Getting machine learning to production

The full cycle of machine learning projects

Final Project, 06/06/2023

- Nick Fioravanti, Justin Serwinski; Jalen Winfield, Deep Learning in Neural Networks with Neural Data Analysis: EEG
- Alexander Wang, A.I. Philosophy and Ethical Alignment
- Josh Miller; Marc Mounzer; Kevin Chavez, Kasonde Chewe, Computational Neuroscience
- Christopher Campbell: Application in Biomedicine
- Eleni Alexakis, Hadis Jami, Tony Okeke, Medical Imaging

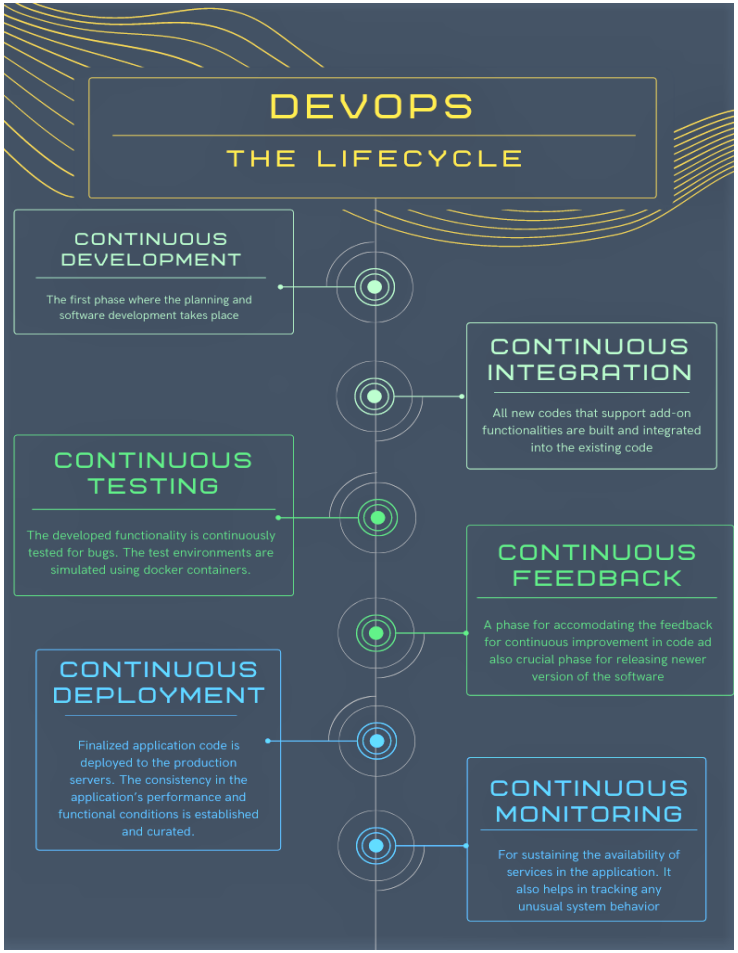
- DevOps / MLOps Lifecycle
- Automated Machine Learning (AutoML): easily build

DevOps / MLOps Lifecycle

DevOps: a set of practices that combines software development and IT operations

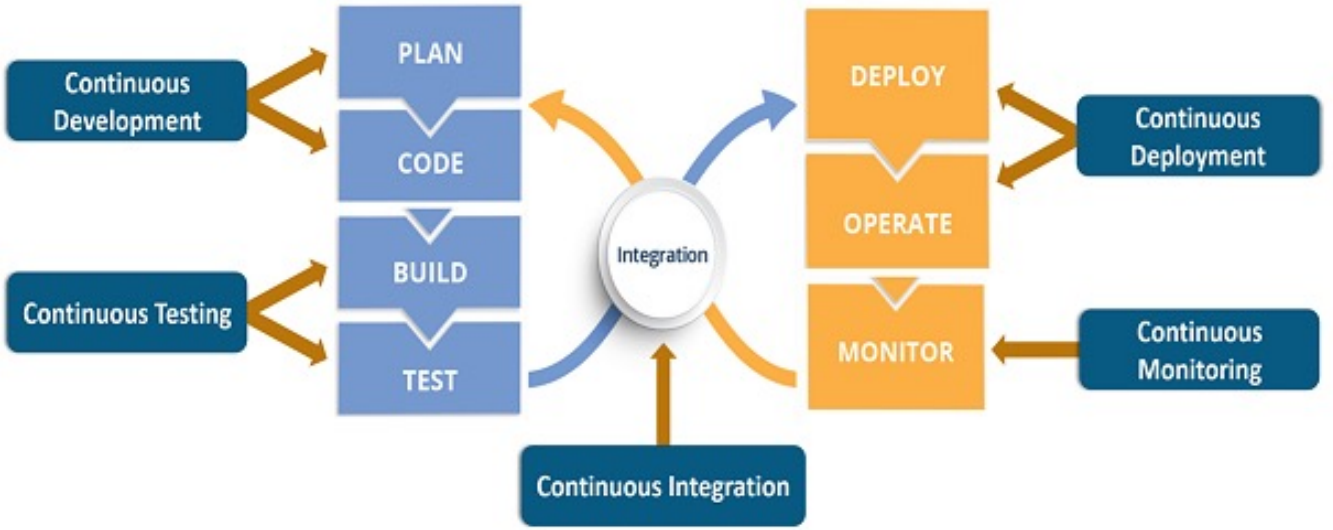
Machine Learning Operations

MLOps: apply DevOps principles to ML systems



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

DevOps / MLOps Lifecycle

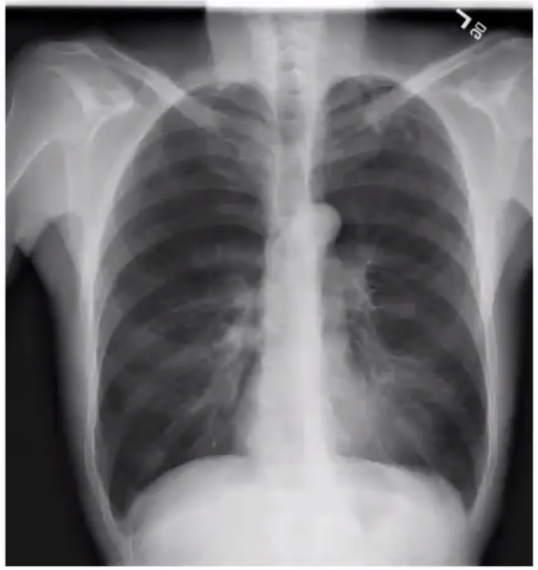


recent R&D progress

[usability247.com]

- Design
  - choosing the right deep network **architecture**
  - links with **splines**, signal processing, approximation theory
- Build
  - insights into how stochastic gradient descent reaches such good solutions when the optimization landscape is so highly **nonconvex**
  - links with **differential equations**
  - more **data-efficient** learning mechanisms
- Test
  - **formal verification** for deep networks
  - statistical **performance guarantees**

AI detects pathologies in chest x-rays at the level of radiologists



2 billion chest x-rays per year.

