The reproduceability crises

Machine Learning Operations

Nicki Skafte Detlefsen,

Postdoc

DTU Compute

What is it?

MLOps

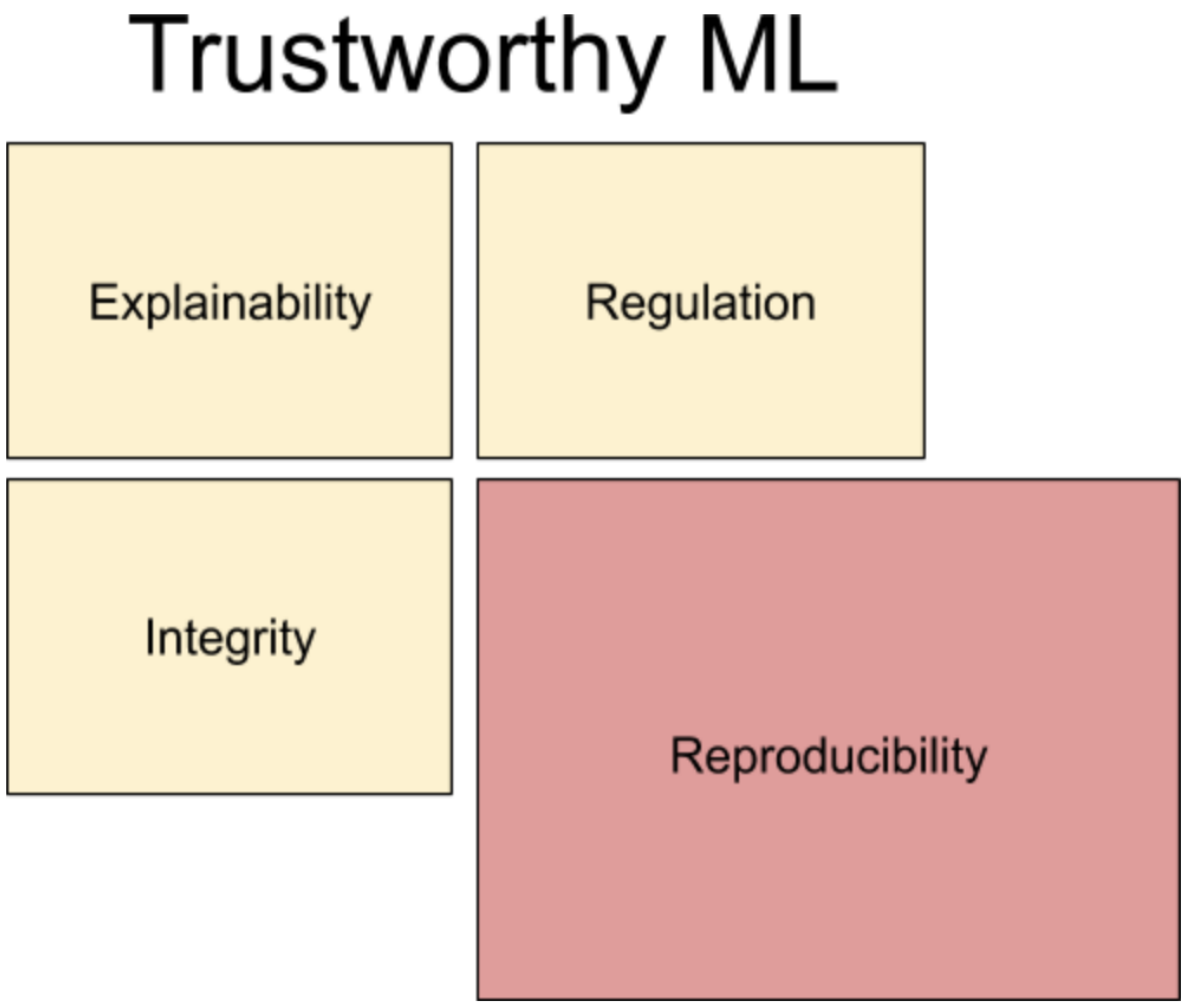
- Being able to reproduce other peoples experimental results is an essential part of the scientific method
- Well known problem thoughout most fields (physics, chemestry, biology and computer science)
- With the rise of deep learning, the problem has only been made worse due to competition

This is where it breaks

Why do we need reproduceability at all?

