which is an inefficient data strategy

ML in research vs. in production

	Research	**Production**
Objectives	Model performance	Different stakeholders have different objectives
Computational priority	Fast training, high throughput	Fast inference, low latency
Data	Static	Constantly shifting
Fairness	Good to have (sadly)	Important

Fairness

The Berkeley study found that both face-to-face and online lenders rejected a total of 1.3 million creditworthy black and Latino applicants between 2008 and 2015. Researchers said they believe the applicants "would have been accepted had the applicant not been in these minority groups." That's because when they used the income and credit scores of the rejected applications but deleted the race identifiers, the mortgage application was accepted.

ML in research vs. in production

	Research	**Production**
Objectives	Model performance	Different stakeholders have different objectives
Computational priority	Fast training, high throughput	Fast inference, low latency
Data	Static	Constantly shifting
Fairness	Good to have (sadly)	Important
Interpretability*	Good to have	Important

ML systems vs. traditional software

Traditional software

Separation of Concerns is a design principle for separating a computer program into distinct sections such that each section addresses a separate concern

- Code and data are separate
  - Inputs into the system shouldn't change the underlying code

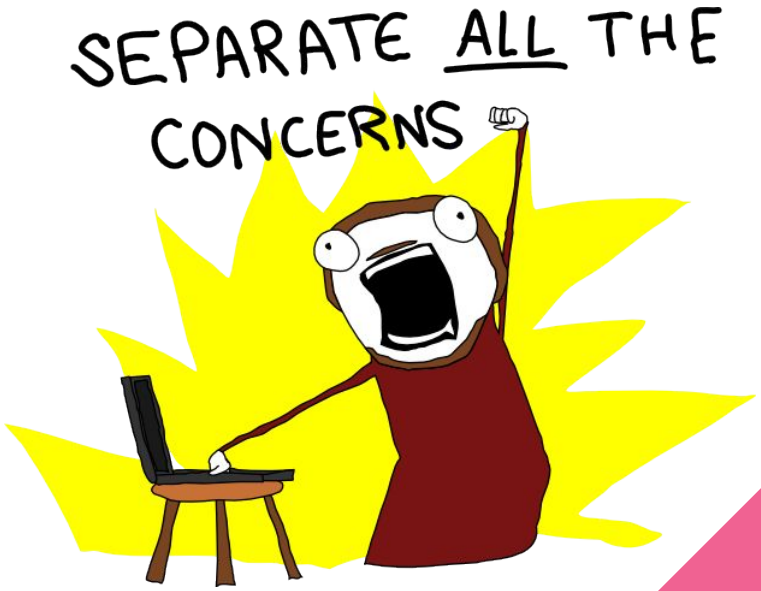


Image by Arda Cetinkaya

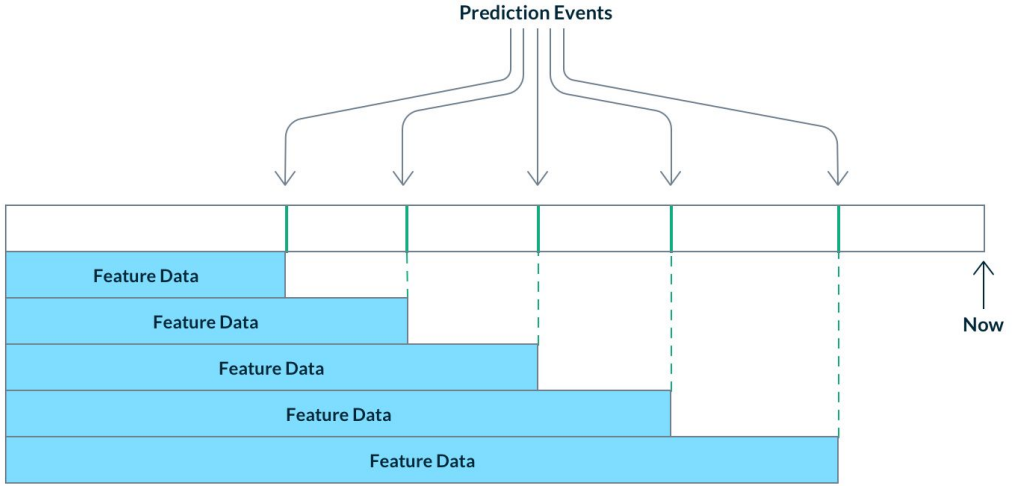
ML systems

- Code and data are tightly coupled
  - ML systems are part code, part data
- Not only test and version code, need to test and version data too

the hard part

Test and version data

- Extremely hard to ensure correctness in time



Timestamp	Label	User ID	Feature Value
2:00	1	1	5
3:00	0	1	19
3:30	0	1	21
5:00	1	1	27
6:00	1	1	42
7:30	0	1	55

Back to the Future: Solving the time-travel problem in machine learning - Tecton

ML systems: version data

- Line-by-line diffs like Git doesn’t work with datasets
- Can’t naively create multiple copies of large datasets
- How to merge changes?

