mindpeak

Speeding up the deep learning development life cycle for cancer diagnostics

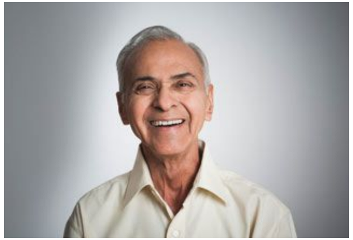
Marc Päpper

30.07.2021

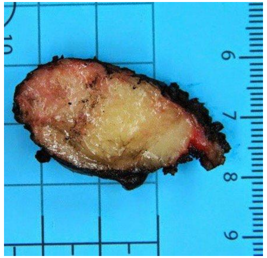
EUROPYTHON
2021 Jul 26 - Aug 1 Online

Our Mission

Increase cancer diagnostic accuracy and make it accessible to everyone who is in need

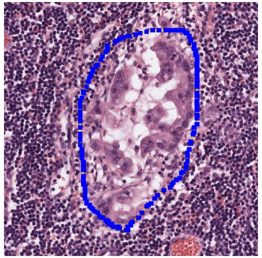



Cancer diagnostics today

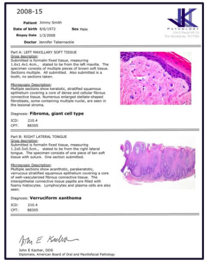


Biopsy

Examination



Diagnosis


2008-15

Treatment

Future cancer diagnosis **not** for everyone?

mindpeak

Demand for pathological diagnostics is increasing by 6% globally.

But numbers of qualified pathologists cannot meet this demand.

BBC Sign in | News | Sport | Weather | Shop | Earth | Travel

NEWS

Home | UK | World | Business | Politics | Tech | Science | Health | Family & Education

Health

Pathologists shortage 'delaying cancer diagnosis'

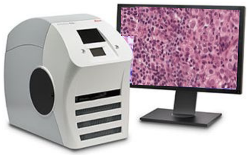
16 September 2018

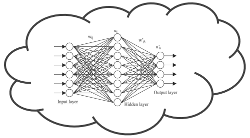
Share

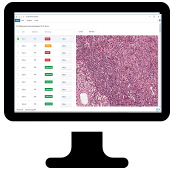
Patients are facing delays in diagnosis because of severe shortages among pathology staff, according to a report seen by the BBC.

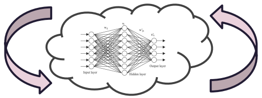
https://www.bbc.co.uk/news/health-45497014

Cancer diagnostics tomorrow







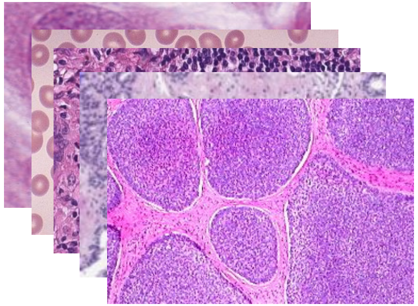


Scanner **AI pre-analysis** **Check on screen** **Improve model**

About MindPeak

mindpeak

- Automation tools for **visual diagnosis in pathology**
- Support cancer experts for reliable and reproducible diagnoses.



CHARITÉ UNIVERSITÄTSMEDIZIN BERLIN

HpH Institut für Hämatopathologie Hamburg

Fraunhofer MEVIS

NVIDIA INCEPTION PROGRAM

IFB innovations starter

DERMATOLOGIKUM HAMBURG

MVZ Labor Dr. Fenner und Kollegen

Kepler Universitäts Klinikum

patho nomics

Bundesministerium für Bildung und Forschung

KMU-innovativ

Vorfahrt für Spitzenforschung im Mittelstand

PHILIPS

3DHISTECH

DIGITAL PATHOLOGY TEAM

GESTALT DIAGNOSTICS

sysmex

INSTITUT FÜR PATHOLOGIE UNIVERSITÄT BERN mindpeak

Our Team and Advisors

Prof. Markus Tiemann

Pathologist and Managing Director at Institute of Haematopathology Hamburg

Prof. Axel Wellmann

Pathologist and Managing Director at Institute of Pathology Celle

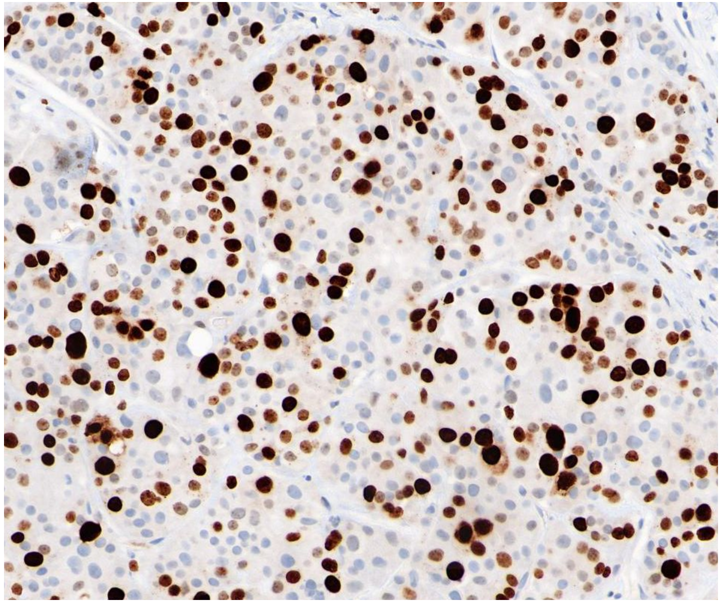
Dr. Thomas Fenner

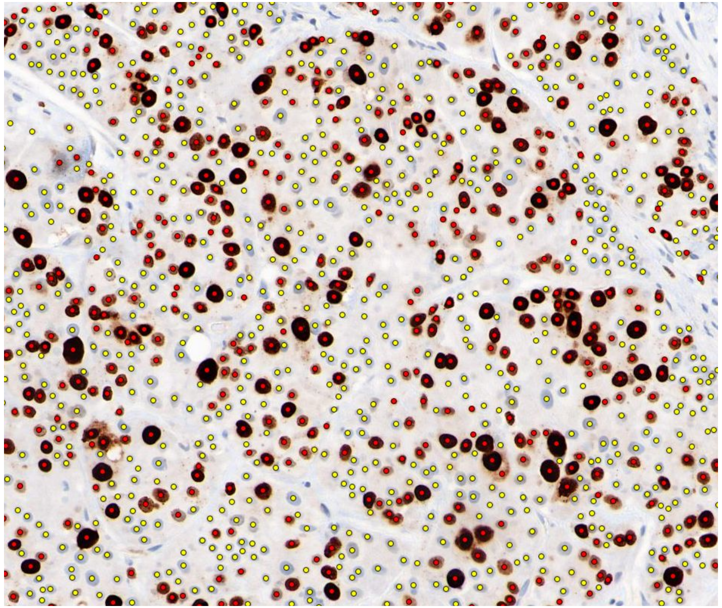
MD for Humane Medicine and Head of Laboratory Fenner