MLOps

Reproduceability is a key component in *Trustworthy ML*



From https://towardsdatascience.com/reproducible-machine-learning-cf1841606805

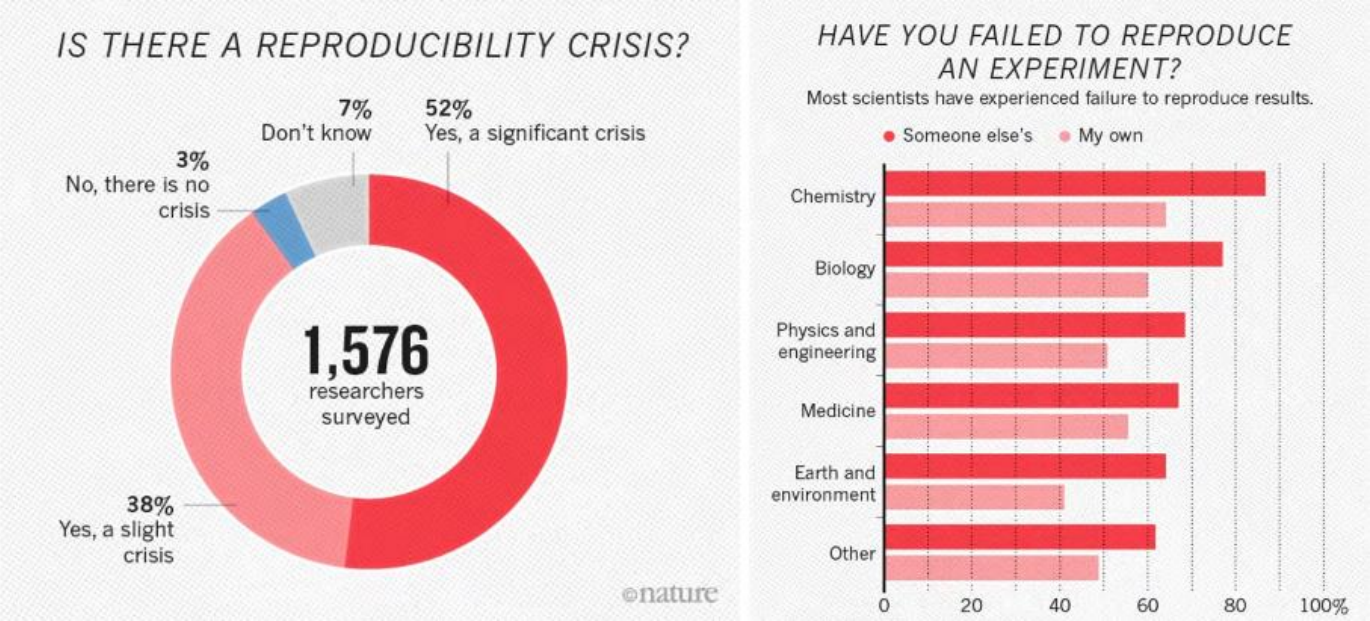
<u>Case:</u>

Imaging an AI agent used for dinostics. Without reproduceability two persons with the exact same symtomes could get different diagnosis

How bad is it?

MLOps

Wow its bad...



https://www.nature.com/news/1-500-scientists-lift-the-lid-on-reproducibility-1.19970

Machine learning around 26% (https://www.aaai.org/GuideBook2018/17248-73943-GB.pdf)

A closer look a machine learning

MLOps

Reimplement 255 and do hypothesis testing on what ”paper features” have an effect

A Step Toward Quantifying Independently Reproducible Machine Learning Research

Edward Raff
Booz Allen Hamilton
raff_edward@bah.com
University of Maryland, Baltimore County
raff.edward@umbc.edu

Abstract

What makes a paper independently reproducible? Debates on reproducibility center around intuition or assumptions but lack empirical results. Our field focuses on releasing code, which is important, but is not sufficient for determining reproducibility. We take the first step toward a quantifiable answer by manually attempting to implement 255 papers published from 1984 until 2017, recording features of each paper, and performing statistical analysis of the results. For each paper, we did not look at the authors code, if released, in order to prevent bias toward discrepancies between code and paper.

