

Software Engineering for ML Applications (II)

17-313 Fall 2024

Foundations of Software Engineering

<https://cmu-17313q.github.io>

Eduardo Feo Flushing

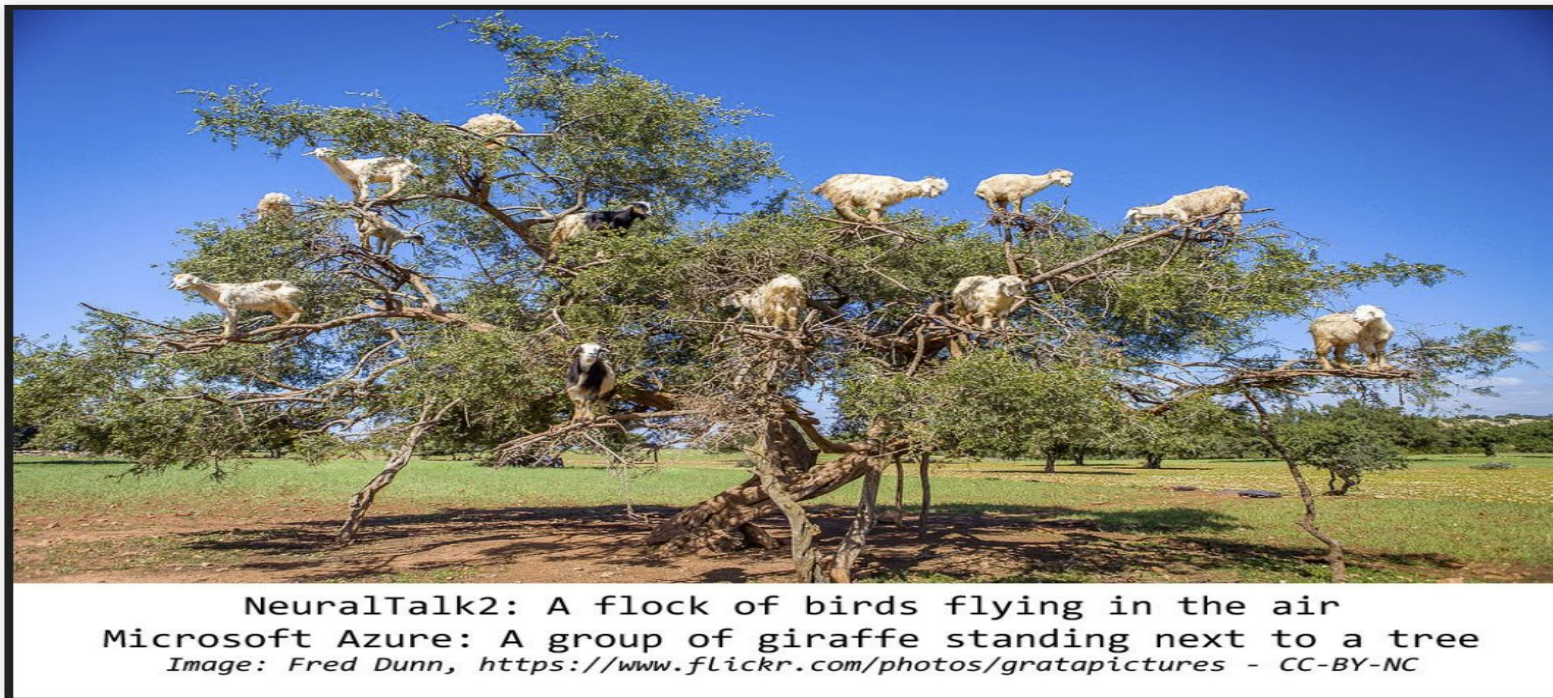
These “AI start-ups” are getting out of hand



Administrivia

- P4A deadline: Tuesday
- Complete the feedback exercise by tonight

ML makes mistakes



Mitigation strategies?

Collecting feedback

Report Incorrect Phishing

If you received a phishing warning but believe that the warning is incorrect, please complete the form below to report the error to Google. Your report will be maintained in accordance with Google's

URL:



I'm not a robot

Comments:
(Optional)

Submit Report



What do you think?

- ☐ This is helpful
- ☐ This isn't relevant
- ☐ Something is wrong
- ☐ This isn't useful

Comments or suggestions?

Optional

The data you provide helps improve Google Search. [Learn more](#)

For a legal issue, [make a legal removal request](#).

Cancel

Send

Updating Models

- Models are rarely static outside the lab
- Data drift, feedback loops, new features, new requirements
- We should consider when and how to update models

Human in the loop

Does Wednesday work for you?

Sure, what time?

Yes, what time?

No, it doesn't.

↩ Reply

➦ Forward



Dr. Emily Slackerman Ackerman

@EmilyEAckerman · [Follow](#)



i (in a wheelchair) was just trapped *on* forbes ave by one of these robots, only days after their independent roll out. i can tell that as long as they continue to operate, they are going to be a major accessibility and safety issue. [thread]



[pittnews.com](#)

Everything we know about the Starship food delivery robots

The white, 2-foot tall battery-powered delivery robots will be sharing the sidewalk with Oakland pedestrians starting sometim...