Geoff Baum

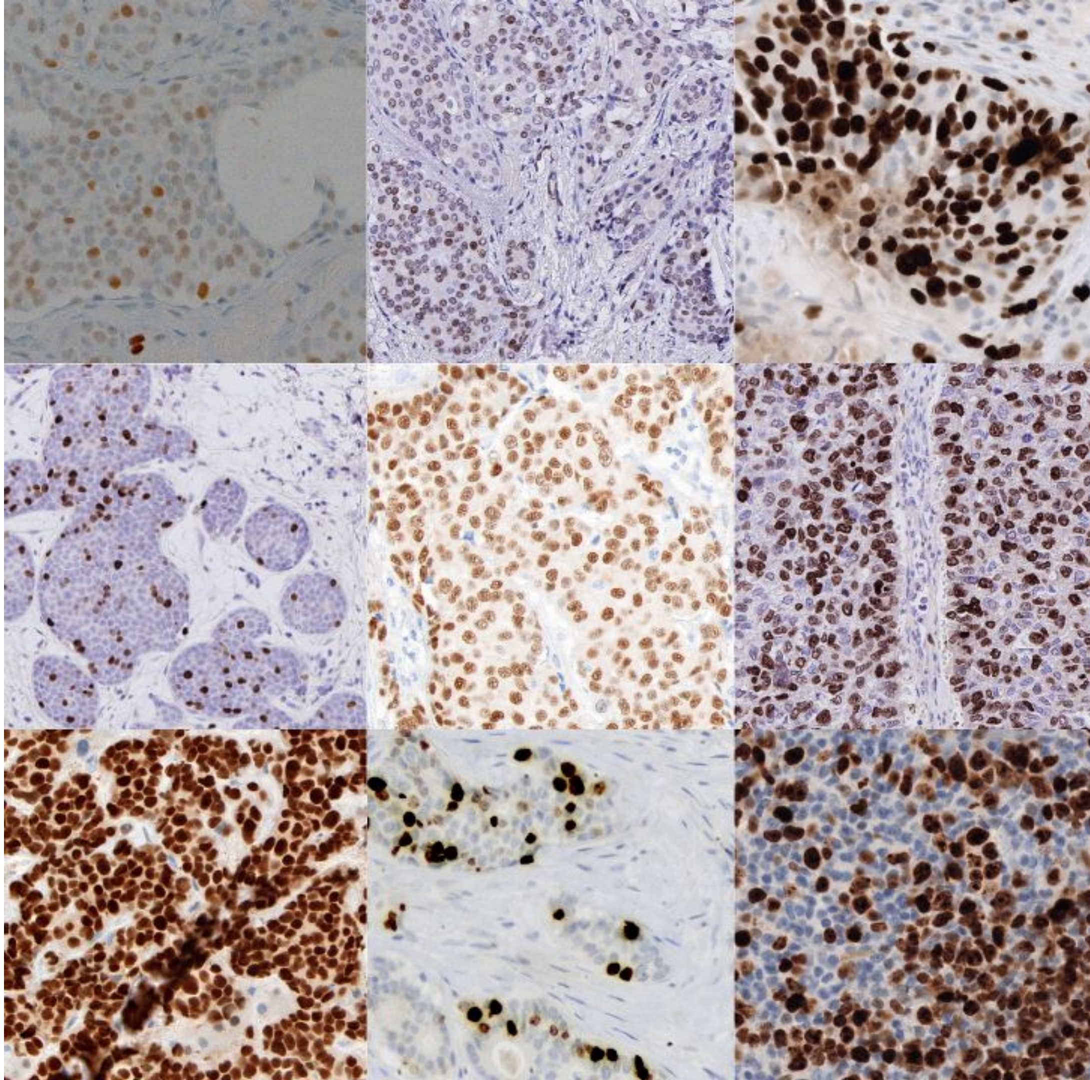
VP of Product at Adobe, co-founded Garage Ventures with Guy Kawasaki, U Stanford