Table 1: Significance test of which paper properties impact reproducibility. Results significant at $\alpha \leq 0.05$ marked with“*”.

Feature	p-value
Year Published	0.964
Year First Attempted	0.674
Venue Type	0.631
Rigor vs Empirical*	1.55×10^{-9}
Has Appendix	0.330
Looks Intimidating	0.829
Readability*	9.68×10^{-25}
Algorithm Difficulty*	2.94×10^{-5}
Pseudo Code*	2.31×10^{-4}
Primary Topic*	7.039×10^{-4}
Exemplar Problem	0.720
Compute Specified	0.257
Hyperparameters Specified*	8.45×10^{-6}
Compute Needed*	8.75×10^{-5}
Authors Reply*	6.01×10^{-8}
Code Available	0.213
Pages	0.364
Publication Venue	0.342
Number of References	0.740
Number Equations*	0.004
Number Proofs	0.130
Number Tables*	0.010
Number Graphs/Plots	0.139
Number Other Figures	0.217
Conceptualization Figures	0.365
Number of Authors	0.497

What is the field trying to do about it?

MLOps

https://paperswithcode.com/

ML Reproducibility Challenge 2021 Spring

RC2021Spring

Online Jul 20 2021 https://paperswithcode.com/rc2020 reproducibility.challenge@gmail.com

Please see the venue website for more information.
Submission Start: Apr 01 2021 12:00AM UTC-0, End: TBD UTC-0

Add: ML Reproducibility Challenge 2021 Spring Submission

All Papers Claimed

Search by paper title and metadata Conference: All Conferences

ToTTo: A Controlled Table-To-Text Generation Dataset
Ankur P. Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, Dipanjan Das
24 Nov 2020 EMNLP 2020 Readers: Everyone 0 Replies

Tired of Topic Models? Clusters of Pretrained Word Embeddings Make for Fast and Good Topics too!
Suzanna Sia, Ayush Dalmia, Sabrina J. Mielke
24 Nov 2020 EMNLP 2020 Readers: Everyone 0 Replies

Towards Interpreting BERT for Reading Comprehension Based QA
Sahana Ramnath, Preksha Nema, Deep Sahni, Mitesh M. Khapra
24 Nov 2020 EMNLP 2020 Readers: Everyone 1 Reply

Dialogue Response Ranking Training with Large-Scale Human Feedback Data
Xiang Gao, Yizhe Zhang, Michel Galley, Chris Brockett, Bill Dolan
24 Nov 2020 EMNLP 2020 Readers: Everyone 0 Replies

Data Rejuvenation: Exploiting Inactive Training Examples for Neural Machine Translation
Wenxiang Jiao, Xing Wang, Shilin He, Irwin King, Michael R. Lyu, Zhaopeng Tu
24 Nov 2020 EMNLP 2020 Readers: Everyone 0 Replies