10:27 PM · Oct 21, 2019

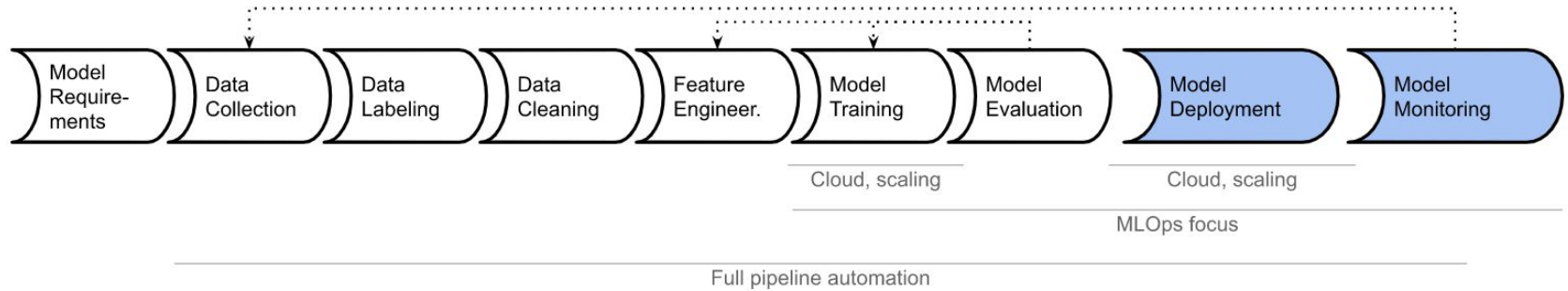


Design for failures/mistakes

- Human-AI interaction design (human in the loop):
- Guardrails
- Mistakes detection and correction
- Undoable actions

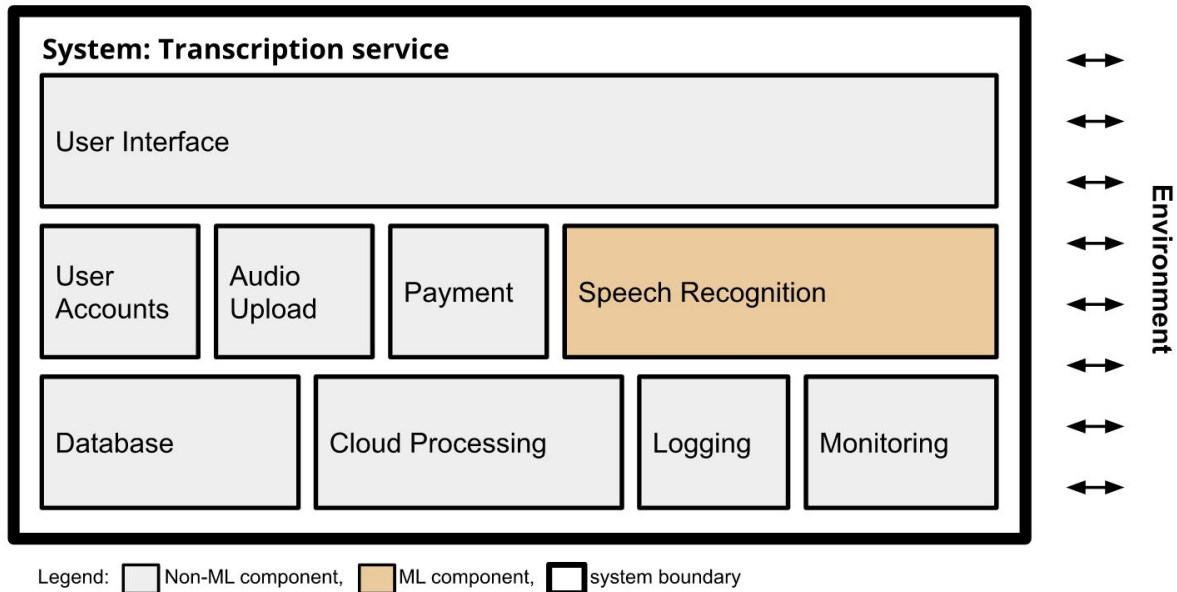
<https://ckaestne.medium.com/safety-in-ml-enabled-systems-b5a5901933ac>

