DASYA
Data-Intensive Systems and Applications

RAD

INDEPENDENT RESEARCH FUND DENMARK

IT UNIVERSITY OF COPENHAGEN

dasya.itu.dk rad.itu.dk dff.dk itu.dk

Data Management and Visualization for Benchmarking Deep Learning Training Systems

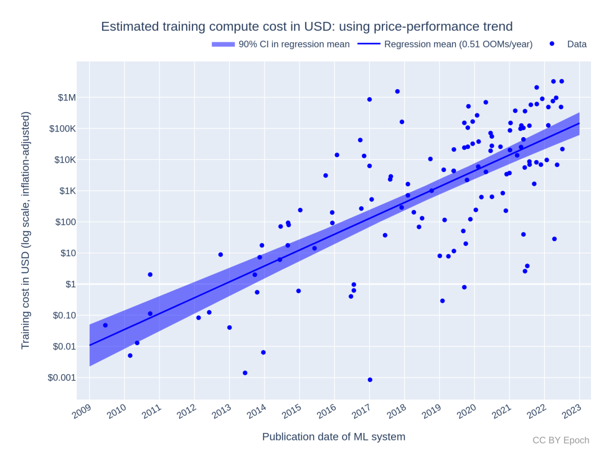
Ties Robroek, Aaron Duane, Ehsan Yousefzadeh-Asl-Miandoab, Pınar Tözün

(IT University of Copenhagen)

7th Workshop on Data Management for End-to-End Machine Learning (DEEM)
June 18th, 2023

Need for systematic benchmarking with hardware metrics

- Energy and resource consumption is increasingly relevant
- Resource costs relevant on both academic and industrial scale



https://epochai.org/blog/trends-in-the-dollar-training-cost-of-machine-learning-systems

Requirements

- 1. Wide configuration support including collocation
- 2. Track hardware metrics in addition to software metrics
- 3. Handle continuous streams of data
- 4. Support multiple visualization use-cases
- 5. Filter large amounts of inconsequential data
- 6. Minimal code impact

Requirements

- 1. Wide configuration support including collocation
- 2. Track hardware metrics in addition to software metrics
- 3. Handle continuous streams of data
- 4. Support multiple visualization use-cases
- 5. Filter large amounts of inconsequential data
- 6. Minimal code impact

Requirements

- 1. Wide configuration support including collocation
- 2. Track hardware metrics in addition to software metrics
- 3. Handle continuous streams of data
- 4. Support multiple visualization use-cases
- 5. Filter large amounts of inconsequential data
- 6. Minimal code impact

Requirements

- 1. Wide configuration support including collocation
- 2. Track hardware metrics in addition to software metrics
- 3. Handle continuous streams of data
- 4. Support multiple visualization use-cases
- 5. Filter large amounts of inconsequential data
- 6. Minimal code impact

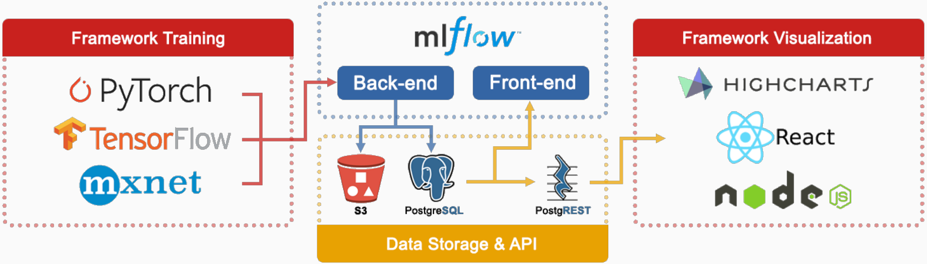
Requirements

- 1. Wide configuration support including collocation
- 2. Track hardware metrics in addition to software metrics
- 3. Handle continuous streams of data
- 4. Support multiple visualization use-cases
- 5. Filter large amounts of inconsequential data
- 6. Minimal code impact

Requirements

- 1. Wide configuration support including collocation
- 2. Track hardware metrics in addition to software metrics
- 3. Handle continuous streams of data
- 4. Support multiple visualization use-cases
- 5. Filter large amounts of inconsequential data
- 6. Minimal code impact

Resource-Aware Data systems Tracker (radT)



