Reading data with Tensorflow

Author of the talk

- Agrin Hilmkil
- Engineering Physics/Math
  @Chalmers
- Masters Thesis @Peltarion
  Feb 2017
- Joined full-time Okt 2017
- Previously interned at
  Microsoft, Burt, VMware,
  Ericsson

Todays goal

- Show you how easy it is to build fast pipelines with tf Datasets and allow you to decide if you should learn more

- Show you where to find what you will need to go beyond the basics

Outline

- **A brief history of loading data**
  - feed_dict, Queues and Datasets
- **How to build a pipeline**
  - Case: Thorax region disease detection on NIH Chest Xrays (78GB)
- **Performance improvements**
  - Caching, Prefetching, Sharding
- **Pitfalls**
- **Grand Finale**: Live training ResNet-152 on 4 Kepler GPUs

A brief history of loading data in Tensorflow

- Feed data into placeholders

- Queues and QueueRunners (Python threads)

- Datasets API (Current recommended method)
  [https://www.tensorflow.org/versions/r1.4/api_guides/python/reading_data]

Feed data

- Define a placeholder
- Build graph from placeholder
- Feed data into placeholder each iteration
- Flexible and can use Python code

```
with tf.Session():
  input = tf.placeholder(tf.float32)
  classifier = ...
  print(classifier.eval(feed_dict={input: my_python_preprocessing_fn()}))
```

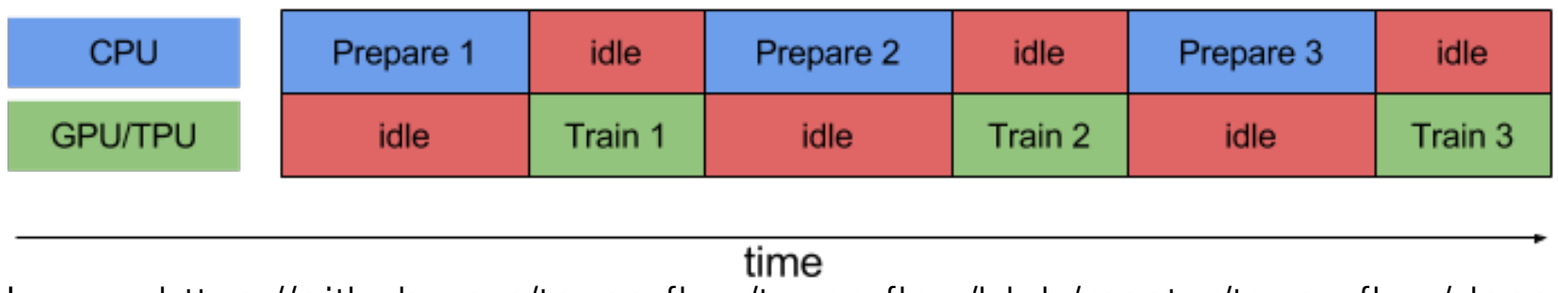


Image: https://github.com/tensorflow/tensorflow/blob/master/tensorflow/docs_src/performance/datasets_performance.md

Queues and QueueRunners

- Define Queues and use dequeue ops as inputs to graph
- Create threads that process data
- Python threads! :(