Example: cancer cell detection

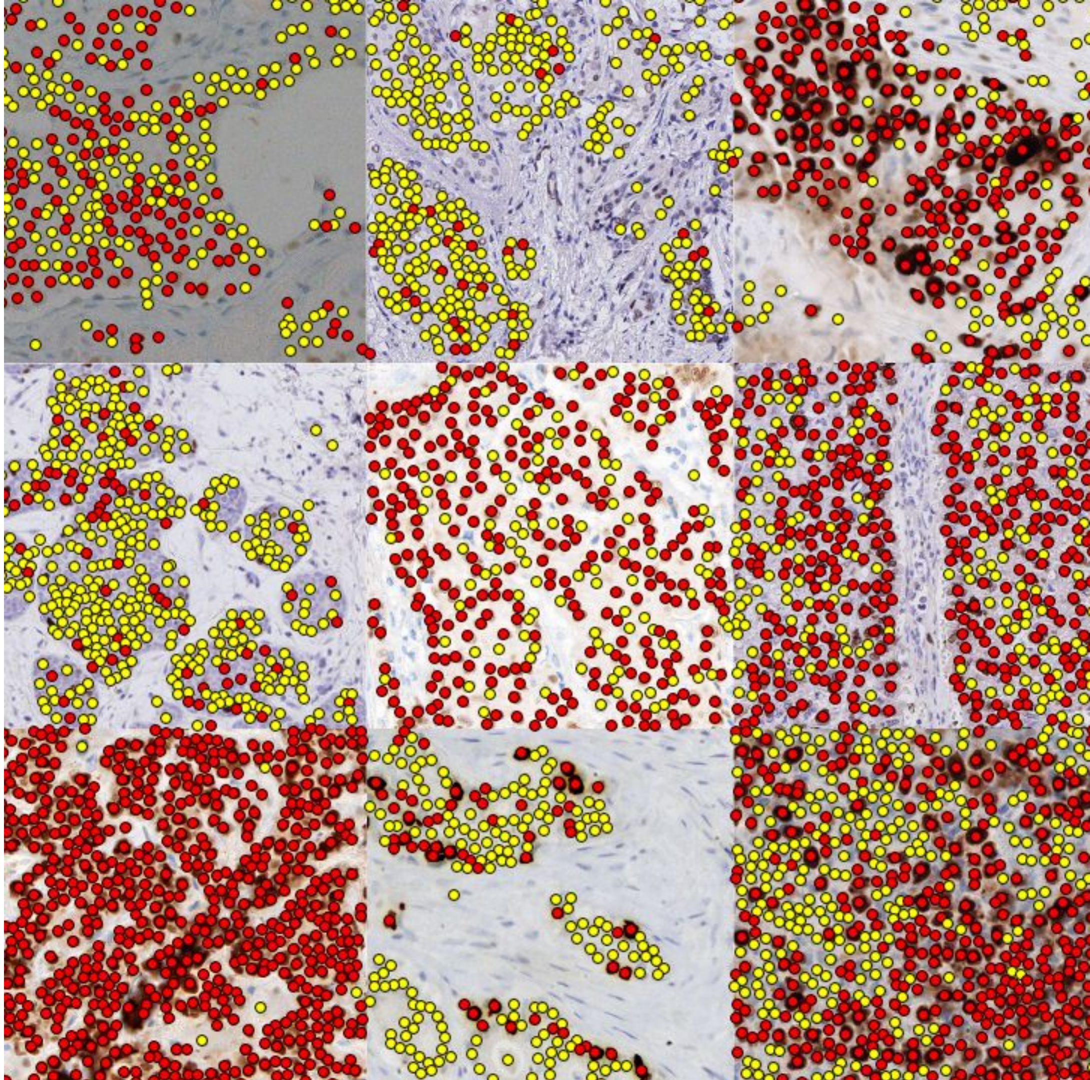




Robustness against many variations in the lab

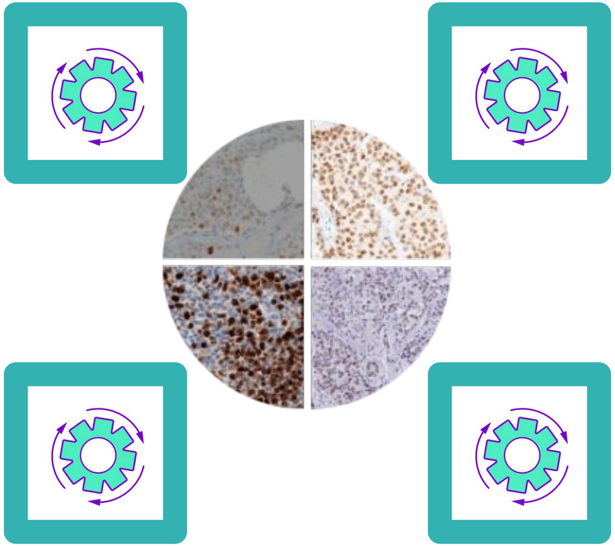


Robustness against many variations in the lab

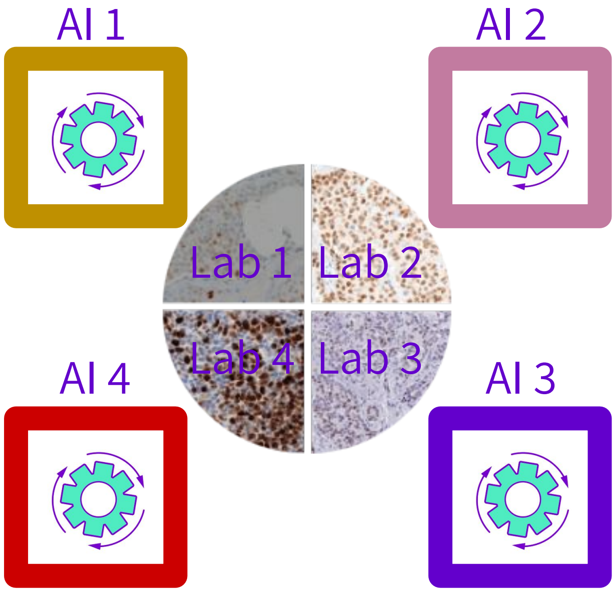


Simplicity

One AI for all labs

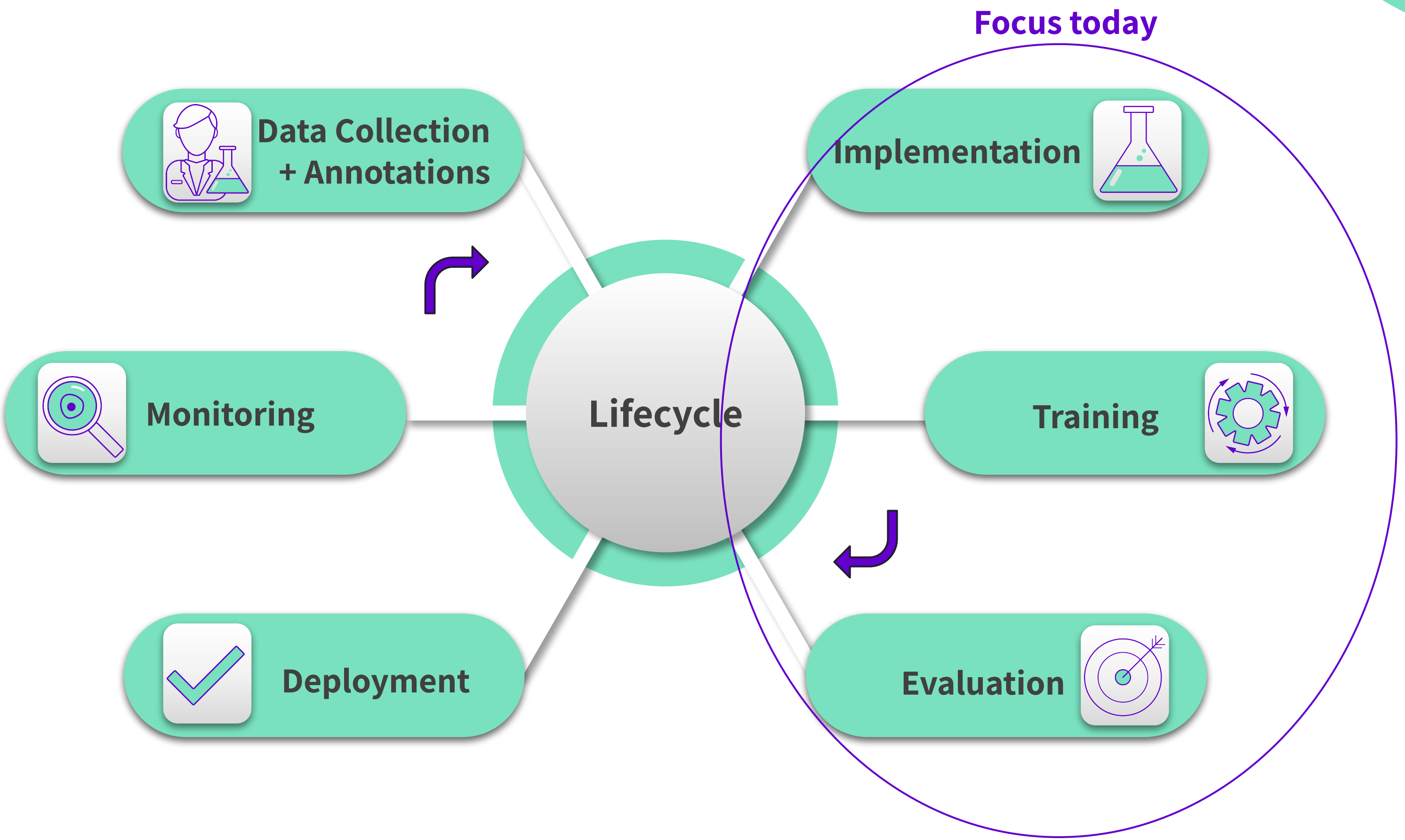


mindpeak

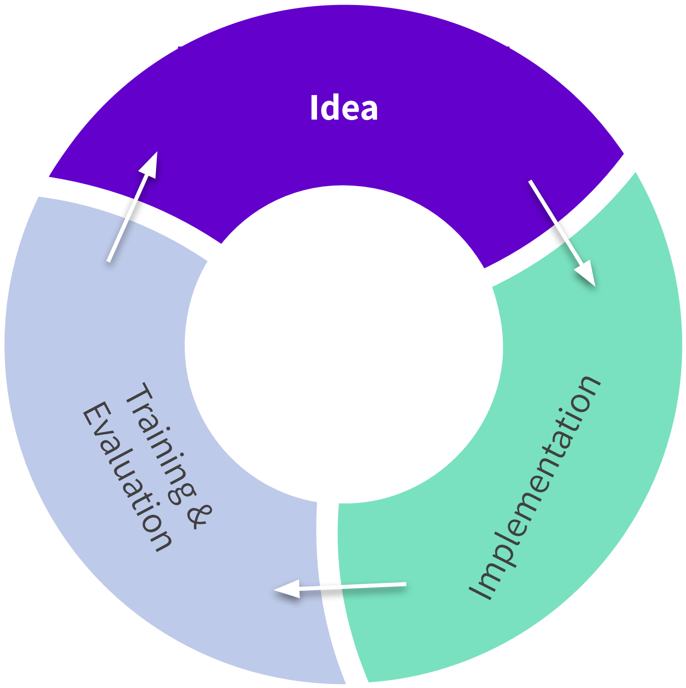


Other Vendors

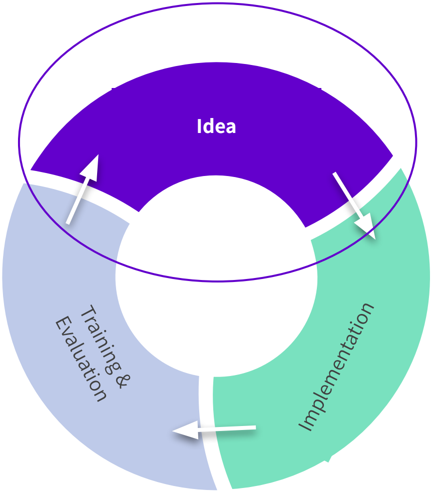
Training a deep learning model



Goal: Test new ideas quickly

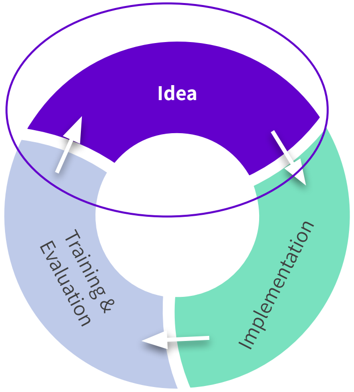


Stage: Idea



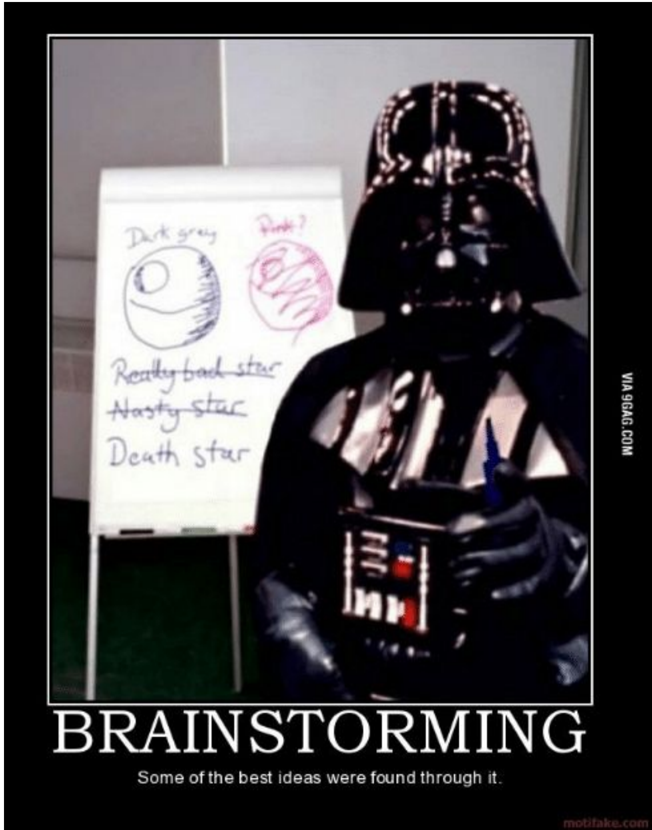
Overview: Idea stage

- Idea generation
  - Without data
  - Data-driven