Framework Training

PyTorch

TensorFlow

mxnet

mlflow

Back-end

Front-end

S3

PostgreSQL

PostgREST

Data Storage & API

Framework Visualization

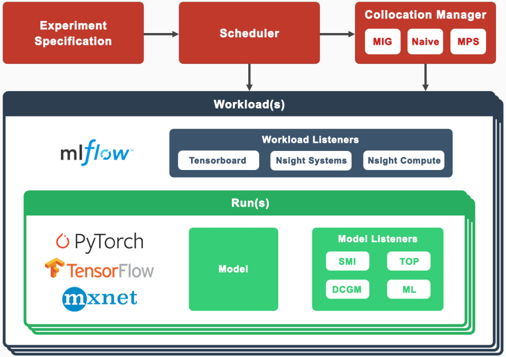
HIGHCHARTS

React

node js

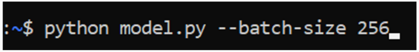
- Tracking and visualization of resources
- Extends MLFlow

1. Wide configuration support including collocation

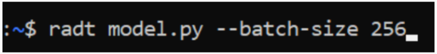


- Single model training (*run*) can be collocated together (*workload*)
- Workloads can be scheduled supporting large experiments

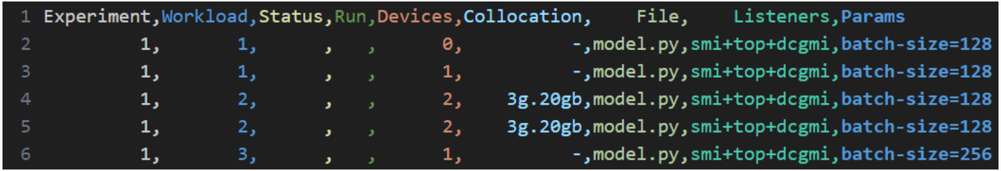
1. Wide configuration support including collocation



```
:~$ python model.py --batch-size 256
```

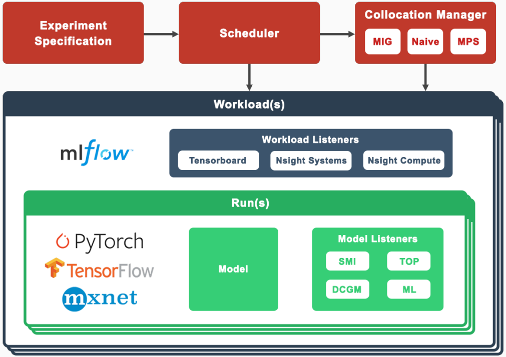


```
:~$ radt model.py --batch-size 256
```



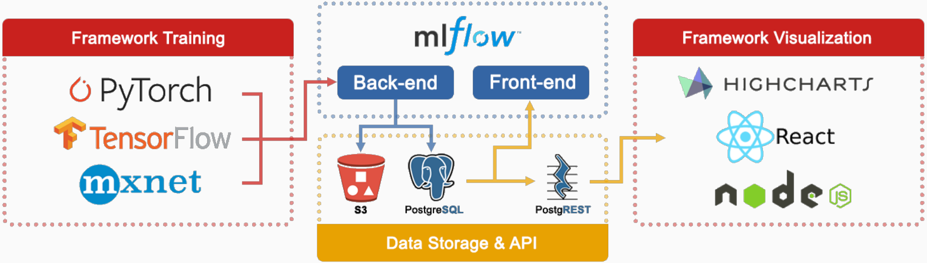
```
Experiment,Workload,Status,Run,Devices,Collocation,    File,    Listeners,Params
         1,       1,      ,   ,      0,          -,model.py,smi+top+dcgmi,batch-size=128
         1,       1,      ,   ,      1,          -,model.py,smi+top+dcgmi,batch-size=128
         1,       2,      ,   ,      2,    3g.20gb,model.py,smi+top+dcgmi,batch-size=128
         1,       2,      ,   ,      2,    3g.20gb,model.py,smi+top+dcgmi,batch-size=128
         1,       3,      ,   ,      1,          -,model.py,smi+top+dcgmi,batch-size=256
```

2. Track hardware and software metrics



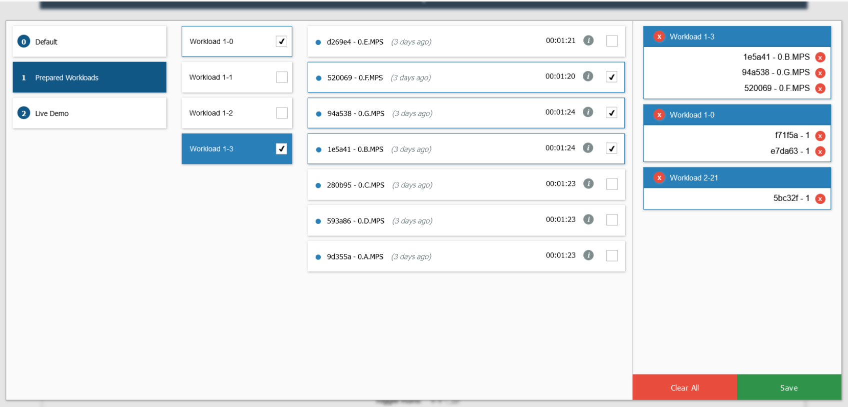
- Model Listeners: per run
- Workload Listeners: exclusive

3. Handle continuous streams of data



- Base configuration of MLFlow provides poor data scaling and interoperability
- MLFlow integration with dedicated PostgreSQL and S3 storage.

