AIDL: PROJECTS

Session 6 (2019/06/04): Full model implementation

tf wtf

https://docs.google.com/presentation/d/1Q4upkprHNVJ6DbGyRm_rwLJNyyP6jTeikq13Rimjm8U/edit?usp=sharing

Previously on *AIDL: Projects*...

Graph scopes

Name scopes (tf.name_scope). Affects:

- tf.Operation
- tf.Variable

Variable scopes (tf.variable_scope). Affects:

- tf.Operation
- tf.Variable
- tf.get_variable

Feeding data: so far

- tf.constants: simple scalar o lightweight numeric values that define some part of the graph and do not change between runs
- tf.placeholder & feed_dict: inject python data to the tensorflow graph at each run

Is feed_dict efficient? Not that much when dealing with big datasets!!

- Difficult to parallelize
- Input pipeline outside of the graph

Can we do something better? Of course! → **tf.data**

Feeding data: tf.data

Benefits:

- Off-the-shelf parallelization and scaling
- Embedded in the graph
- Sooooo much better in terms of coding
- All the input manipulation in one place

Main classes:

- tf.data.Dataset: represents the subgraph with the dataset operations
- tf.data.Iterator: it’s able to iterate through the samples in the dataset and yield those values to the network

Training models

Useful link by Karpathy

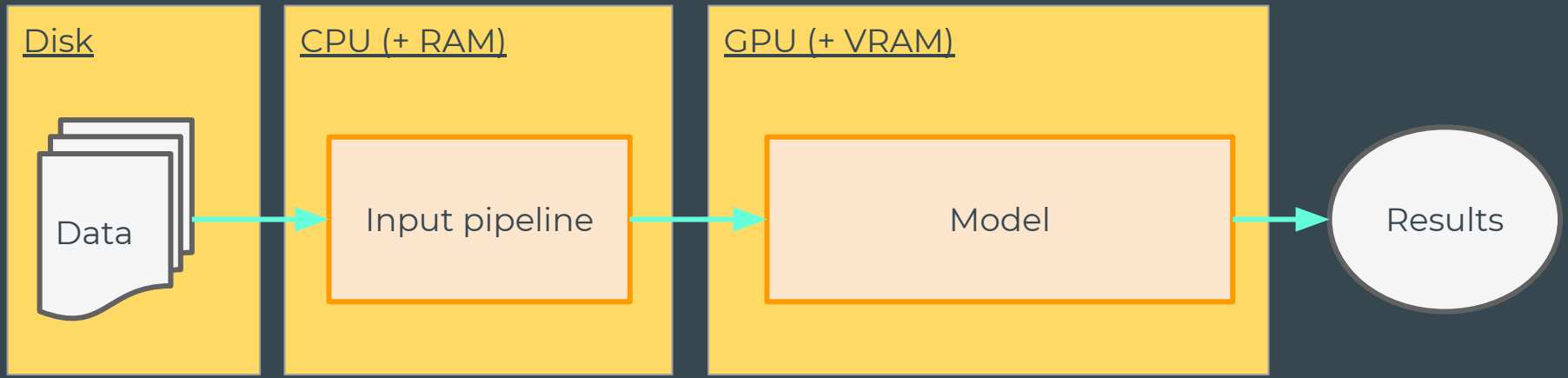
http://karpathy.github.io/2019/04/25/recipe/