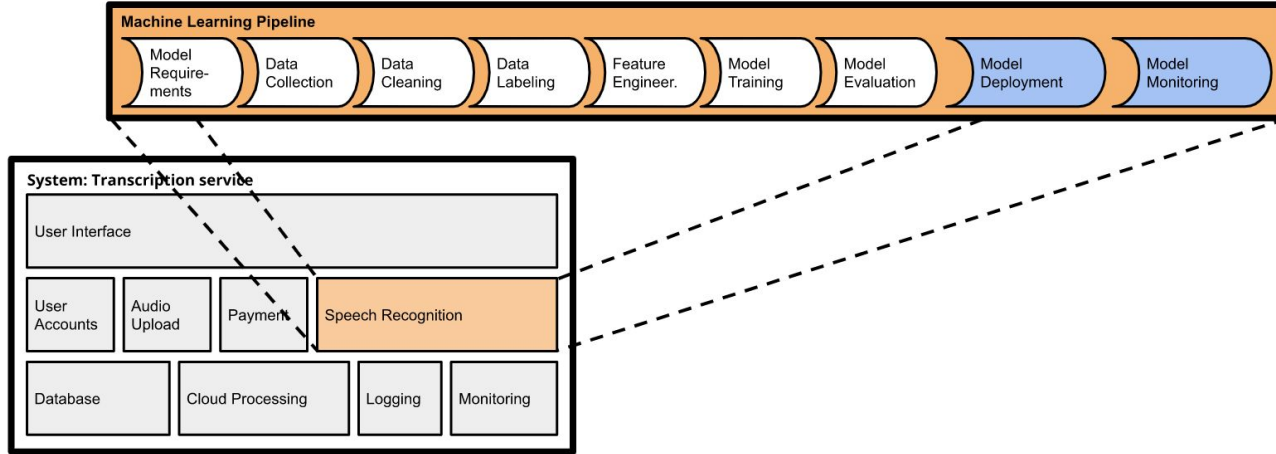
System-wide pipeline

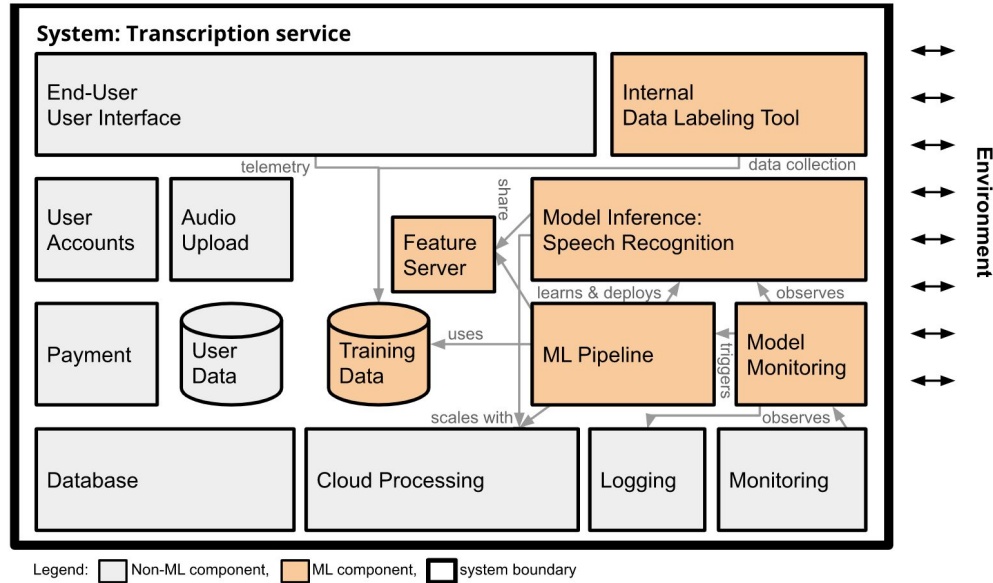


Focus: experimenting, deploying, scaling training and serving, model monitoring and updating

ML models as part of a system







Traditional vs. System-wide ML Pipeline

- Traditional
 - Get labeled data
 - Identify and extract features
 - Split data into training and evaluation set
 - Learn model from training data
 - Evaluate model on evaluation data
 - Repeat, revising features
- With production data
 - Evaluate model on production data; monitor
 - Select production data for retraining
 - Update model regularly

Outline

- Why ML/AI projects fail?
 - Data quality
 - Fairness issues
- What's wrong with the model-centric pipeline?
- **Are there any new challenges?**
- What is ML Ops?

What (real) challenges are there in building and deploying systems with ML?

The road to production: a paradigm shift



T-shaped professionals



I-Shaped

Deep expertise in one topic



Generalist

Broad knowledge of many topics,
but not expert in any



T-Shaped

Expert in one topic and broad
knowledge of other topics

What makes software with ML challenging?

- Lack of specification (unreliability, uncertain output, mistakes)?

Lack of specification

```
/**  
    Return the text spoken within the audio file  
    ????  
*/  
String transcribe(File audioFile);
```

What makes software with ML challenging?

- Lack of specification (unreliability)?
- Complexity?

Complexity in Engineering Systems

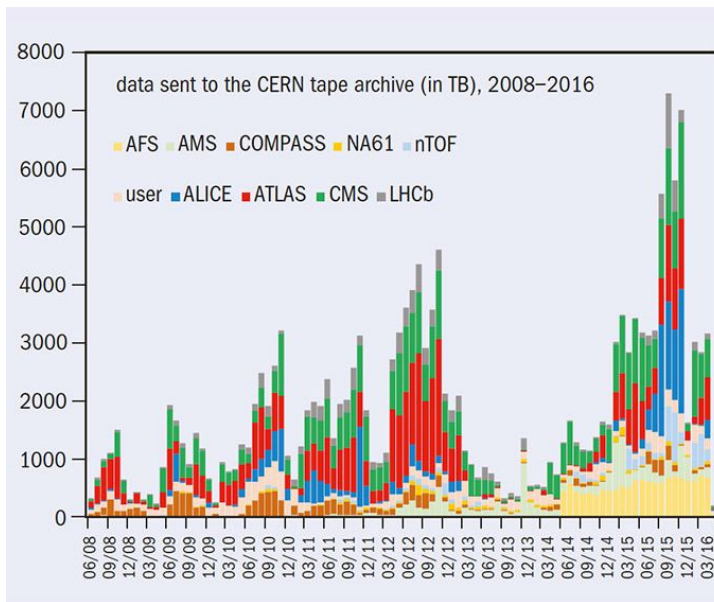
- Automobile ~30K parts
- Airplane ~3M parts
- MS Office ~40M LOC
- Debian ~400M LOC



What makes software with ML challenging?