- Efficient Annotations
- Metrics - define your target goals



Idea Generation – without data

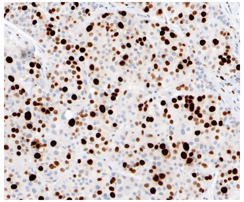
- Use brainstorming techniques
  - But avoid groupthink
- Organize and group cards together
- Have a short, timeboxed discussion
- Then use limited voting sticky dots
- Evaluate e.g. with the Business Model Canvas



Data -driven idea generation

- Use a subset of your validation set for decisions:
  - Categorize the errors
  - Focus on high error categories

Example: let’s improve Stroma -> Immune misclassification!



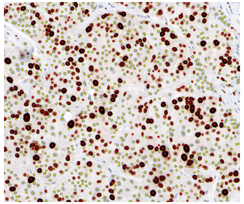
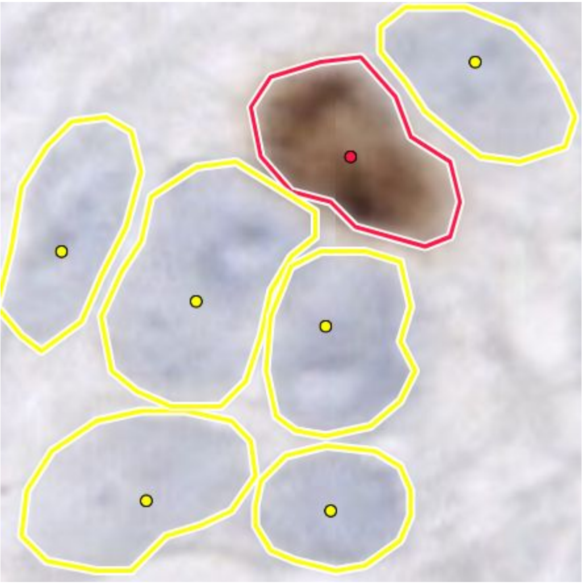
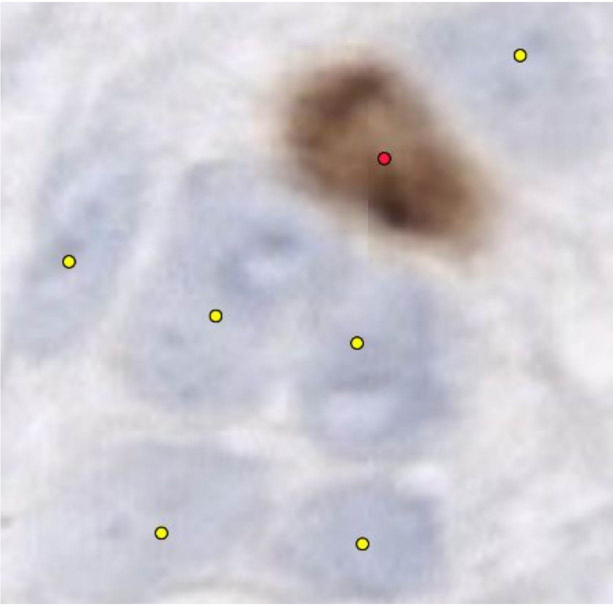