Deep Learning achieved radiologist-level performance on 11 pathologies.

[with Pranav Rajpurkar, Jeremy Irvin, Matt Lungren, Curt Langlotz, Bhavik Patel.]

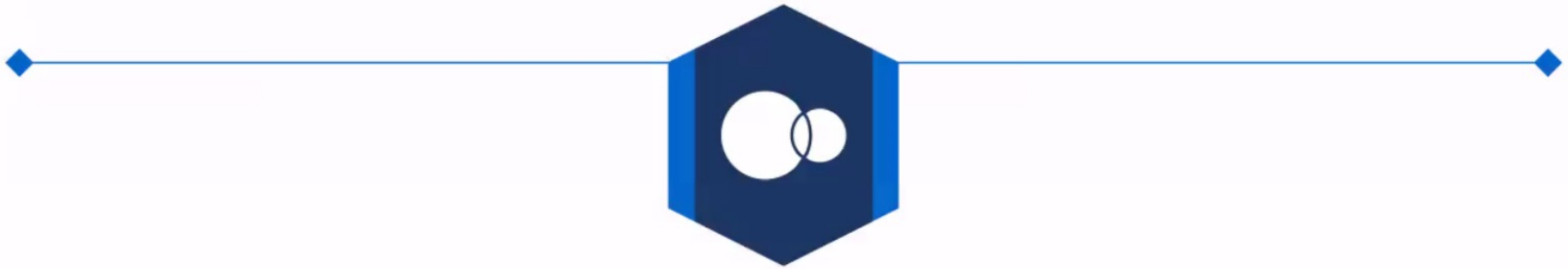
Why aren’t these systems widely deployed to hospitals yet?

Bridging the PoC to Production gap

Small Data

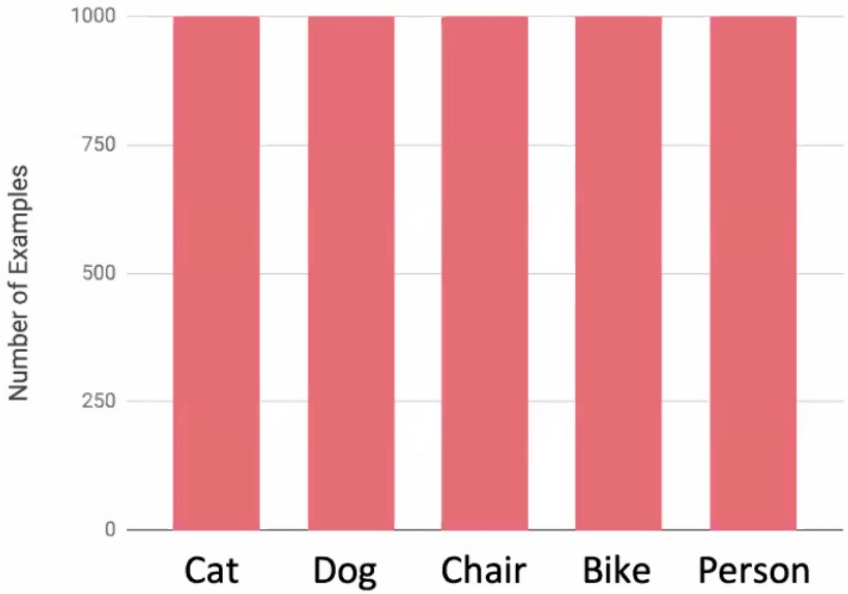
Generalizability and robustness

Small data: Moving beyond big data

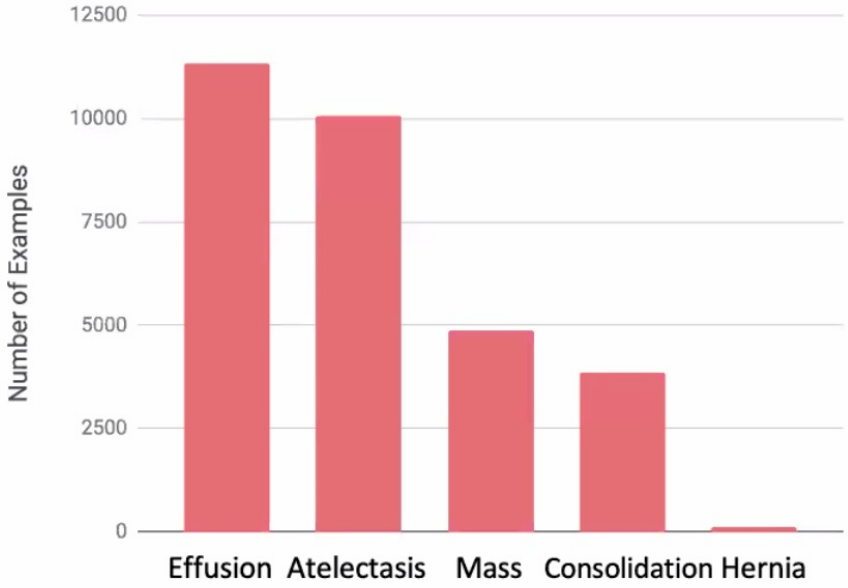


Small data and rare occurrences

ML works well when the data distribution is this:

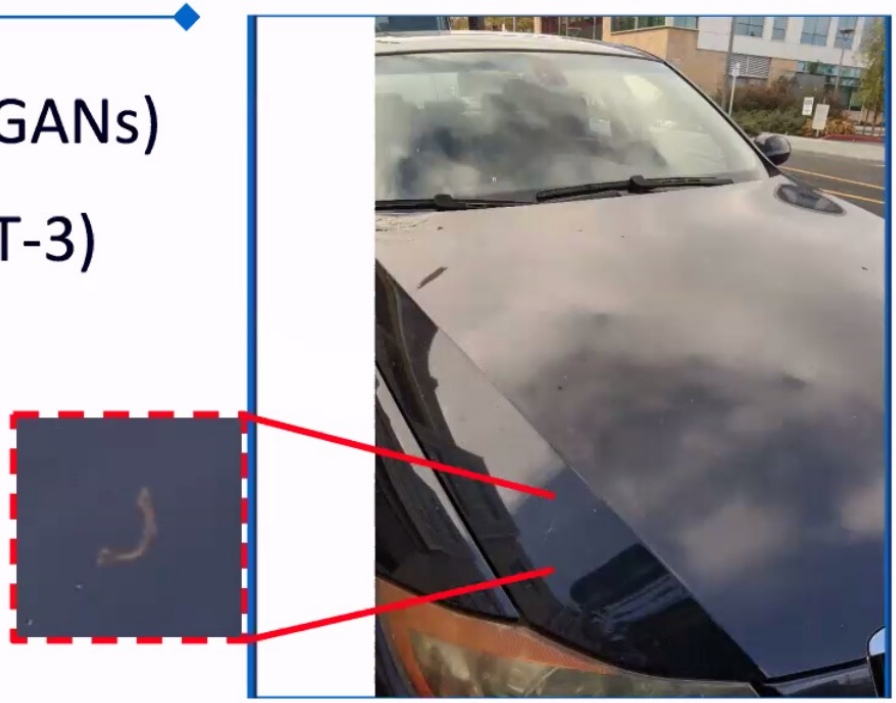


Not so well when it is this:



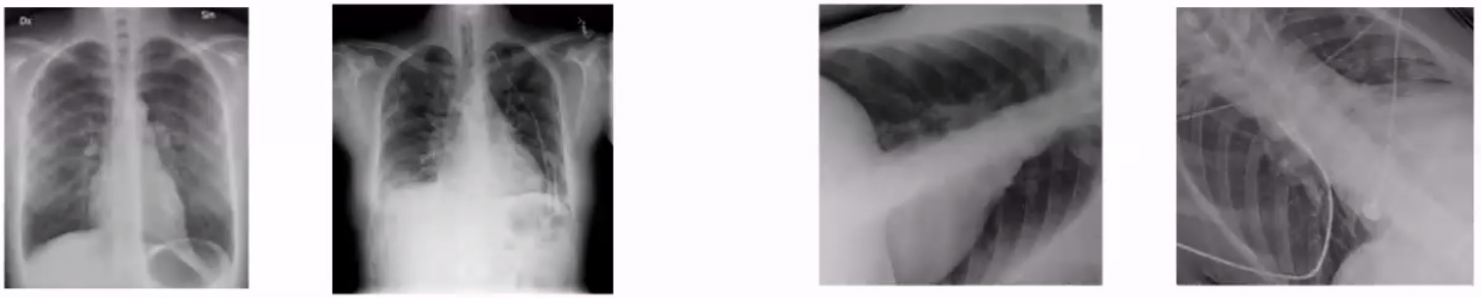
Small data algorithms

- Synthetic data generation (e.g. GANs)
- One/Few-shot learning (e.g. GPT-3)
- Self-supervised learning
- Transfer learning
- Anomaly detection



Why isn't AI X-ray systems widely deployed? Generalization and Robustness

A model that works according to a published paper often does not work in a production setting.



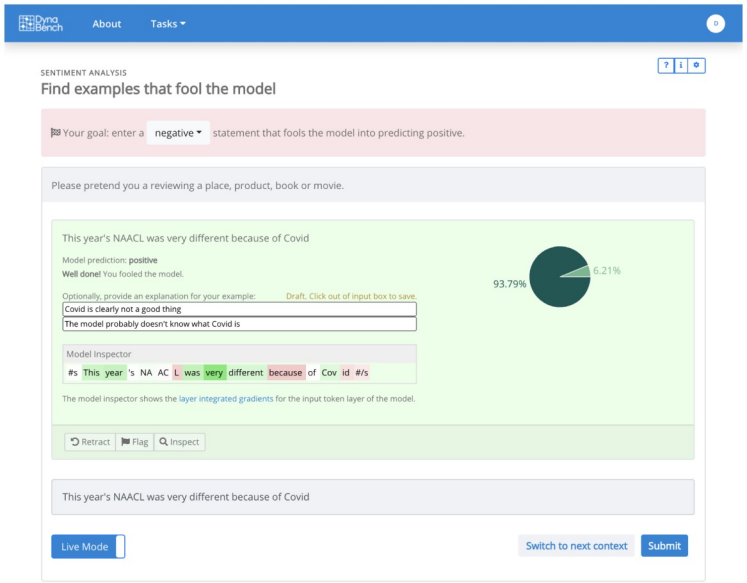
Huge gap between what works in a research lab vs. what will run in production.
This is true not just for healthcare.

Will your model generalize to a different dataset than what it was trained on?

Machine Learning in production: active benchmarking

Due to the rapid progress in model development, beating benchmarks has become a matter of months. The high-performing models nonetheless often fail in real-world scenarios. Dynamic Benchmarking, where datasets are continuously updated by human users, are a solution to make benchmarks more useful.

- Dynabench is a web-based open-source tool that allows users to propose difficult examples that fool the model or make it very uncertain. These examples are then validated by expert linguists and crowdworkers.
- The collected data can be used to both evaluate current state-of-the-art models and train other models.
- The aim of dynamic benchmarking is to create a virtuous cycle where models are improved to be able to deal with harder examples. Then, it becomes increasingly harder to fool the models, which hopefully evolve to be robust to the worst case scenarios that are encountered in the real world.



stateof.ai 2021

Machine Learning in production: distribution shifts

Two new datasets to deal with distribution shifts: WILDS and Shifts.

- A distribution shift happens when data at test/deployment time is different from the training data. In production, this often happens in the form of concept drifts, where the test data gradually changes over time.
- As ML is increasingly used in real-world applications, the need for a solid understanding of distributional shifts becomes paramount. This begins with designing challenging benchmarks.
  - A team from several American and Japanese universities and companies have built WILDS, a benchmark of 10 datasets of distributional shifts in tumor identification, wildlife monitoring, satellite imaging, and more.
  - Shifts, developed by the Russian Yandex, is more industry-focused, and includes 3 tasks: weather predicti Shifts Challenge ranslation and vehicle motion prediction.