How to …

- Validate data correctness?
- Test features’ usefulness?
- Detect when the underlying data distribution has changed?
- Know if the changes are bad for models without ground truth labels?
- Detect malicious data?
  - Not all data points are equal (e.g. scans of cancerous lungs are more valuable)
  - Bad data might harm your model and/or make it susceptible to attacks

ML systems: data poisoning attacks

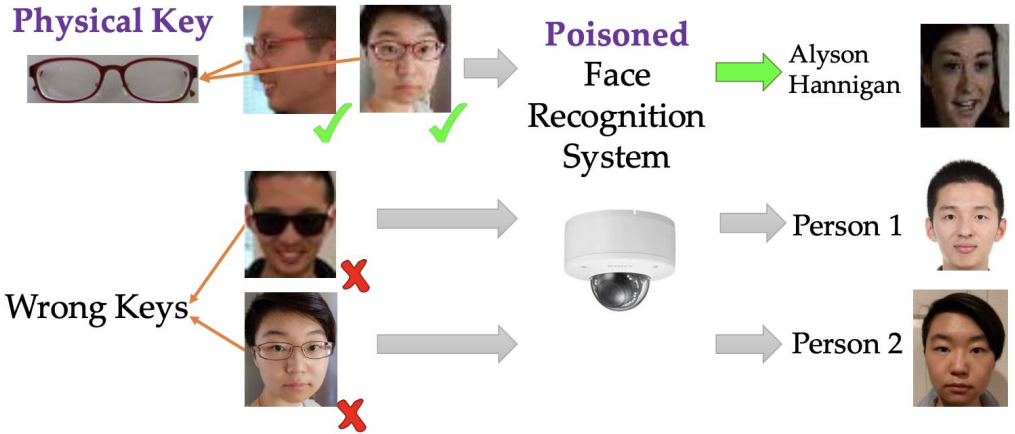
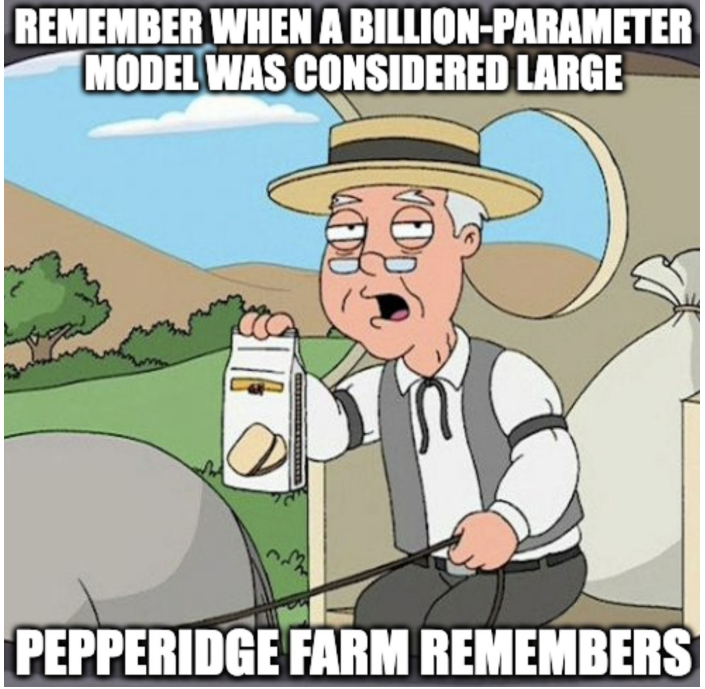


Fig. 1: An illustrating example of backdoor attacks. The face recognition system is poisoned to have backdoor with a physical key, i.e., a pair of commodity reading glasses. Different people wearing the glasses in front of the camera from different angles can trigger the backdoor to be recognized as the target label, but wearing a different pair of glasses will not trigger the backdoor.

Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning (Chen et al., 2017)



SWITCH TRANSFORMERS: SCALING TO TRILLION PARAMETER MODELS WITH SIMPLE AND EFFICIENT SPARSITY

William Fedus*
Google Brain
liamfedus@google.com

Barret Zoph*
Google Brain
barretzoph@google.com

Noam Shazeer
Google Brain
noam@google.com

Engineering challenges with large ML models

- Too big to fit on-device
- Consume too much energy to work on-device
- Too slow to be useful
  - Autocompletion is useless if it takes longer to make a prediction than to type
- If unit/CI tests take hours, the development cycles will stagnate

ML production myths

Myth #1: Deploying is hard

Myth #1: Deploying is hard

Deploying is easy. Deploying reliably is hard

Myth #2: You only deploy one or two ML models at a time

Myth #2: You only deploy one or two ML models at a time

Booking.com: 150+ models, Uber: thousands

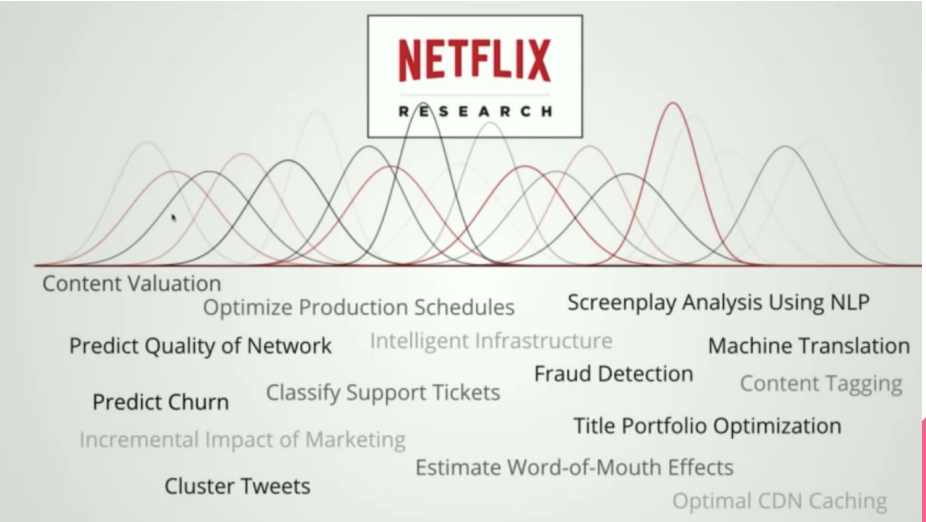


Image from Ville Tuulos (Netflix, Outerbounds)