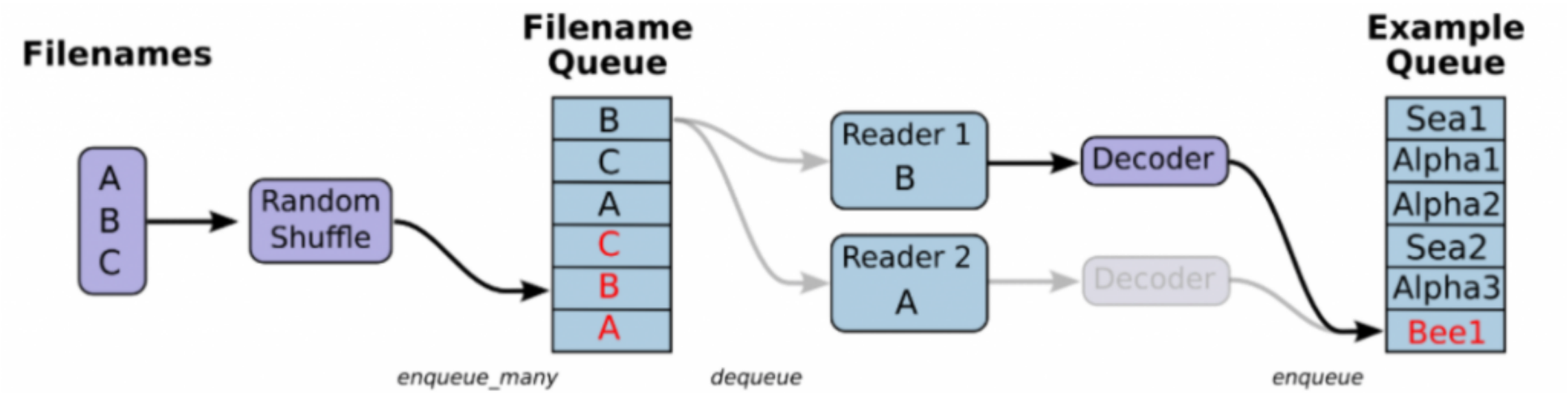


Image: https://www.tensorflow.org/api_guides/python/reading_data

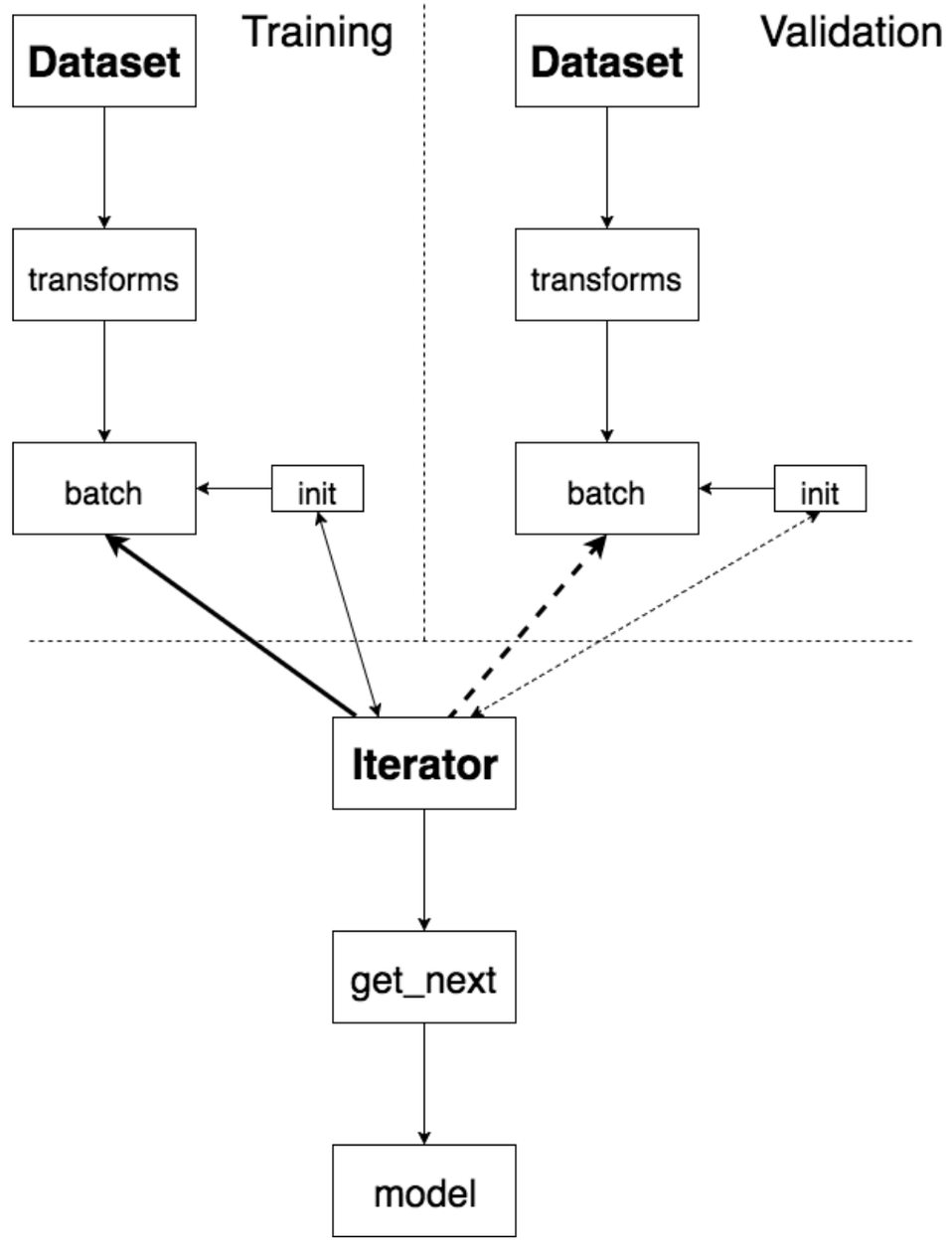
Tensorflow Datasets API

- Define a pipeline or logical plan
- Threads in C++
- Advanced features for good performance