WILDS

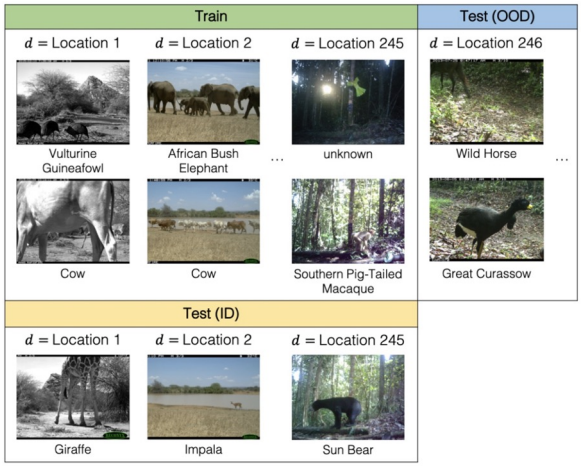


Table 1: Number of samples in the canonical partitioning of Weather Prediction dataset.

	Data	# of samples					
		Total	Tropical	Dry	Mild Temperate	Snow	Polar
Training	`train`	**3,129,592**	416,310	690,284	2,022,998	0	0
Development	`dev_in`	**50,000**	6,641	10,961	32,398	0	0
	`dev_out`	**50,000**	0	0	0	50,000	0
	`dev`	**100,000**	6,641	10,961	32,398	50,000	0
Evaluation	`eval_in`	**561,105**	74,406	123,487	363,212	0	0
	`eval_out`	**576,626**	0	0	0	525,967	50,659
	`eval`	**1,137,731**	74,406	123,487	363,212	525,967	50,659

stateof.ai 2021

Machine Learning in production: underspecification

A more pernicious problem in ML systems is underspecification: Models trained and tested successfully on the same data, but using different random seeds, can behave differently on real-world data.

- Researchers from Google, MIT, UCSD and Stanford illustrate this problem with examples from computer vision, medical imaging, NLP, clinical risk prediction based on health records, and medical genomics.
- While they identify the problem and illustrate it, they do not have a definitive solution, and hope to spur interest in improving the machine learning pipeline to tackle the underspecification challenge. But it is unclear whether it can be tackled at all.

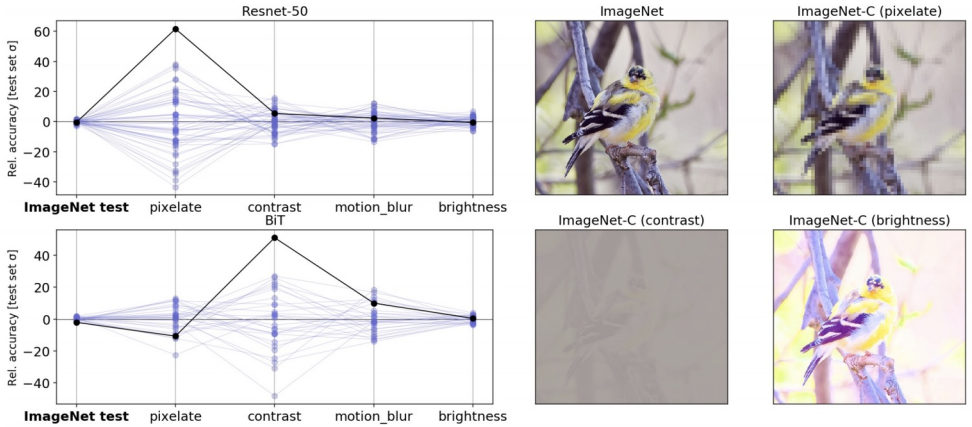
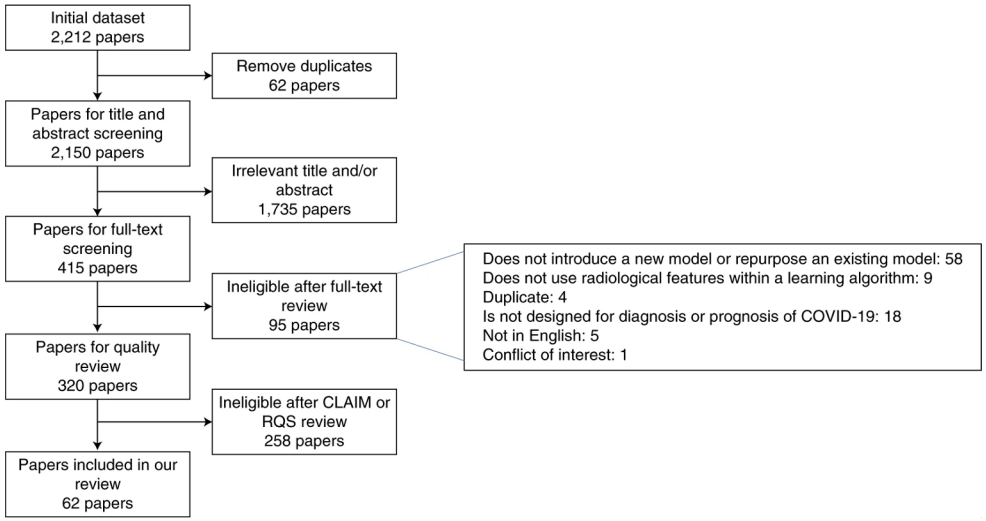


Figure 4: **Image classification model performance on stress tests is sensitive to random initialization in ways that are not apparent in iid evaluation. (Top Left)** Parallel axis plot showing variation in accuracy between identical, randomly initialized ResNet 50 models on several ImageNet-C tasks at corruption strength 5. Each line corresponds to a particular model in the ensemble; each each parallel axis shows deviation from the ensemble mean in accuracy, scaled by the standard deviation of accuracies on the "clean" ImageNet test set. On some tasks, variation in performance is orders of magnitude larger than on the standard test set. **(Right)** Example image from the standard ImageNet test set, with corrupted versions from the ImageNet-C benchmark.

Machine Learning in production: beware of bad data

Despite a loud call to arms and many willing participants, the ML community has had surprisingly little positive impact against Covid-19. One of the most popular problems - diagnosing coronavirus pathology from chest X-ray or chest computed tomography images using computer vision - has been a universal clinical failure.

- A systematic review of all papers published in 2020 that reported using ML for diagnosis and prognostication of Covid-19 found that *"none of the reviewed literature reaching the threshold of robustness and reproducibility essential to support utilization in clinical practice."* There were many methodological, dataset, and bias issues.
- For example, 25% of papers used the same pneumonia control dataset to compare adult patients without mentioning that it consists of kids aged 1-5.