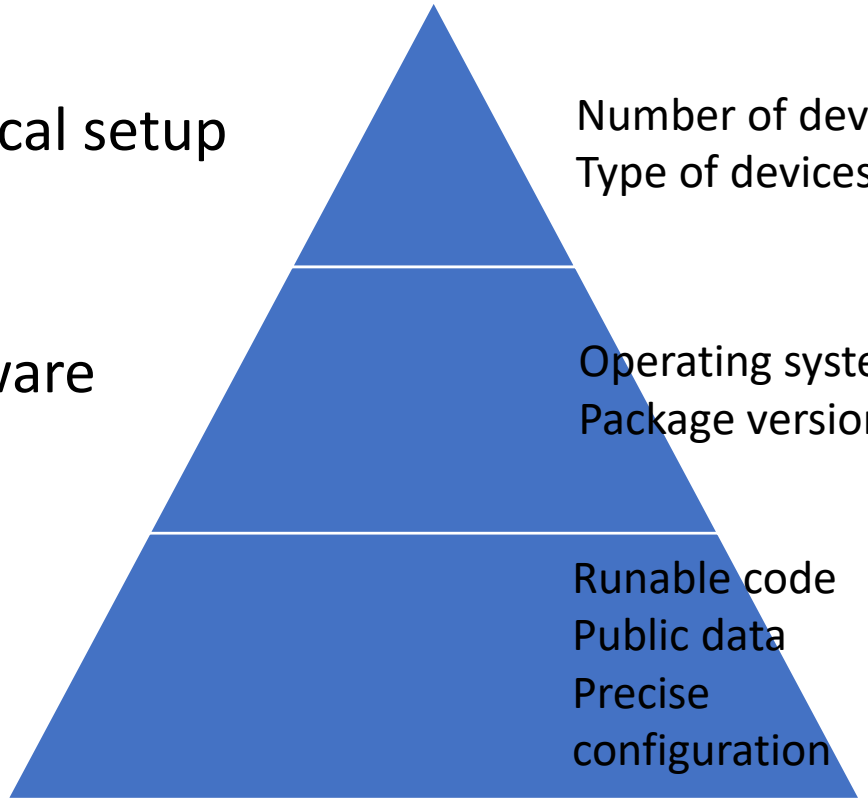
Neurips checklist:

3. If you ran experiments...
   - (a) Did you include the code, data, and instructions needed to **reproduce** the main experimental results (either in the supplemental material or as a URL)?
     - The instructions should contain the exact command and environment needed to run to reproduce the results.
     - Please see the NeurIPS code and data submission guidelines for more details.
     - Main experimental results include your new method and baselines. You should try to capture as many of the minor experiments in the paper as possible. If a subset of experiments are reproducible, you should state which ones are.
     - While we encourage release of code and data, we understand that this might not be possible, so "no because the code is proprietary" is an acceptable answer.
     - At submission time, to preserve anonymity, remember to release anonymized versions.
   - (b) Did you specify all the **training details** (e.g., data splits, hyperparameters, how they were chosen)?
     - The full details can be provided with the code, but the important details should be in the main paper.
   - (c) Did you report **error bars** (e.g., with respect to the random seed after running experiments multiple times)?
     - Answer "yes" if you report error bars, confidence intervals, or statistical significance tests for your main experiments.
   - (d) Did you include the amount of **compute** and the type of **resources** used (e.g., type of GPUs, internal cluster, or cloud provider)?
     - Ideally, you would provide the compute required for each of the individual experimental runs as well as the total compute.
     - Note that your full research project might have required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper). The total compute used may be harder to characterize, but if you can do that, that would be even better.
     - You are also encouraged to use a CO_2 emissions tracker and provide that information. See, for example, the experiment impact tracker (Henderson et al.), the ML CO_2 impact calculator (Lacoste et al.), and CodeCarbon.

What can you do about it?