4. Support multiple visualization use-cases



0 Default

1 Prepared Workloads

2 Live Demo

☑ Workload 1-0

□ Workload 1-1

□ Workload 1-2

☑ Workload 1-3

● d269e4 - 0.E.MPS *(3 days ago)* 00:01:21 □

● 520069 - 0.F.MPS *(3 days ago)* 00:01:20 ☑

● 94a538 - 0.G.MPS *(3 days ago)* 00:01:24 ☑

● 1e5a41 - 0.B.MPS *(3 days ago)* 00:01:24 ☑

● 280b95 - 0.C.MPS *(3 days ago)* 00:01:23 □

● 593a86 - 0.D.MPS *(3 days ago)* 00:01:23 □

● 9d355a - 0.A.MPS *(3 days ago)* 00:01:23 □

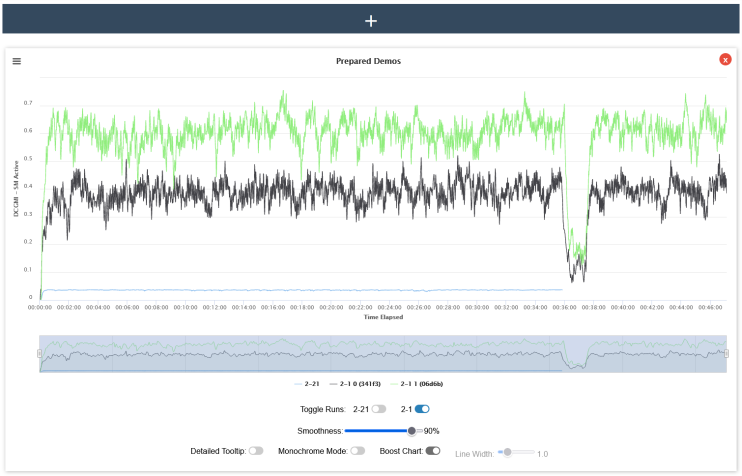
☒ Workload 1-3
- 1e5a41 - 0.B.MPS ☒
- 94a538 - 0.G.MPS ☒
- 520069 - 0.F.MPS ☒

☒ Workload 1-0
- f71f5a - 1 ☒
- e7da63 - 1 ☒

☒ Workload 2-21
- 5bc32f - 1 ☒

- Select data to compare in a hierarchical way

5. Filter large amounts of inconsequential data



Toggle Runs: 2-21 ○ 2-1 ●

Smoothness: 90%

Detailed Tooltip: ○ Monochrome Mode: ○ Boost Chart: ● Line Width: 1.0

- Create and share insights

6. Minimal code impact

PyTorch

TensorFlow

mxnet

⋮

```python
from radtrun import radT
...
with radT() as run:
  # Training loop
```

- No code required for hardware metric tracking
- Software tracking via MLFlow integration

- Minimal code for finer-grained control

Demo

Conclusion

- We need systematic benchmarking of both software and hardware resources

https://github.com/Resource-Aware-Data-systems-RAD/radt

- Track small and large experiments, including collocated ones
- Real-time, scalable data tracking and processing
- Efficient and effective data exploration

Thank you!

DASYA
Data-Intensive Systems and Applications

RAD

INDEPENDENT
RESEARCH FUND
DENMARK

IT UNIVERSITY OF COPENHAGEN

dasya.itu.dk

rad.itu.dk

dff.dk

itu.dk

Conclusion

- We need systematic benchmarking of both software and hardware resources

https://github.com/Resource-Aware-Data-systems-RAD/radt

- Track small and large experiments, including collocated ones
- Real-time, scalable data tracking and processing
- Efficient and effective data exploration

Thank you!

DASYA
Data-Intensive Systems and Applications

dasya.itu.dk

RAD

rad.itu.dk

INDEPENDENT
RESEARCH FUND
DENMARK

dff.dk

IT UNIVERSITY OF COPENHAGEN

itu.dk