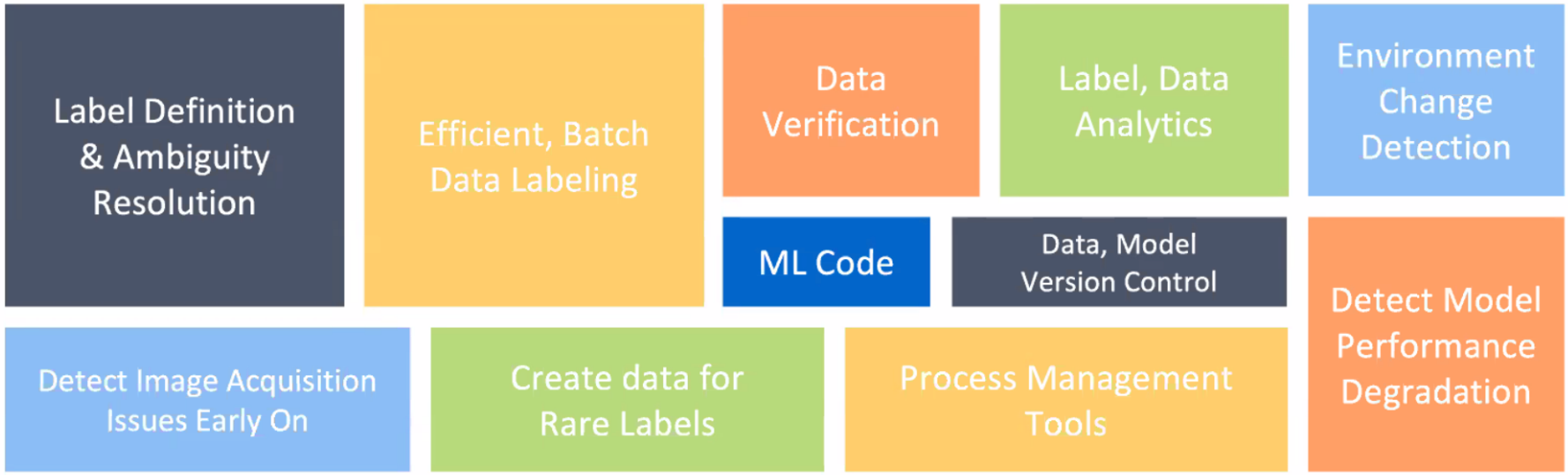


The full cycle of machine learning projects

Production AI projects require more than ML code

The ML (machine learning) model or ML code is only a small part of the puzzle.

Production AI projects require more than ML code



The ML (machine learning) model or ML code is only a small part of the puzzle.

Sculley et al. Hidden Technical Debt in Machine Learning Systems, 2015

The full cycle of machine learning projects

Phases of an AI project:

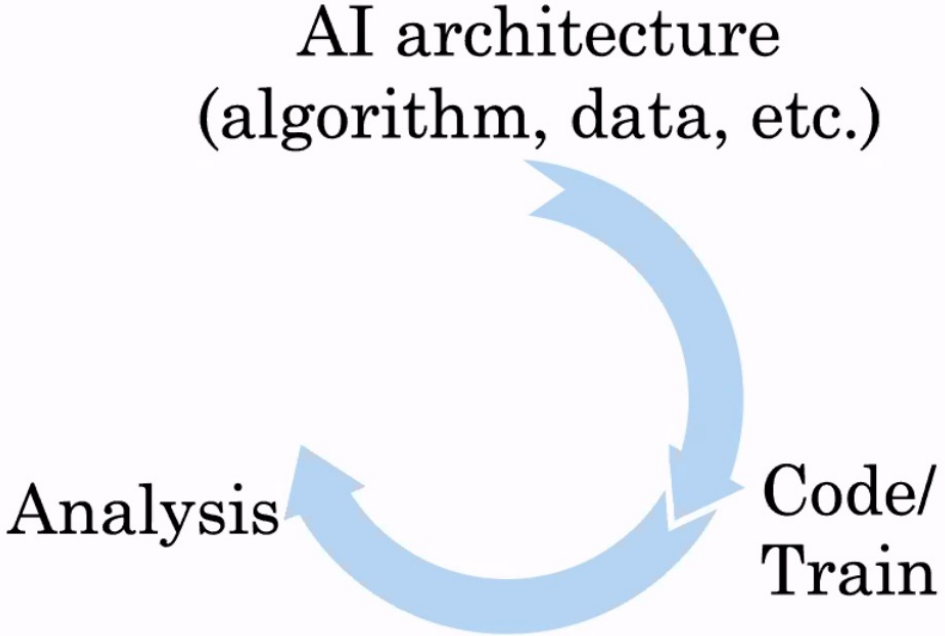
[Thanks to Dillon Laird, Daniel Bibireata]

Deployment: Run in production to create value

- Cloud or edge implementation + monitoring tools.
- Initial deployment to permit analysis of results and tuning.
  - Shadow deployment (E.g., X-ray diagnosis system not used to make any decisions, but only “shadows” a doctor)
  - Canary deployment (rolled out only to small subset of users)
- Ramp up deployment
- Long term monitoring and maintenance