MLOps

- Make sure to document everything about our experiments



Document this step as thoroughly as possible

Closer look

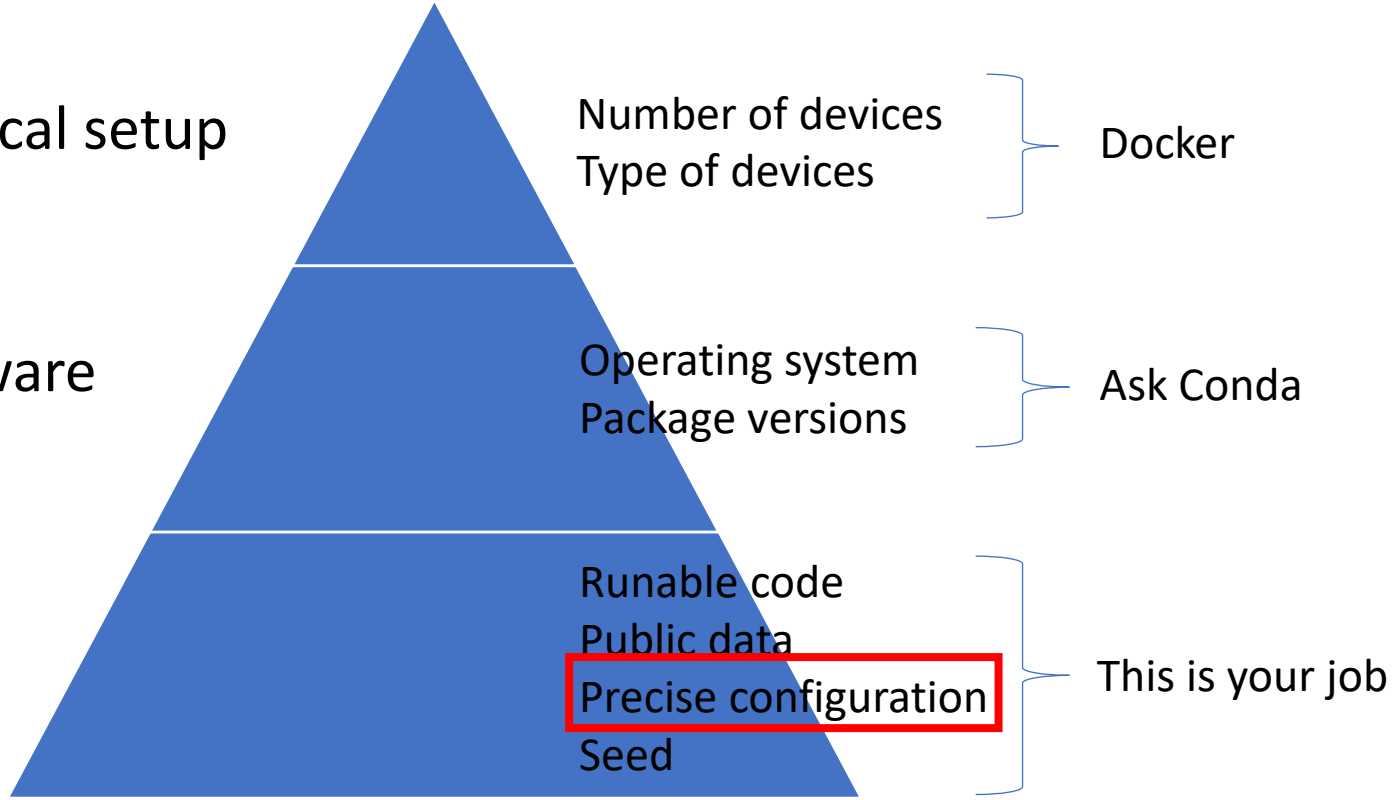


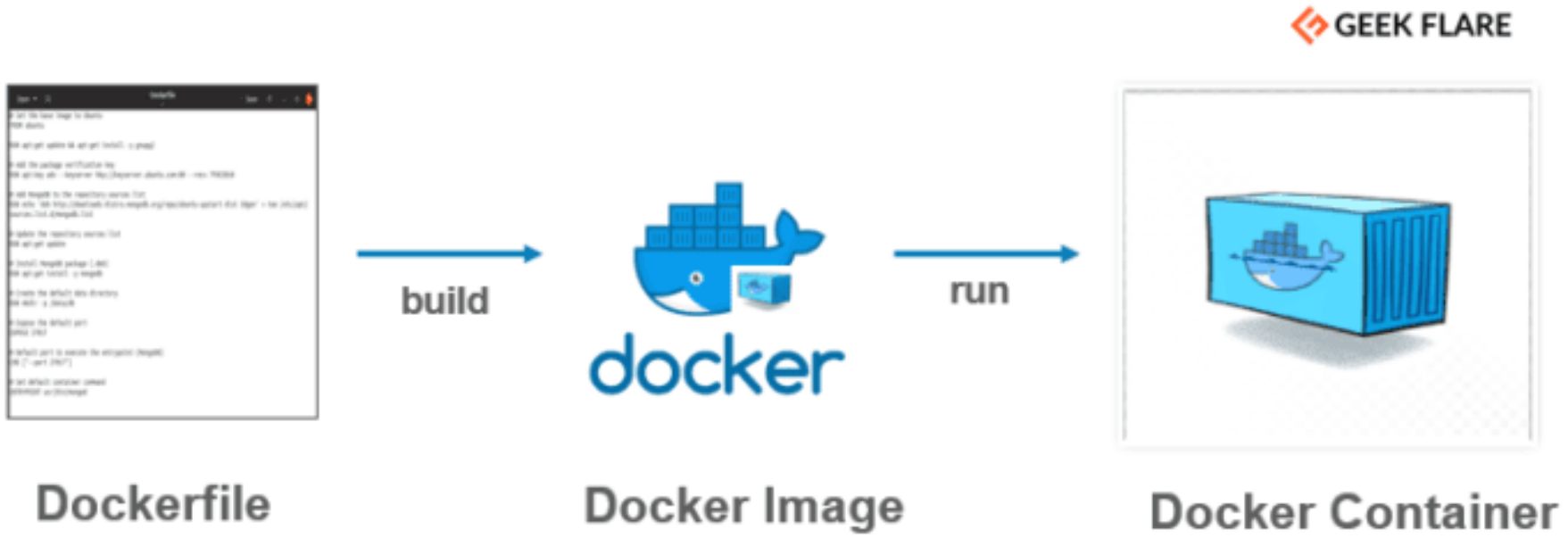
Table 1: Significance test of which paper properties impact reproducibility. Results significant at $\alpha \leq 0.05$ marked with"*".