Image number	Stroma -> Immune	Immune -> Stroma	Scan artefacts
1	x		
2		x	x
3	x		x
...			
99		x	x
100		x	x
Total Counts	4	7	35

Efficient Annotations



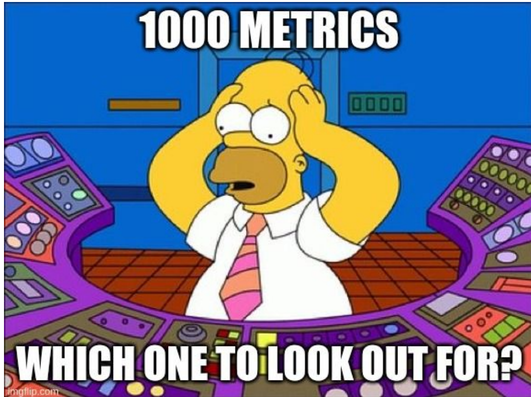
vs



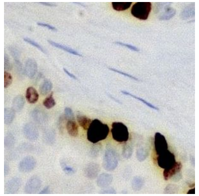
Metrics - define your target goals

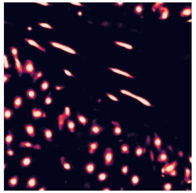
mindpeak

- Single metric for comparability
  - Aggregates like F1
  - Weighted average of aspects
  - Min operator to enforce equal importance
- Single metric validates your ideas
- Sub metrics
  - figure out where to improve
  - guide to generate next ideas

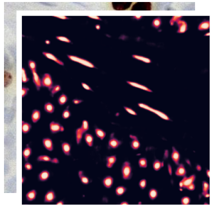


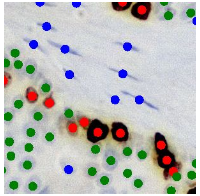
Metrics - Mindpeak example





Cell Detection: Precision + Recall -> F1 Score





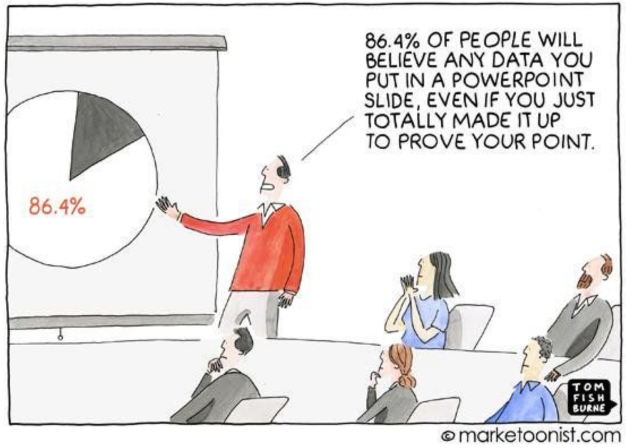
Cell Classification: Precision + Recall -> F1 Score for each class
Weighted combination of single F1 scores -> Overall F1 Score

Target Metric: Combination of Detection F1 Score + Overall Classification F1 Score

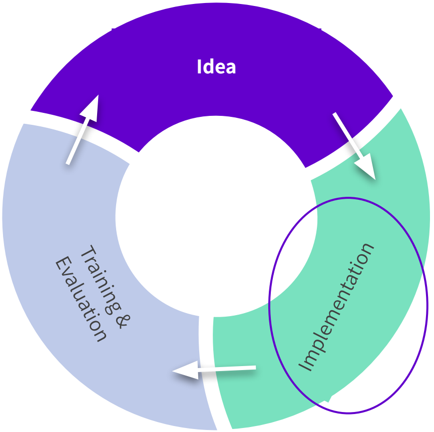
Idea stage - summary

mindpeak

- Think about efficient annotations
- Define single metric to guide experimentation
- Use your data to drive ideas
- Track errors your current model makes and categorize them
- Identify ideas matching high error categories



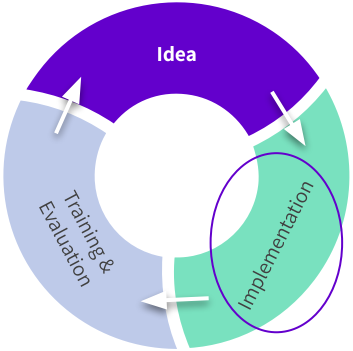
Stage: Implementation



Overview: Implementation stage

- Code quality (Linter, Typing, Refactorings)
- Automation: CI pipelines
- Profiling:
    - Interplay CPU vs GPU usage -> Nvidia Profiler Nsight
- Reproducibility

```
def greeting(name: str) -> str:
    return 'Hello ' + name
```



Code quality - comments as code

mindpeak

```python
def loss(predictions, targets, background_index):
    # filter out background targets
    mask = targets != background_index
    predictions = predictions[mask]
    targets = targets[mask]

    loss = F.l1_loss(predictions, targets)

    return loss
```

```python
def _filter_background_targets(predictions, targets, background_index):
    mask = targets != background_index

    return predictions[mask], targets[mask]

def loss(predictions, targets, background_index):
    predictions, targets = _filter_background_targets(predictions, targets, background_index)

    loss = F.l1_loss(predictions, targets)

    return loss
```

https://www.paepper.com/blog/tags/refactoring/

Code quality - use einops library

```
def _flatten_by_channel(prediction):
    batch_size, channels, height, width = prediction.shape
    permuted = prediction.permute((0, 2, 3, 1))
    final = permuted.contiguous().view((batch_size * height * width, channels))
    return final
```

```
def _flatten_by_channel(prediction):
    return rearrange(prediction, 'batch channel height width -> (batch height width) channel')
```

https://www.paepper.com/blog/tags/refactoring/

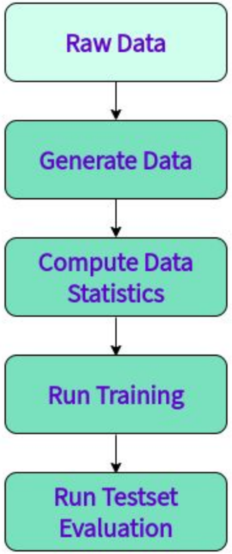
CI pipelines

mindpeak

- Take advantage of automation
  - Automatic checks for code format / PEP 8 etc
  - Automatic unit tests
    - Especially metrics / losses / data loading / data transformation
  - Automatic docker image build
  - Automatic deployment of demo model / visualization prototype
- Bugs caught early are the best -> save time