Some definitions

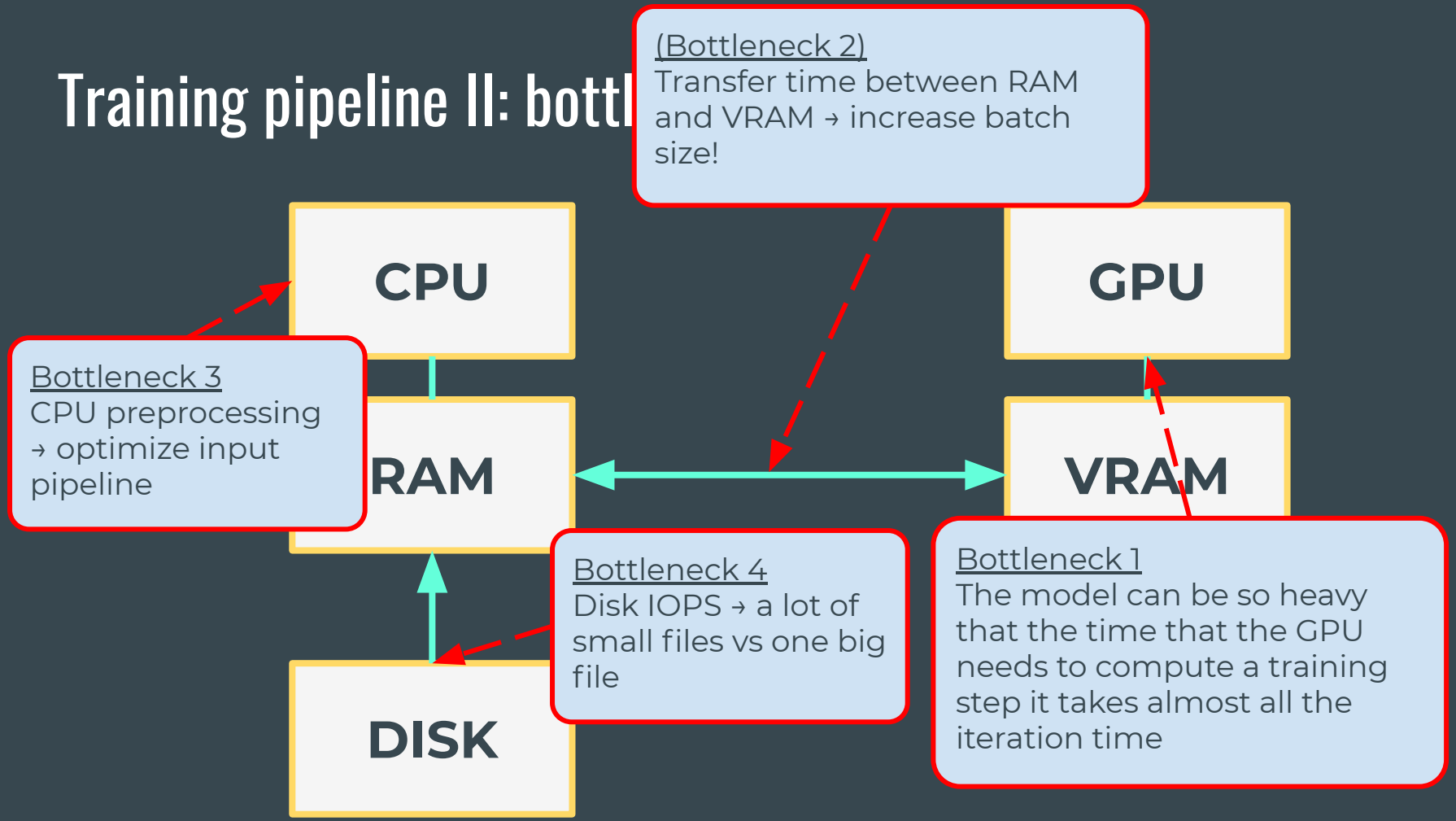
- Iteration time: clock time from one iteration to another
- Train step time: time needed to compute forward and backward pass
- Ingestion time: time needed to compose a batch and send it to the model

Iteration time = ingestion time + train step time

Training pipeline I: overview



Training pipeline II: bottl



Training steps: cookbook!

1. Get to know your data and its format
2. Build your input pipeline to ingest the dataset → try to run it in an infinite loop and monitor the iteration time, trying to reduce it
3. Implement the most basic model you can and try to train it
4. Look after an overfit!! (small subset of data)
5. Make yourself a baseline
6. Improve the model analysing the error gaps between bayes error-train and train-val

Input pipeline

Feeding data: main ops

Dataset:

- from_generator, from_tensor_slices: create dataset from source data
- shuffle: exactly that
- repeat: how many times should the dataset be repeated (epochs!)
- map: transform the samples
- batch: group samples into batches
- prefetch: create a buffer of items for improved performance
- make_one_shot_iterator: create the iterator of this dataset

Iterator:

- get_next: obtain the reference to the next samples of the dataset

Feeding data: example

```
dataset = tf.data.Dataset.range(100)
    .shuffle(10)
    .repeat(num_epochs)
    .map(parse_fn, num_parallel_calls=10)
    .batch(batch_size)
    .prefetch(prefetch_batches)

iterator = dataset.make_one_shot_iterator()
batch = iterator.get_next()
x, y = batch     # Instead of the placeholders!!!
```

Links:

https://www.tensorflow.org/guide/datasets
https://www.tensorflow.org/api_docs/python/tf/data/Dataset
https://www.tensorflow.org/guide/performance/datasets

Exercise IV

Implement ImageNet input pipeline

- Create an input pipeline for ImageNet
- How many iterations per second can you achieve?

Questions?