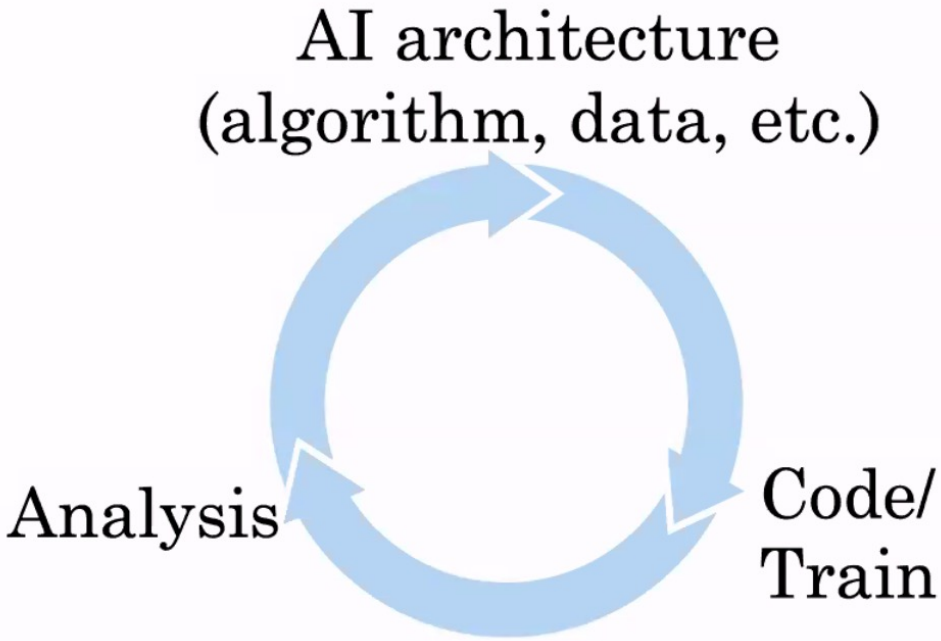
Modeling: Build/train an AI model

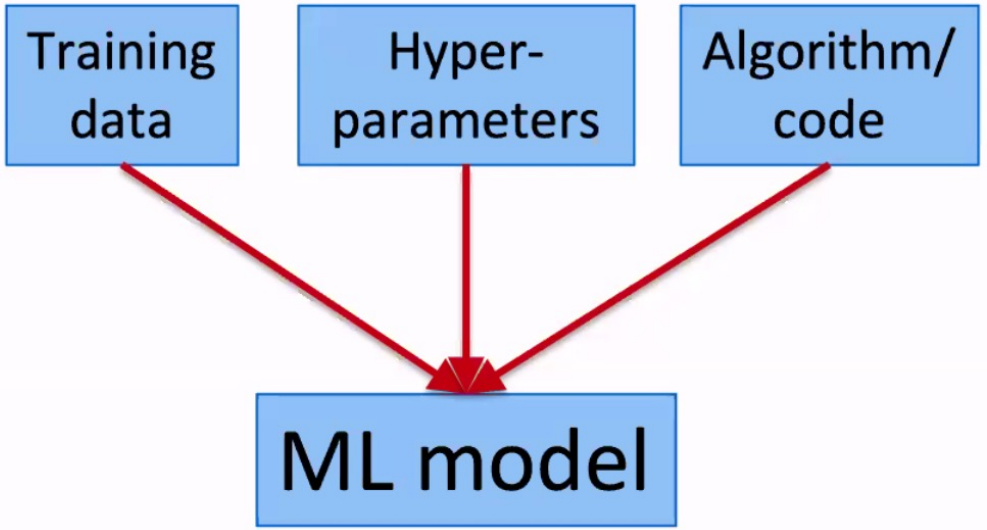
Highly iterative process: More like debugging than development



Modeling: Build/train an AI model

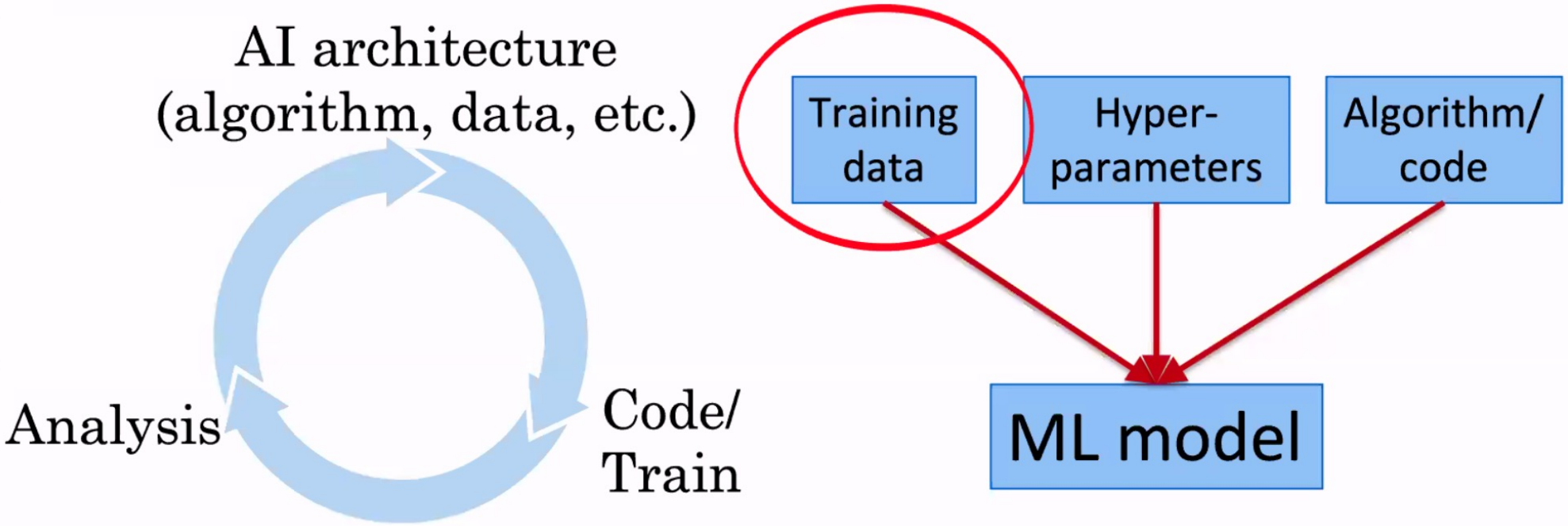
Highly iterative process: More like debugging than development





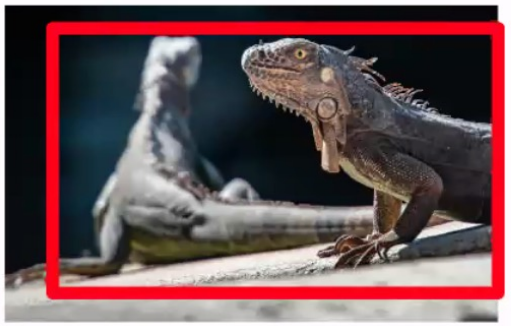
Modeling: Build/train an AI model

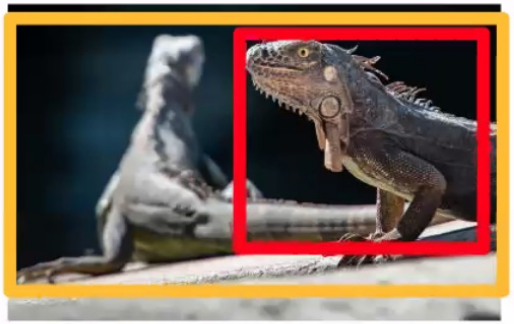
Highly iterative process: More like debugging than development

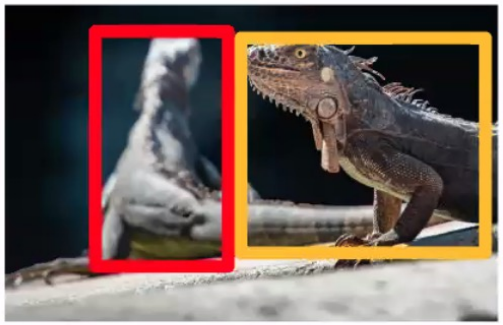


Data: Acquire data for model

- Inventory data sources. E.g., what user data is there?
  - Don't wait for "perfect" data to get started
- Decide on clear data definitions
  - E.g., "Draw bounding box around iguanas"
  - Examples in speech recognition, agriculture, ….
  - What if even experts don't agree? Rethinking human-level performance.