  (caching [disk and in-memory], prefetching, sharding)

Image: https://github.com/tensorflow/tensorflow/blob/master/tensorflow/docs_src/performance/datasets_performance.md

Using Datasets

- Define an initial dataset (e.g. paths to images)
- Perform transformations (e.g. load images)
- Batch to desired size
- Define an iterator and use get_next to get data



NIH Chest Xrays

- Thorax region diseases (potentially multiple labels)
- Multiple patients, multiple visits per patient
- 112120 Scans at 1024x1024 (78GB)

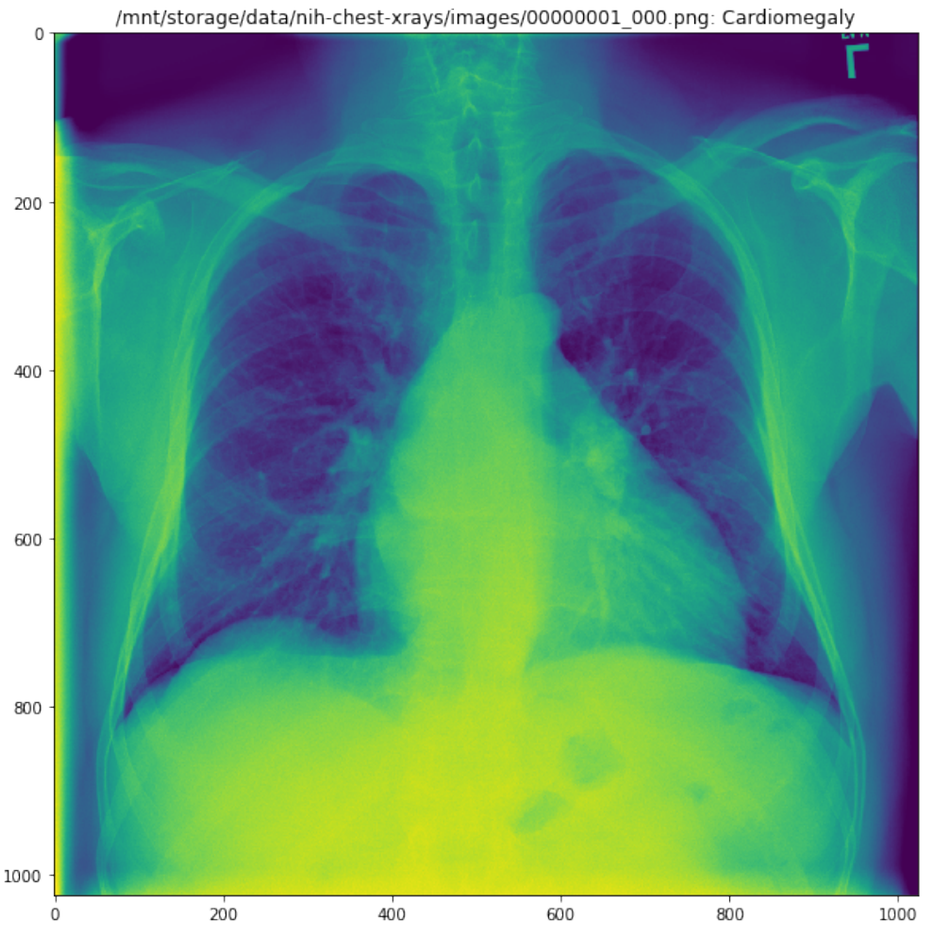
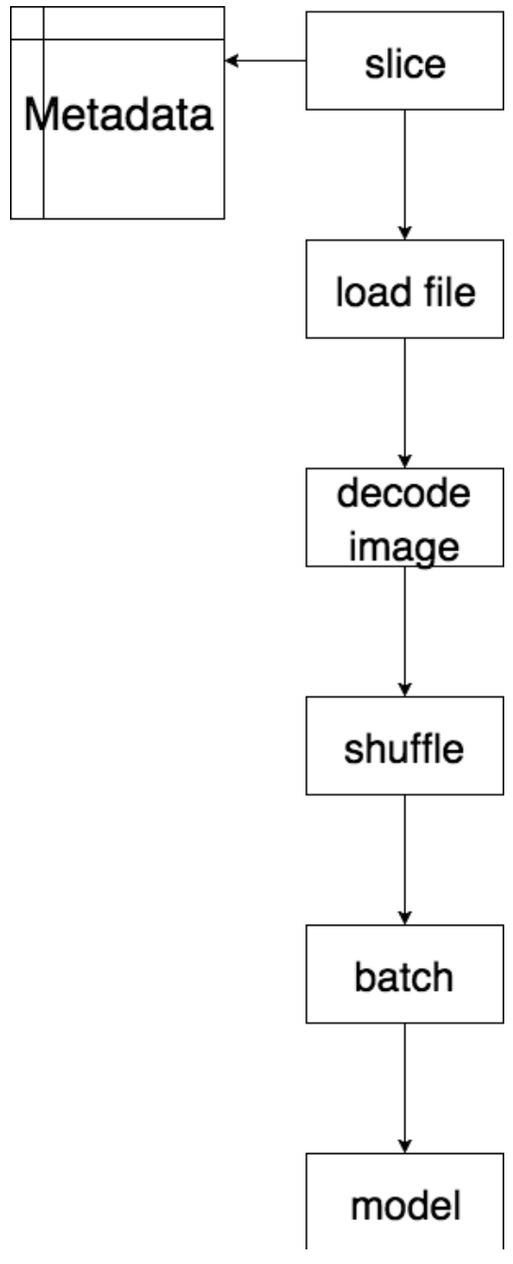