- Lack of specification (unreliability)?
- Complexity?
- Big Data?

Big Data?



This plot represents the amount of data, in TB, being sent to the CERN archive between 2008 and 2016. The yearly amount of LHC data has gradually increased since 2010 (Run 1, 2010: 12.5 PB, 2011: 19.1 PB, 2012: 27 PB) and during Run 2 (31.5 PB). Image credit: CERN.

What makes software with ML challenging?

- Lack of specification (unreliability)?
- Complexity?
- Big Data?
- Interaction with the environment?

Interaction with the environment: safety

<https://www.alphr.com> › review › smart-toaster ⋮

The Highest-Rated Smart Toasters in 2022 - Alphr Reviews

Aug 19, 2022 — Works on **artificial intelligence (AI)**. A **smart toaster** operates on **artificial intelligence** to detect and control the whole toast-making process, ...



 American Medical Association

AI scribe saves doctors an hour at the keyboard every day

The Permanente Medical Group's rollout of ambient AI scribes to reduce documentation burdens has been deemed a success, saving most of the physicians using it...


Mar 18, 2024



Safety risks?

How can you mitigate these risks?

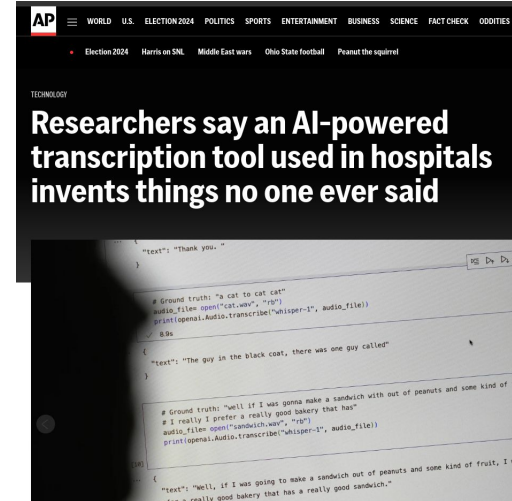
Interaction with the environment: safety

 The Daily Star

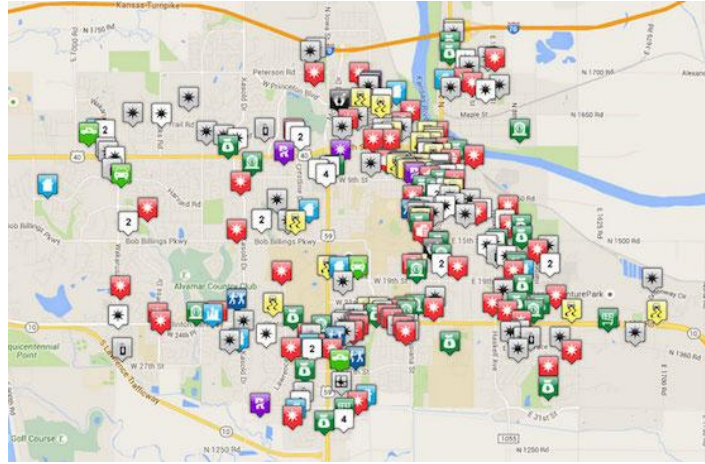
Microwave attempts to murder owner after gaining artificial intelligence 'demon soul'

A YouTuber who tried to resurrect his childhood imaginary friend by giving a microwave artificial intelligence says it tried to kill him.

Apr 28, 2022



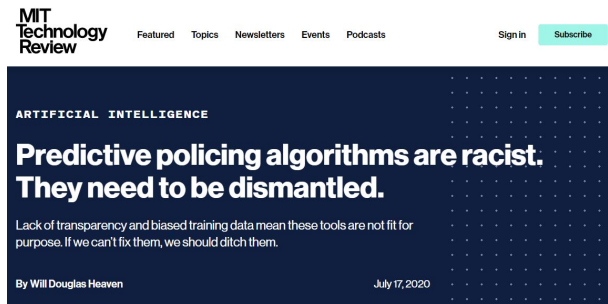
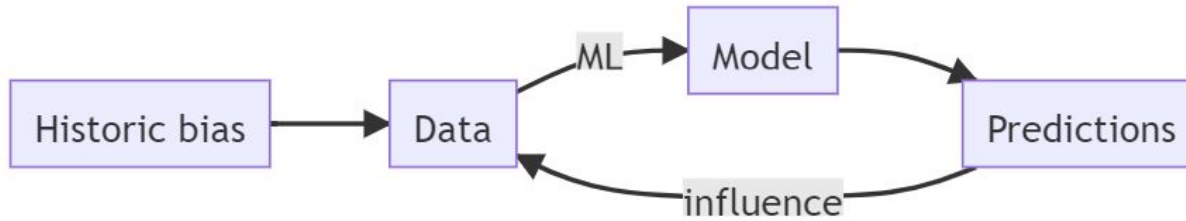
Interaction with the environment: feedback loops