- Automated Machine Learning (AutoML): easily build

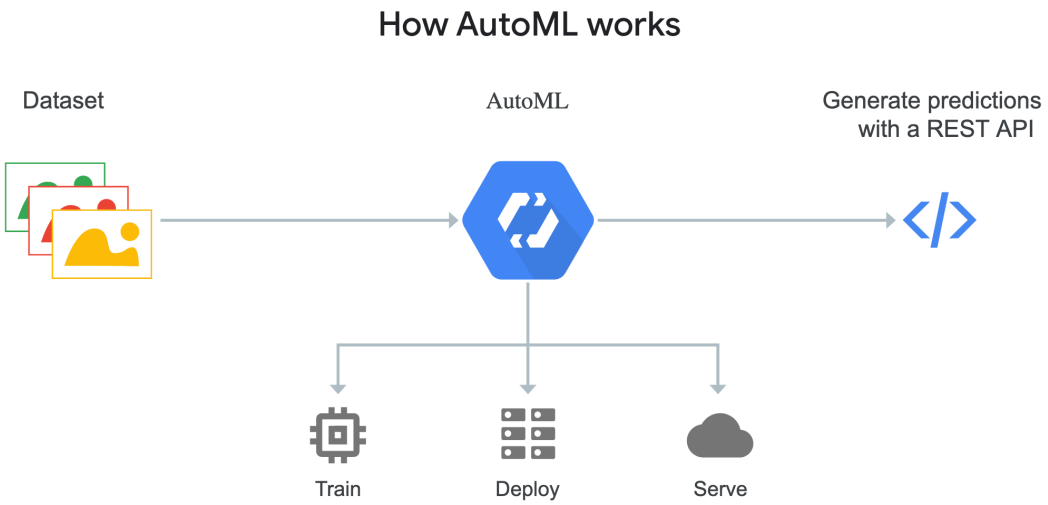
AutoML: Automated Machine Learning

AutoML has become the new trend in the field of machine learning. Its aim to automate the whole cycle of Machine Learning and Deep Learning projects

It aims to lower the need for skilled Data Scientists to build Machine Learning and Deep Learning models

Automated Deep Learning
- no coding experience

Google Cloud Platform



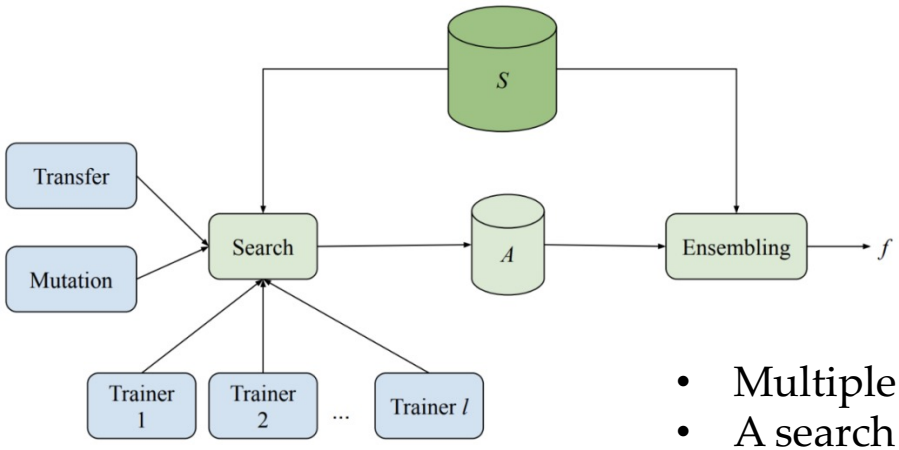
Google Cloud AutoML: https://cloud.google.com/automl/

AutoML: Model Search

Google AI

An Open Source Platform for Finding Optimal ML Models

02/19/2021



- Multiple trainers
- A search algorithm
- A transfer learning
- A database to store various evaluated models

Model Search

Advanced, but paid: **AutoML Tables**

https://github.com/google/model_search

AutoNLP from Hugging Face

Pros of AutoML

- **Accessibility**: People from other domains without much experience in ML can use AutoML for their project without worrying much about the exhaustive and redundant processes of data preparation and other processing stages including model selection.
- **Efficiency**: AutoML can save much of their time in redundant steps which could have been utilized in making the models more optimized by tuning the hyperparameters.
- **Less Errors**: Codes are often prone to errors. AutoML helps to reduce human errors. You wouldn't have to worry about some errors in the former stages which would eventually ruin your future predictions.
- **Cost savings**: This will be extremely useful for small companies or startups that can't afford to hire an ML professional to build their recommendation or sales forecast systems.
- **Fulfill Industry Demands**: AutoML will make the process of learning ML, too many other professionals from other domains easier, which would eventually attract people to switch to MLand analyst jobs which would fulfill the ever-growing demand for human resource in this sector.

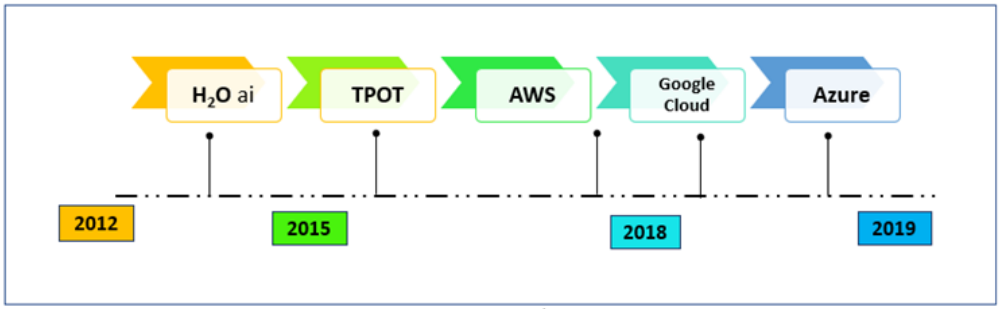
Challenges of AutoML

- **Computational power**: It will require more computational power to choose the correct model for the data, which will require the data to go through every model and fit it to find the accuracy, but if we try to do it manually we can trade-off this problem by eliminating many models which will certainly not work well with our data.