On reproducibility

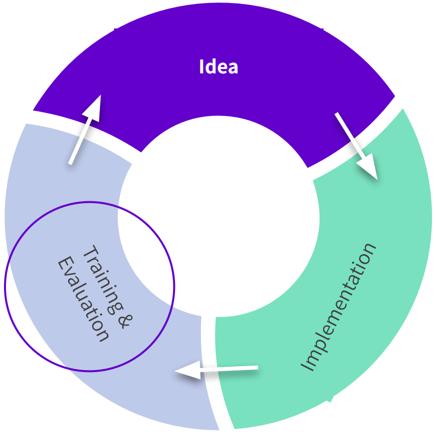
- Track your data with your code
  - Which experiment ran with what data?
  - Comparability between experiments
- We use data version control (dvc)
  - Tracking the data
  - Generating pipelines for training and evaluation
- Track your metrics and tensorboard logs
- Have an easy automated pipeline for comparing with the previous model



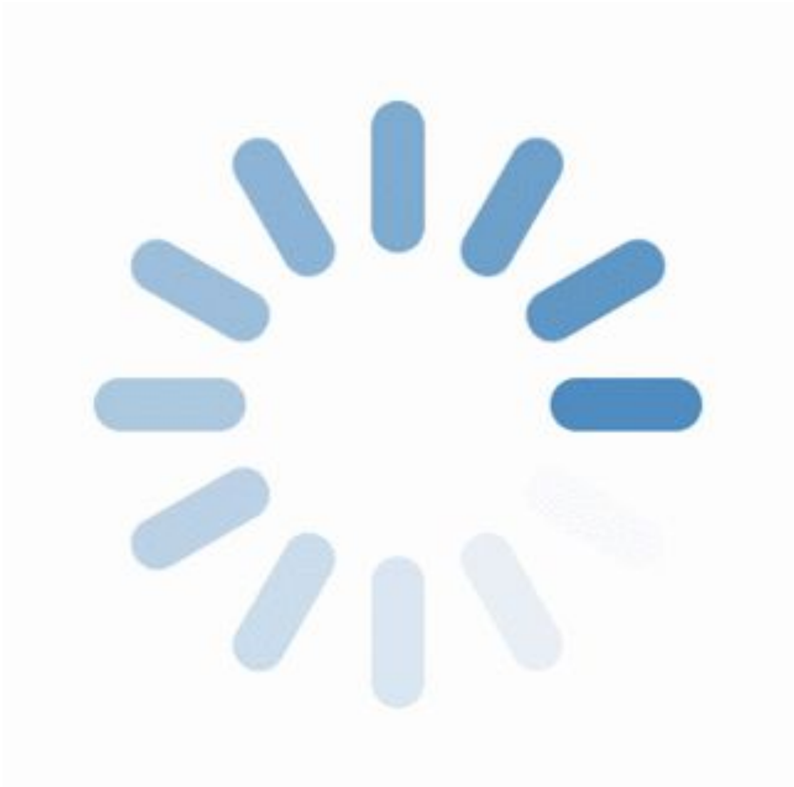
Implementation stage - summary

- Push for high code quality
  - Touch code -> refactor
  - Linting + typing + tests
- Take advantage of automation: CI
- Use a profiler to avoid easy bottlenecks
- Achieve reproducibility and track your data

Stage: Training + Evaluation



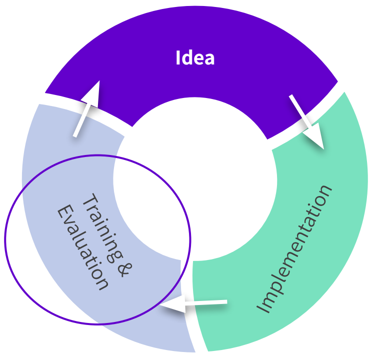
Training & Evaluation



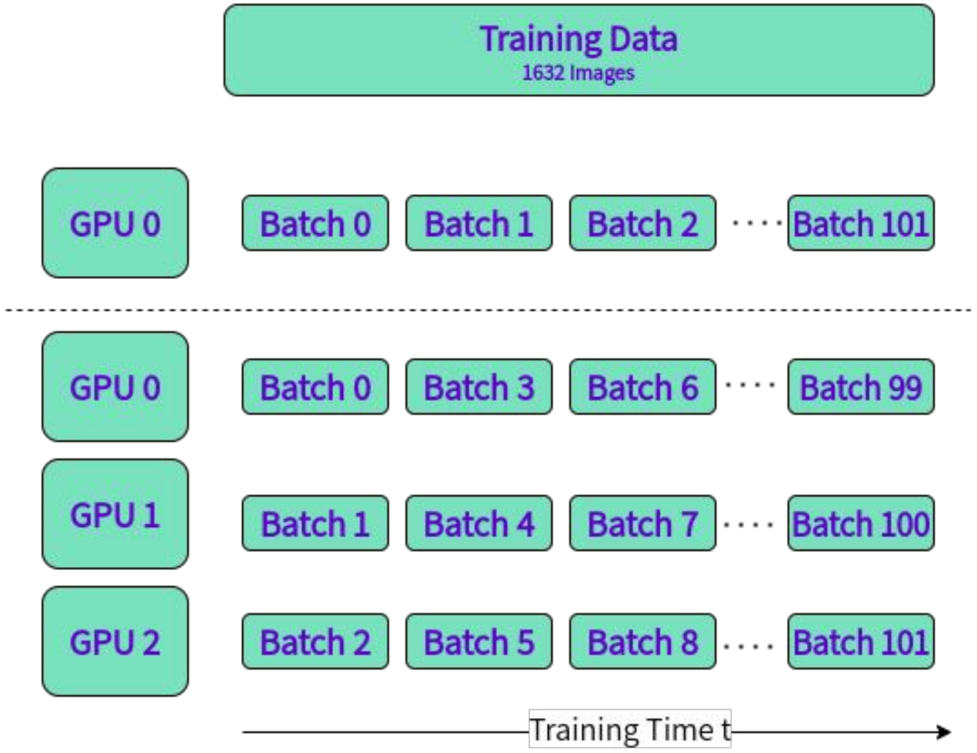
Overview: Training & Evaluation stage

mindpeak

- Multi GPU Training for speedup
- Dataset reduction techniques
- Single metric as guidance