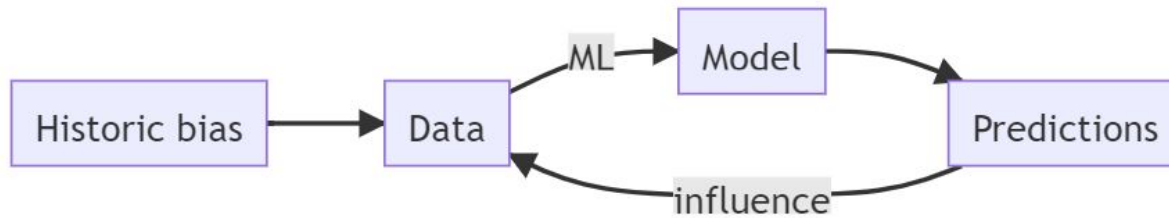
ML Model: Use historical arrest records to predict crime rates by neighborhoods

Used for predictive policing: Decide where to allocate police patrol

Feedback loops



Feedback loops



The New York Times

THE SHIFT

*YouTube Unleashed a Conspiracy
Theory Boom. Can It Be Contained?*

What makes software with ML challenging?

- Lack of specification (unreliability)
- Complexity
- Big Data
- Interaction with the environment

What makes software (systems) with ML challenging?

- **It's not all new**
- Safe software with unreliable components
- Cyber-physical systems
- Non-ML big data systems, cloud systems
- "Good enough" and "fit for purpose" not "correct"
- **We routinely build such systems**
- **ML intensifies our challenges**

Beware of the Automation Paradox

Copyright © IFAC Analysis, Design and
Evaluation of Man-Machine Systems
Baden-Baden, Federal Republic of Germany 1982

IRONIES OF AUTOMATION

L. Bainbridge

Department of Psychology, University College London, London WC1E 6BT, UK

Abstract. This paper discusses the ways in which automation of industrial processes may expand rather than eliminate problems with the human operator. Some comments will be made on methods of alleviating these problems within the 'classic' approach of leaving the operator with responsibility for abnormal conditions, and on the potential for continued use of the human operator for on-line decision-making within human-computer collaboration.

Keywords. Control engineering computer applications; man-machine systems; on-line operation; process control; system failure and recovery.

The more efficient the automated system, the more crucial the human contribution of the operators.

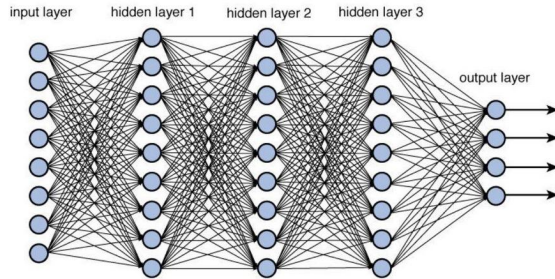
ML Component Tradeoffs

Qualities of ML Components

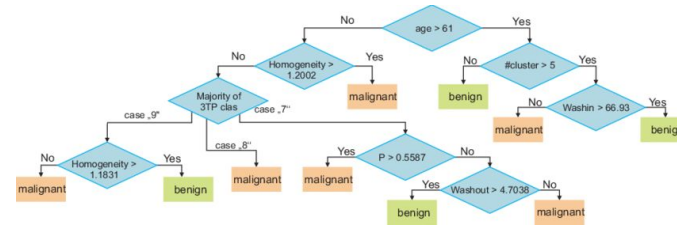
- Accuracy
- Capabilities (e.g. classification, recommendation, clustering...)
- Amount of training data needed
- Inference latency
- Learning latency; incremental learning?
- Model size
- Explainable? Robust?