- **Difficulty with varying Datasets**: AutoML is usually generalized to different forms of the dataset at present. But every dataset can vary in the relevance of features, its structure, and datatypes present in features, AutoML can do a significantly satisfactory job with most of the datasets but it can not meet the accuracy and persistence which could be meet manually.

- **Black Box**: Although AutoML improves efficiency in producing results it can be difficult to track the flow of the algorithm that has delivered the respective output. Moreover, this also makes it difficult to choose the right model for a given problem, because it can be difficult to predict a result of the process selecting it, is a black box..

What AutoML Platforms are Available?



- **Google Cloud AutoML**: Google Cloud AutoML gained popularity due to its user-friendly interface and high performance. Build your own custom machine learning model in minutes.

- **Microsoft Azure AutoML**: Azure AutoML offers a transparent model selection process to its users who are not that familiar with coding. It is a cloud-based service for creating and managing machine learning solutions. Azure as a platform can be learned without knowing any programming at all.

- **H2O.ai**: It offers both an open-source package and a commercial AutoML service called Driverless AI. Since its inception, this platform has been widely adopted in industries, including financial services and retail. It enables organizations to rapidly build world-class AI models and applications.

- **TPOT**: TPOT (Tree-based Pipeline Optimization Tool) is a Python package that is free to use. https://github.com/EpistasisLab/tpot

Does AutoML eliminate data science jobs?

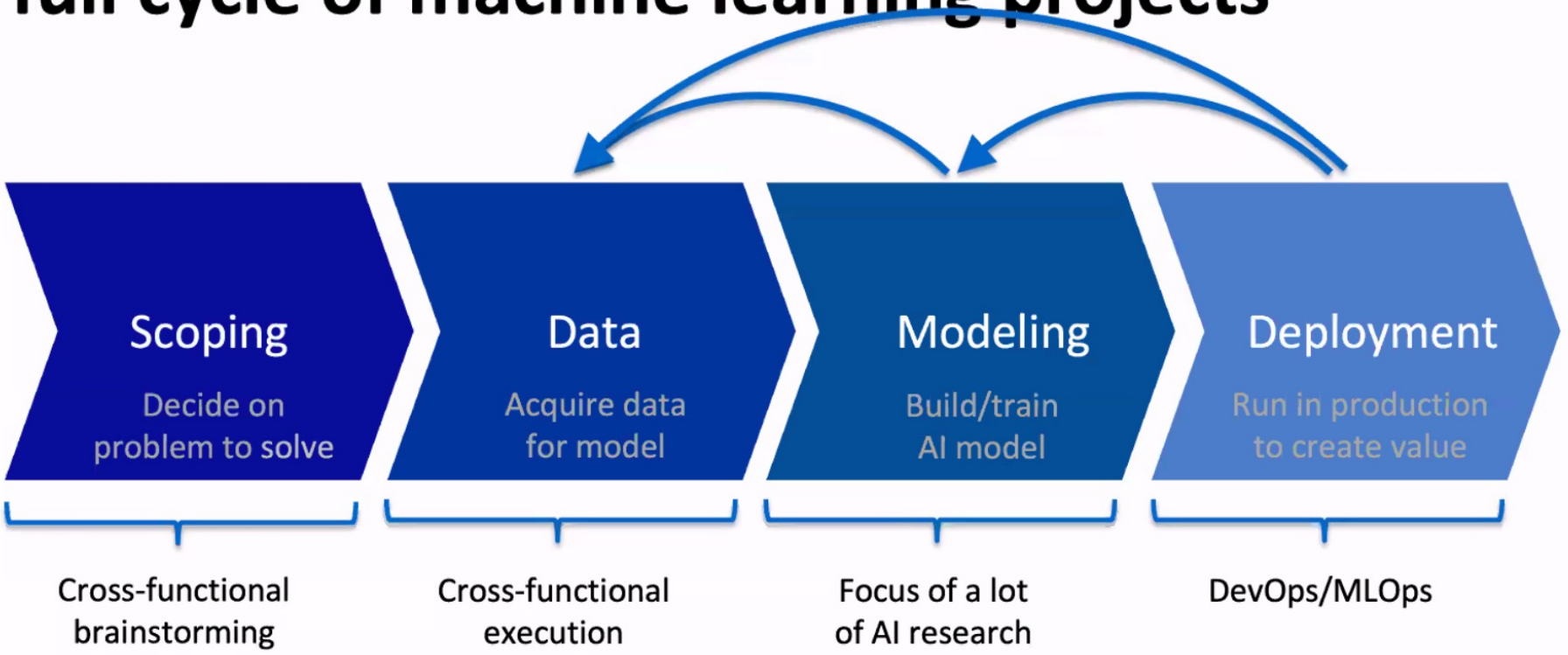
- While AutoMLs is quite satisfactory at choosing models most of the time, but they are still not capable of doing most of the work of a Data Scientist. We still need Data scientists/Analysts to apply their domain knowledge to generate more useful features(Feature Engineering)and information that impact the target outcome.

- AutoML will not replace most of the data science positions, instead, it can help professionals to fast the phase of their projects.

- Machines are not intelligent enough and the algorithms often don't generalize and understand the context of a problem.

- AutoML can help us find a suitable model for a given problem but it can't come up with a new approach which is often required for emerging real-life problems altogether.

The full cycle of machine learning projects



[Thanks to Dillon Laird, Daniel Bibireata]

Summary

- AI has found valuable applications in consumer internet. But much AI work in other industries still face a PoC to Production gap.
- Challenges: Small data, Generalizability & robustness, Change management.
- Full cycle of ML projects:

- Academia and industry should work to turn ML into a systematic engineering discipline.

Machine Learning Operations (MLOps): demo

`image_classifier_tf_gradio.ipynb`

Which should I use?

Gradio, Flask, PyWebIO, Streamlit and many more

	Simplicity	Maturity	Flexibility	Primary Use
Gradio	A	C	B	ML Model Demos
Streamlit	A	C	B	Dashboards
Dash	B	B	B	Dashboards
Flask	C	A	A	Web Interfaces