	Image Index	Finding Labels	Follow-up #	Patient ID	Patient Age	Patient Gender	View Position	OriginalImageWidth	OriginalImageHeight	OriginalImagePixelSpacing_x
0	00000001_000.png	Cardiomegaly	0	1	058Y	M	PA	2682	2749	0.143
1	00000001_001.png	Cardiomegaly|Emphysema	1	1	058Y	M	PA	2894	2729	0.143
2	00000001_002.png	Cardiomegaly|Effusion	2	1	058Y	M	PA	2500	2048	0.168
3	00000002_000.png	No Finding	0	2	081Y	M	PA	2500	2048	0.171
4	00000003_000.png	Hernia	0	3	081Y	F	PA	2582	2991	0.143

Building the pipeline

- Load metadata in Pandas
- Produce binary label matrix
- Build basic Dataset
- Load the file and decode images
- Prepare for training



Pitfalls (that we've run into)

- Shuffle required on all datasets when reinitialising
- Caching should happen before repeat
- Caching will create a lock-file that is released once the Dataset is out of range (tf.errors.OutOfRangeError thrown)

What more can you do?

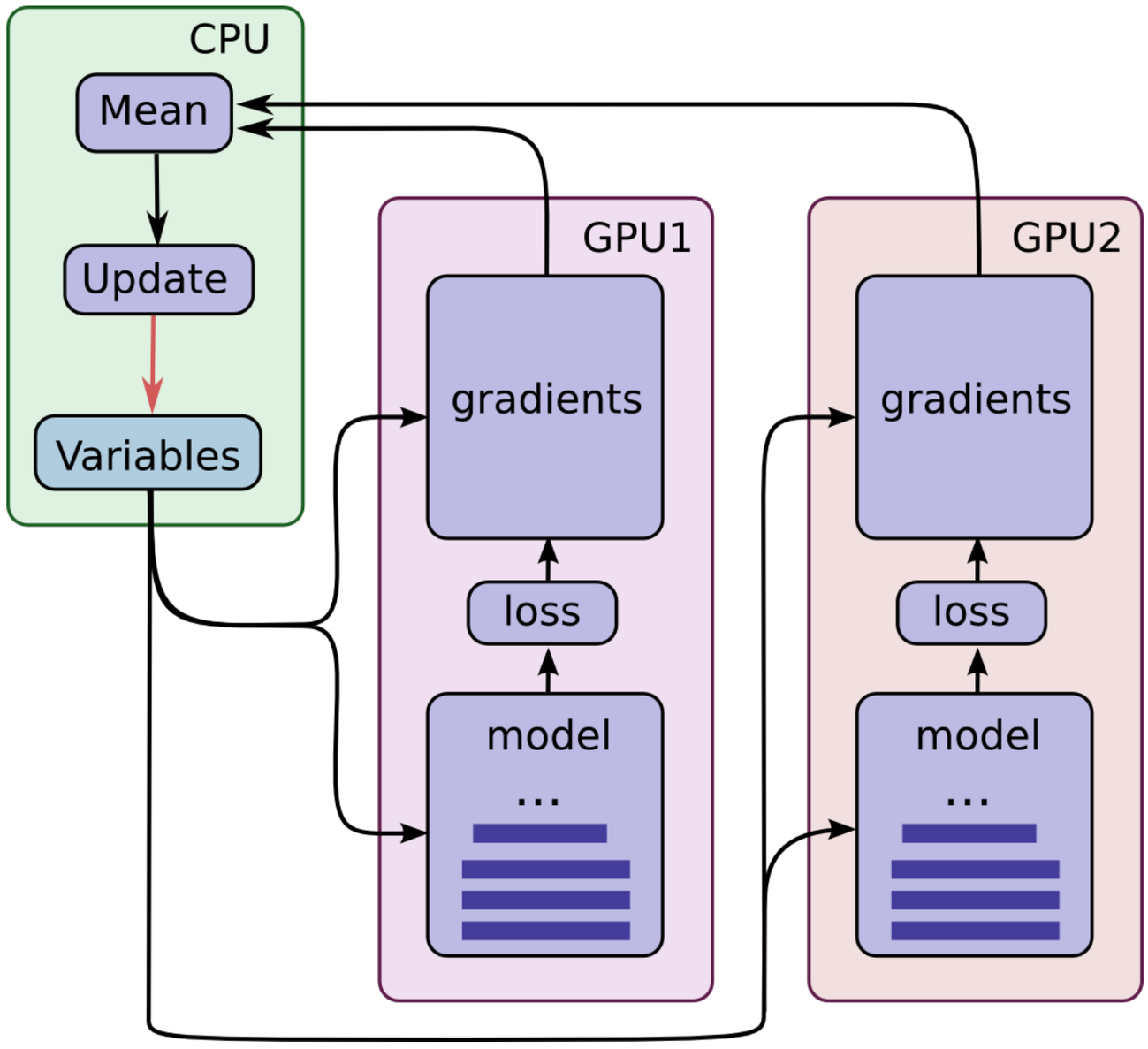
- Optimize for performance
  (prefetch, cache)

- Use with distributed Tensorflow (sharding)

- Use multiple datasets (training and validation)

- Initialise on feed_dict

- Good resources:
  https://www.tensorflow.org/programmers_guide/datasets
  https://github.com/tensorflow/tensorflow/blob/master/tensorflow/docs_src/performance/datasets_performance.md



Data Parallelisation Model (ResNetv2 152)

Image:
TensorFlow https://www.tensorflow.org/tutorials/deep_cnn

More questions?

Thank you for listening!

References

- Tensorflow Dataset guide
  https://www.tensorflow.org/versions/r1.4/programmers_guide/datasets

- Tensorflow general data reading guide
  https://www.tensorflow.org/versions/r1.4/api_guides/python/reading_data

- Details on TFRecords
  https://www.tensorflow.org/versions/r1.4/api_guides/python/python_io

- Yet unreleased performance guide for Datasets
  https://github.com/tensorflow/tensorflow/blob/master/tensorflow/docs_src/performance/datasets_performance.md

- Google blogpost on Datasets
  https://developers.googleblog.com/2017/09/introducing-tensorflow-datasets.html?m=1

- ResNet152 Implementation used
  https://github.com/tensorflow/models/blob/master/research/slim/nets/resnet_v2.py

- TensorFlow CIFAR10 Tutorial (Including Multi-GPU guide)
  https://www.tensorflow.org/tutorials/deep_cnn