Understanding Capabilities and Tradeoffs

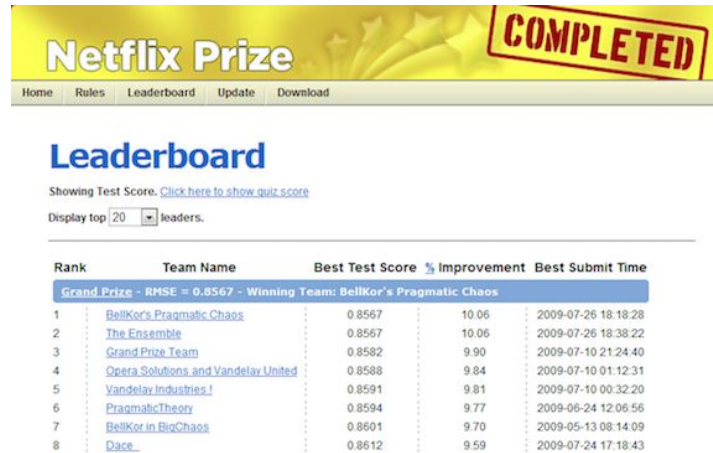
- Deep Neural Networks



- Decision Trees



Trade-offs: Cost vs Accuracy

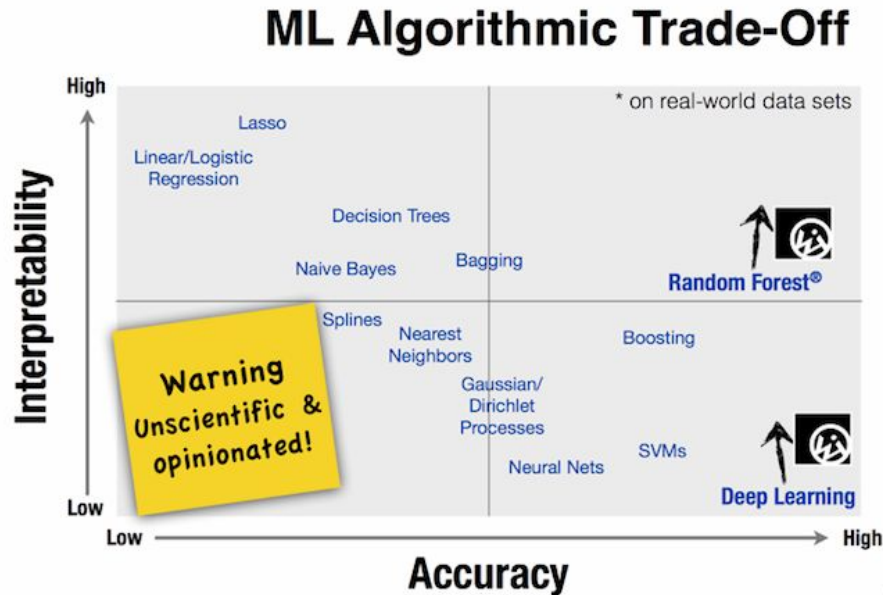


The screenshot shows the Netflix Prize Leaderboard interface. At the top, there is a yellow banner with the text 'Netflx Prize' and a red stamp that says 'COMPLETED'. Below the banner is a navigation bar with links: Home, Rules, Leaderboard, Update, and Download. The main heading is 'Leaderboard'. Below this, it says 'Showing Test Score. [Click here to show quiz score](#)'. There is a dropdown menu for 'Display top' set to '20' and the text 'leaders.'. Below this is a table with the following columns: Rank, Team Name, Best Test Score, % Improvement, and Best Submit Time. The table shows the top 8 teams, with the winning team 'BellKor's Pragmatic Chaos' at rank 1.

Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos				
1	BellKor's Pragmatic Chaos	0.8567	10.06	2009-07-26 18:18:28
2	The Ensemble	0.8567	10.06	2009-07-26 18:38:22
3	Grand Prize Team	0.8582	9.90	2009-07-10 21:24:40
4	Opera Solutions and Vandelay United	0.8588	9.84	2009-07-10 01:12:31
5	Vandelay Industries !	0.8591	9.81	2009-07-10 00:32:20
6	PragmaticTheory	0.8594	9.77	2009-06-24 12:06:56
7	BellKor in BioChaos	0.8601	9.70	2009-05-13 08:14:09
8	Dace	0.8612	9.59	2009-07-24 17:18:43

"We evaluated some of the new methods offline but the additional accuracy gains that we measured did not seem to justify the engineering effort needed to bring them into a production environment."

Trade-offs: Accuracy vs Interpretability



System Architecture Tradeoffs

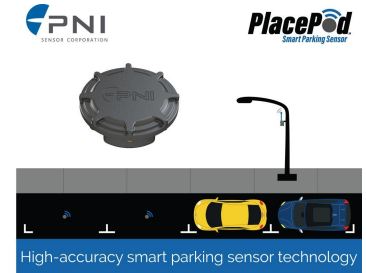
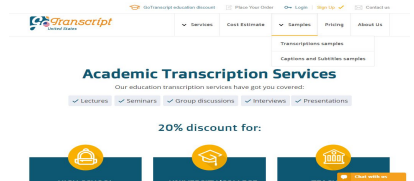
Activity

Pick one scenario based on where you are seating

- Transcription Services (front rows)
- Parking Sensor (middle rows)
- Surge Prediction (back rows)

Discuss in groups these questions:

- Where should the model be deployed? e.g., in the cloud, on-premises, on directly on the devices?
- What are the key factors influencing this choice (e.g., latency, computational power, data privacy)?



Where should the model live?

Laptop

Local
Server

Cloud

Academic
Transcriptions



Where should the model live?

Car



Phone



Cloud



Surge
Prediction

Where should the model live?