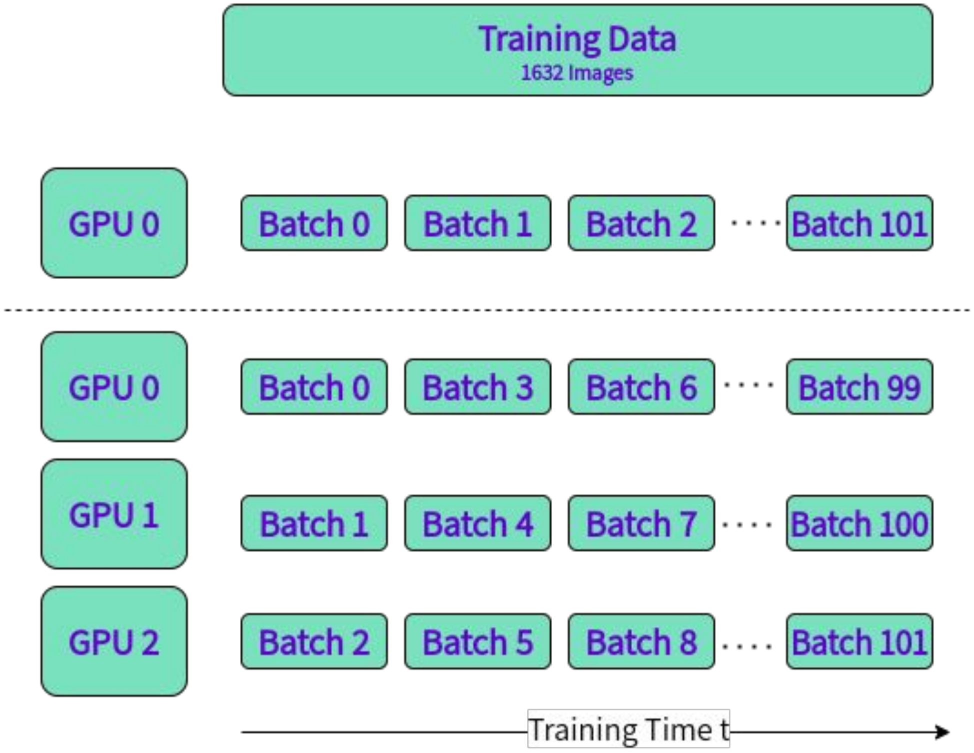


PyTorch Data Parallelization

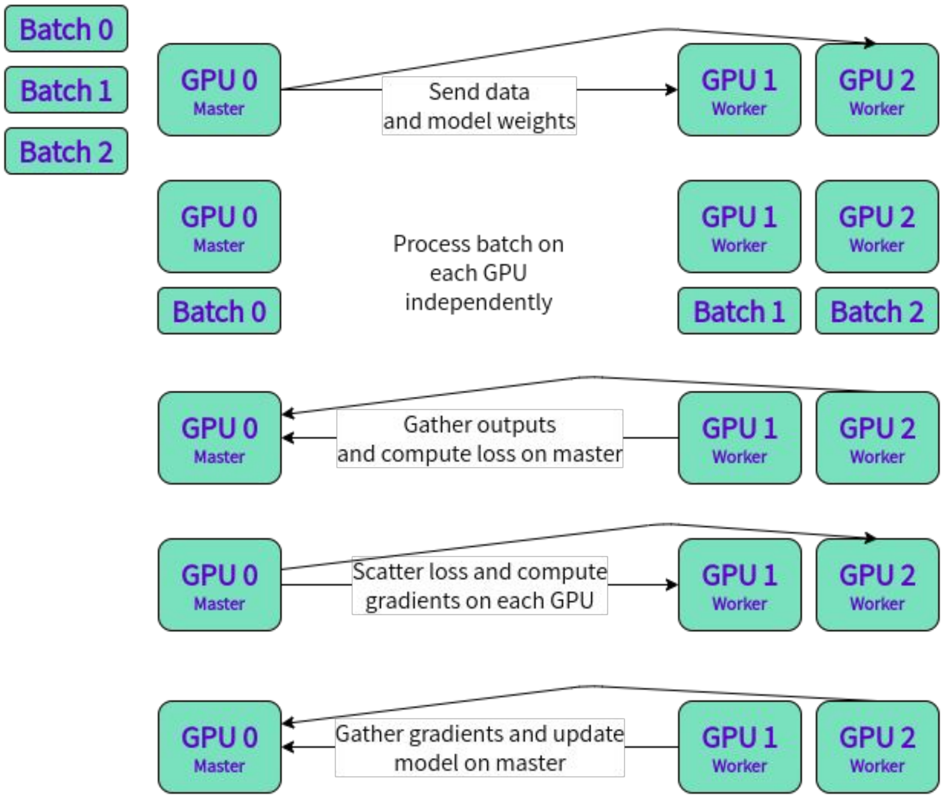


Speedup 3x?!

PyTorch Data Parallelization



PyTorch Data Parallelization



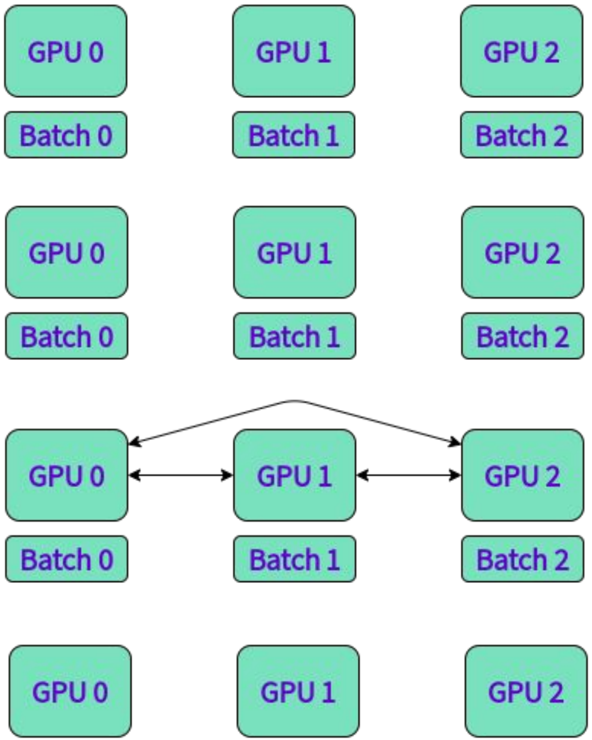
Pytorch Distributed Data Parallelization

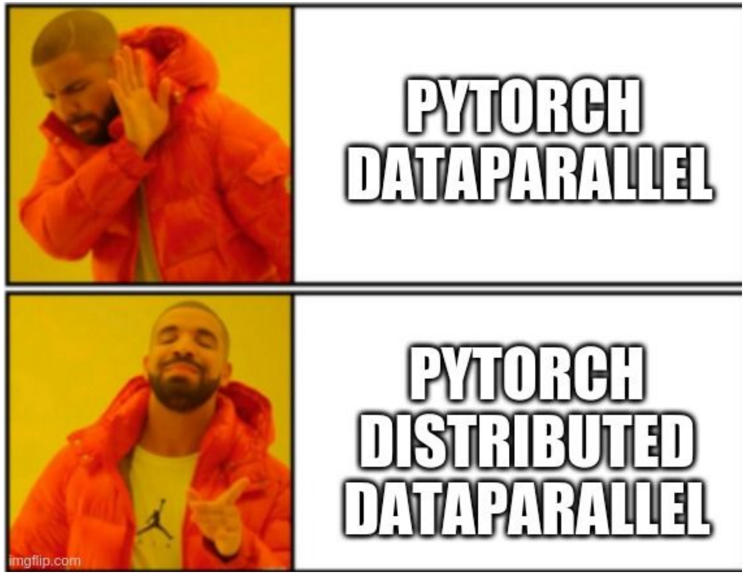
Load and process batch on each GPU separately