Feature	p-value
Year Published	0.964
Year First Attempted	0.674
Venue Type	0.631
Rigor vs Empirical*	1.55×10^{-9}
Has Appendix	0.330
Looks Intimidating	0.829
Readability*	9.68×10^{-25}
Algorithm Difficulty*	2.94×10^{-5}
Pseudo Code*	2.31×10^{-4}
Primary Topic*	7.039×10^{-4}
Exemplar Problem	0.720
Compute Specified	0.257
Hyperparameters Specified*	8.45×10^{-6}
Compute Needed*	8.75×10^{-5}
Authors Reply*	6.01×10^{-8}
Code Available	0.213
Pages	0.364
Publication Venue	0.342
Number of References	0.740
Number Equations*	0.004
Number Proofs	0.130
Number Tables*	0.010
Number Graphs/Plots	0.139
Number Other Figures	0.217
Conceptualization Figures	0.365
Number of Authors	0.497

Docker

MLOps

- A way to create containerize applications = specialized VMs



How to construct the VM

File with the VM

Running VM

Configurations

MLOps

There is a lot of subjective choices that we do when running experiments, most noteable the hyperparameters.

Parameters in script	Argparser	Config files

`class hparams:`
`  lr = 0.1`
`  batch_size = 16`
`  num_layers = 5`
	`python my_script.py \`
	`  --lr 0.1 \`
	`  --batch_size 16 \`
	`  --num_layers 5`
		`experiment1.yaml`
		
		`lr: 0.001`
		`batch_size: 16`
		`num_layers: 5`
		
		`python my_script.py \`
		`  config=experiment1.yaml`

Not easy to configure Experiments may be lost if not carefull	Easy to configure Falls on user to save the configuration	Easy to configure Parameters are systematically saved with the experiment

MLOps

A framework for elegantly configuring complex applications.

https://github.com/facebookresearch/hydra

Correctly using configuration files makes sure that everything is documented

Example:

```
├── conf
│   ├── config.yaml
│   └── dataset
│       ├── cifar10.yaml
│       └── imagenet.yaml
└── my_app.py
```

```python
import hydra
from omegaconf import DictConfig

@hydra.main(config_path="config.yaml")
def my_app(cfg: DictConfig) -> None:
    print(cfg.pretty())

if __name__ == "__main__":
    my_app()
```

Other frameworks

- https://github.com/IDSIA/sacred

- https://mlflow.org/

Meme of the day

MLOps

Programmers

This code is unreadable and your dataset is flawed! No one will be able to reproduce your results!

It's not my fault the legacy environment is messed up! We still have 97.3% unit test coverage!

Scientists

This code is unreadable and your dataset is flawed. No one will be able to reproduce your results.

I know