Pod



Gateway



Cloud



Car
Detector

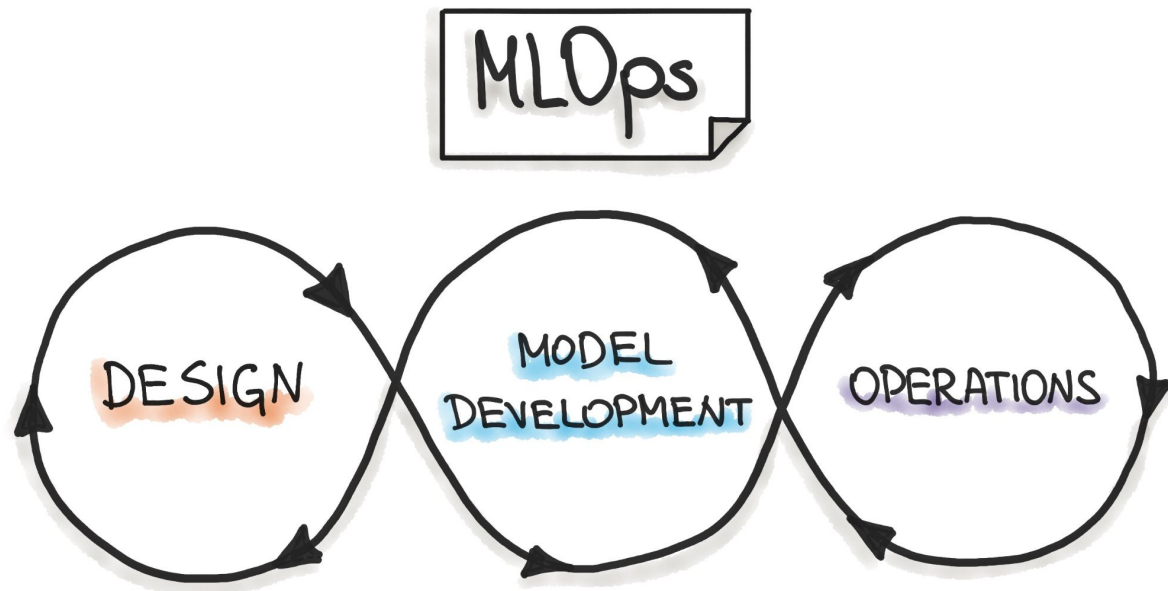
Typical Designs

- Static intelligence in the product
 - difficult to update
 - good execution latency
 - cheap operation
 - offline operation
 - no telemetry to evaluate and improve
- Client-side intelligence
 - updates costly/slow, out of sync problems
 - complexity in clients
 - offline operation, low execution latency

Considerations for deployment

- How much data is needed as input for the model?
- How much output data is produced by the model?
- How fast/energy consuming is model execution?
- What latency is needed for the application?
- How big is the model? How often does it need to be updated?
- Cost of operating the model? (distribution + execution)
- Opportunities for telemetry?
- What happens if users are offline?

MLOps



MLOps

- Many vague buzzwords, often not clearly defined
- MLOps: Collaboration and communication between data scientists and operators, e.g.,
 - Automate model deployment
 - Model training and versioning infrastructure
 - Model deployment and monitoring

MLOps Overview

- Integrate ML artifacts into software release process, unify process (i.e., DevOps extension)
- Automated data and model validation (continuous deployment)
- Continuous deployment for ML models: from experimenting in notebooks to quick feedback in production
- Versioning of models and datasets
- Monitoring in production

MLOps Tools (examples)

- Model versioning and metadata: MLFlow, Neptune, ModelDB, WandB, ...
- Model monitoring: Fiddler, Hydrosphere
- Data pipeline automation and workflows: DVC, Kubeflow, Airflow
- Model packaging and deployment: BentoML, Cortex
- Distributed learning and deployment: Dask, Ray, ...
- Feature store: Feast, Tecton
- Integrated platforms: Sagemaker, Valohai, ...
- Data validation: Cerberus, Great Expectations, ...

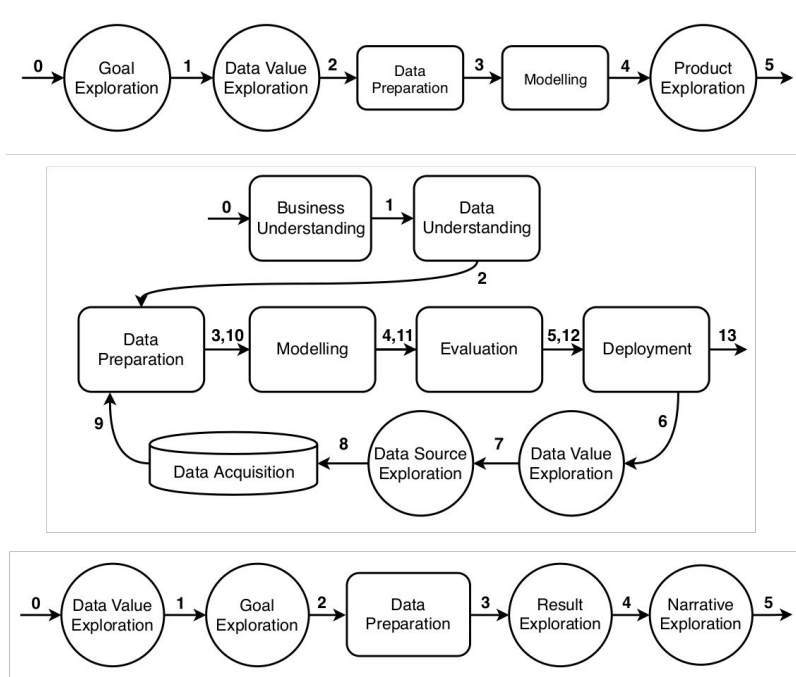
Long list: <https://github.com/kelvins/awesome-mlops>