Compute loss and gradient separately on each GPU

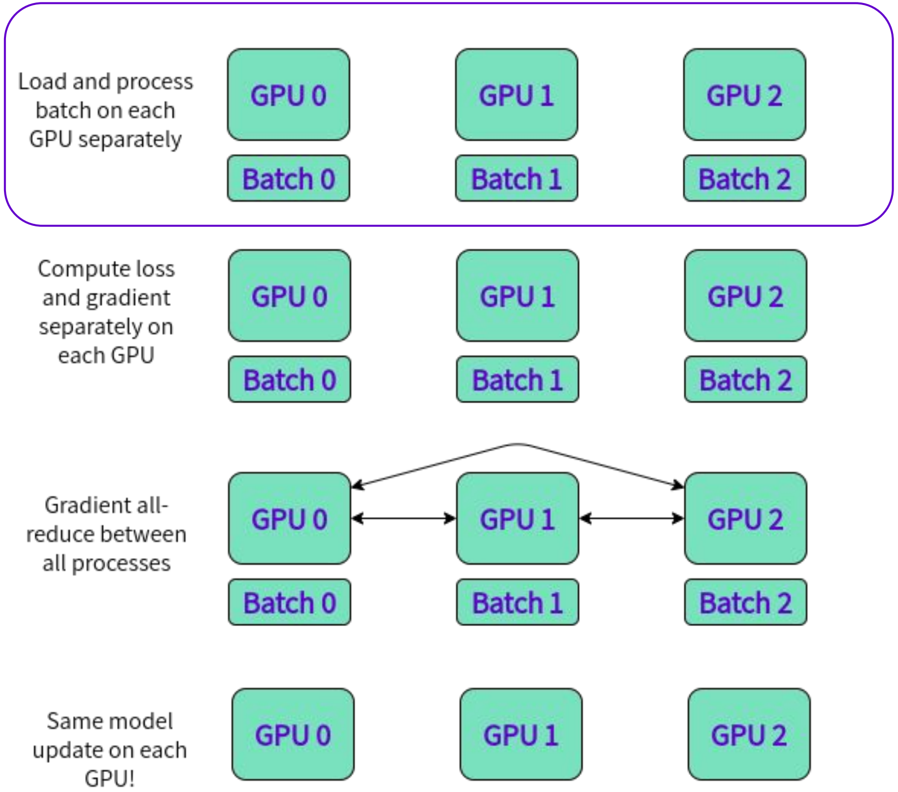
Gradient all-reduce between all processes

Same model update on each GPU!





Pytorch Distributed Data Parallelization



You need to use a distributed sampler which takes care of loading different data on each GPU

Reference:
torch.utils.data.distributed.DistributedSampler

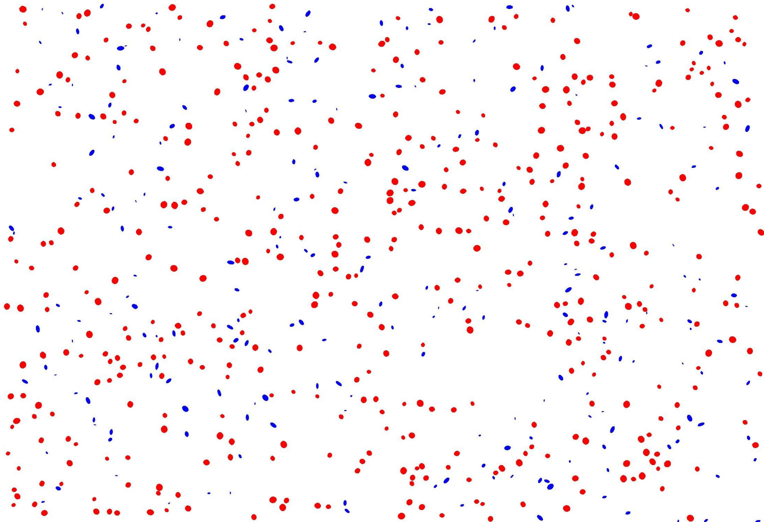
\+ Learning Rate Warmup,
Learning Rate Scaling

Goyal et al. 2018:
https://arxiv.org/pdf/1706.02677.pdf

Dataset reduction techniques

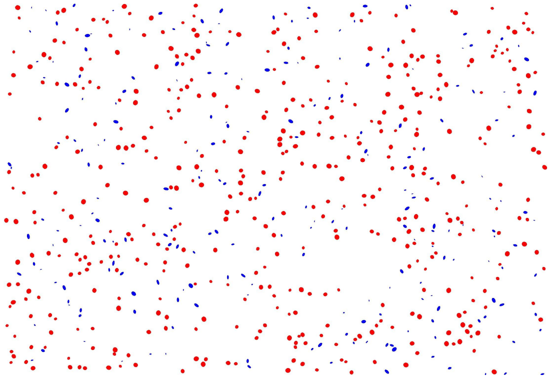
mindpeak

- Shrink your dataset to a representative smaller set
  - Not easy to define representative
  - Risk of losing out on information
  - Easy to do
- Create a toy dataset
  - Much more work
  - Highly customizable via parameters
  - Not sure how it translates to real data
  - Very good for idea prototyping
    - e.g. label noise experiments



Training + evaluation stage - summary

- Take advantage of multi GPU training
- Dataset reduction techniques for faster training
  - Small representative subset
  - Toy dataset
- Single metric to select if your idea is a winner



Disappointment

mindpeak

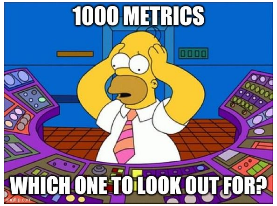
Disappointment

mindpeak

- ML: lots of iteration
- ~~Ideas~~
- Learn from it:
  - Why?
  - Your data / problem?
  - New ideas?
- Write a diary / Confluence page
- Quality + Speed -> Next idea

Summary

Image number	Stroma -> Immune	Immune -> Stroma	Scan artefacts
1	x		
2		x	x
3	x		x
...			
99		x	x
100		x	x
Total Counts	4	7	35



- Use data-driven idea generation
- Have a single evaluation metric to target
- Automate all the things (especially tests / CI pipelines)
- Track your data with your code by using dvc
- Strive for high code quality
    - Comments as code
    - Use einops for dimensional readability
- Use Distributed Data Parallel to train on multiple GPUs
- Take advantage of learning opportunities from experiments
- Iterate quickly + have fun

mindpeak

Looking forward to your questions!

Get in touch

@ marc@mindpeak.ai

@mpaepper

Marc Päpper