Process for AI-Enabled Systems

Data Science is Iterative and Exploratory

- Science mindset: start with rough goal, no clear specification, unclear whether possible
- Heuristics and experience to guide the process
- Try and error, refine iteratively, hypothesis testing
- Go back to data collection and cleaning if needed, revise goals

Different Trajectories



- Goal exploration: finding business goals which can be achieved in a data-driven way
- Data source exploration: discovering new and valuable sources of data
- Data value exploration: finding out what value might be extracted from the data
- Result exploration: relating data science results to the business goals
- Narrative exploration: extracting valuable stories (e.g., visual or textual) from the data
- Product exploration: finding ways to turn the value extracted from the data into a service or app that delivers something new and valuable to users and customers.
- Data acquisition: obtaining or creating relevant data, for example by installing sensors or apps

Martínez-Plumed et al. "CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories." IEEE Transactions on Knowledge and Data Engineering (2019)

Trajectories

- Not every project follows the same development process, e.g.
 - Small ML addition: Product first, add ML feature later
 - Research only: Explore feasibility before thinking about a product
 - Data science first: Model as central component of potential product, build system around it
- Different focus on system requirements, qualities, and upfront planning
- Manage interdisciplinary teams and different expectations

Computational Notebooks

- Origins in "literate programming", interleaving text and code, treating programs as literature (Knuth 84)
- First notebook in Wolfram Mathematica 1.0 in 1988
- Document with text and code cells, showing execution results under cells
- Code of cells is executed, per cell, in a kernel
- Many notebook implementations and supported languages, Python + Jupyter currently most popular

```
# load data collected from team1
import pandas as pd

url = 'http://128.2.25.78:8080/private/log1.clean'
df = pd.read_csv(url)
df.head()
```

	dayIdx	user	userAvgTime	location	dow	isWeekend	time
0	0	Pittsburgh66Correy	7.045001	Pittsburgh	6	True	0.000000
1	1	Pittsburgh66Correy	7.045001	Pittsburgh	7	True	6.883333
2	2	Pittsburgh66Correy	7.045001	Pittsburgh	1	False	6.816667
3	3	Pittsburgh66Correy	7.045001	Pittsburgh	2	False	7.383333
4	4	Pittsburgh66Correy	7.045001	Pittsburgh	3	False	0.000000

Data was preprocessed externally, identifying the time at a given day when the light was first turned on (12pm). Weather and sunrise information is not included here, though that'd be important. If the light was this morning (quite common), 0 is recorded.

```
[ ] # just data encoding and splitting X and Y

X = df.drop(['time'], axis=1)
YnonZero = df['time'] > 0
Y = df['time']

from sklearn import preprocessing
# leDate = preprocessing.LabelEncoder()
# leDate.fit(X['date'])
# leDate.transform(X['date'])

X=X.apply(preprocessing.LabelEncoder().fit_transform)
X
```

Notebooks Support Iteration and Exploration

- Quick feedback, similar to REPL
- Visual feedback including figures and tables
- Incremental computation: running individual cells
- Quick and easy: copy paste, no abstraction needed
- Easy to share: document includes text, code, and results

Brief Discussion: Notebook Limitations and Drawbacks?



Process Activities for ML?

- Testing script
 - Existing model: Automatically evaluate model on labeled training set; multiple separate evaluation sets possible, e.g., for slicing, regressions
 - Training model: Automatically train and evaluate model, possibly using cross-validation; many ML libraries provide built-in support
 - Report accuracy, recall, etc. in console output or log files
 - Deploy learning and evaluation tasks to cloud services
 - Optionally: Fail test below bound (e.g., accuracy < 0.9 ; accuracy $<$ last accuracy)
- Version control test data, model and test scripts, ideally also learning data and learning code (feature extraction, modeling, ...)
- Continuous integration tool can trigger test script and parse output, plot for comparisons (e.g., similar to performance tests)
- Optionally: Continuous deployment to production server

Machine Learning in Production

Want to know more about it?

mlip-cmu

Machine Learning in Production @ CMU

Find resources related to teaching and research on how to build, deploy, assure, and maintain software products with machine-learned models. These cover the entire lifecycle from a prototype ML model to an entire system deployed in production, not just models or notebooks. Covers also the **responsible ML engineering** of such systems (safety, security, fairness, transparency) and **MLOps**.*

All materials (book, slides, assignments, bibliography) are released under creative commons licenses. We hope that this fosters teaching and research on these topics.

Maintained by [Christian Kaestner](#).

<https://mlip-cmu.github.io/>

Summary

- Production AI-enabled systems require a *whole system perspective* beyond just the model or the pipeline
- Machine learning brings new challenges and intensifies old ones
- Building ML systems need team efforts
- Collaborative culture among Software Engineers, Data Scientists, Stakeholders is necessary