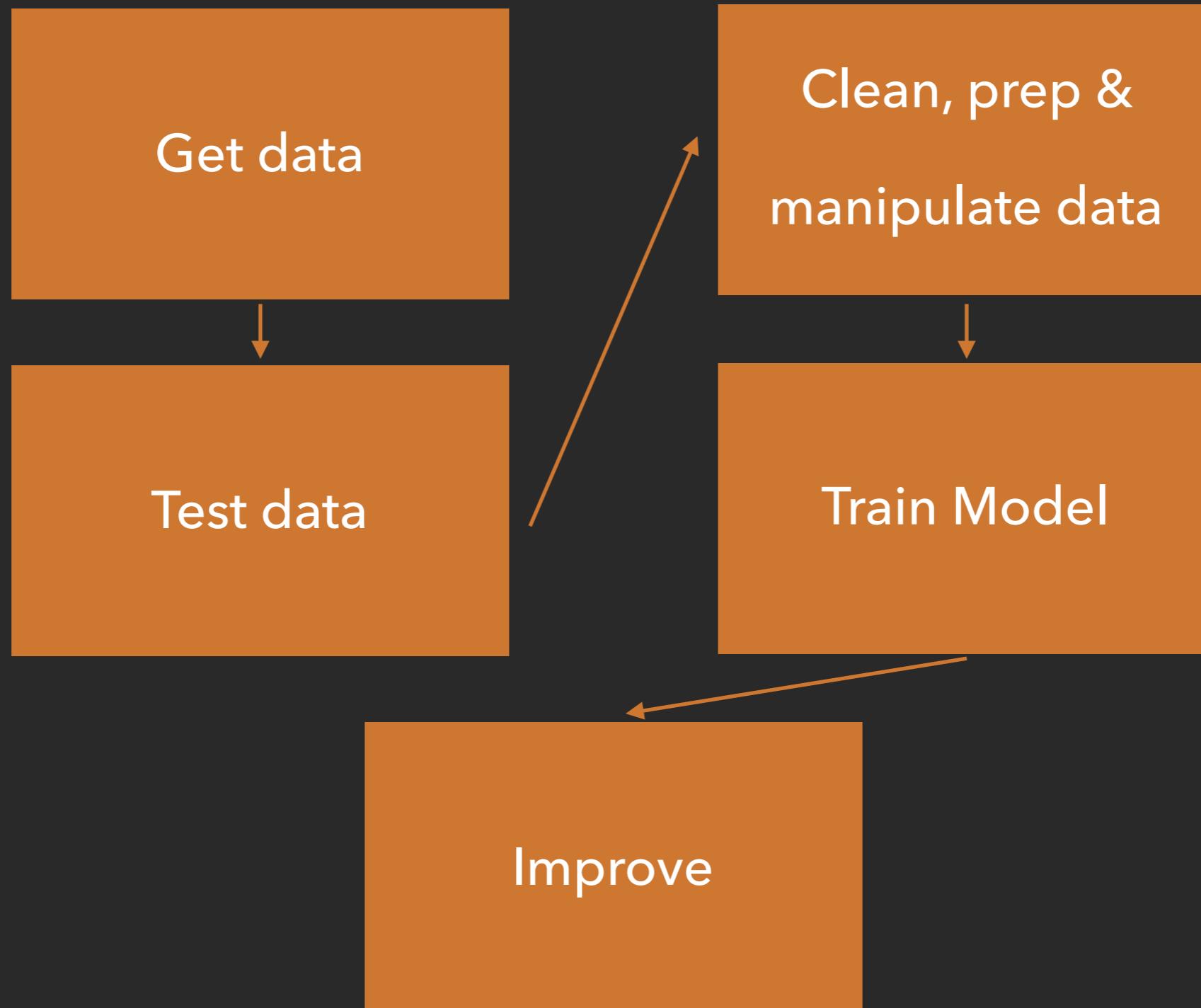


TENSORFLOW

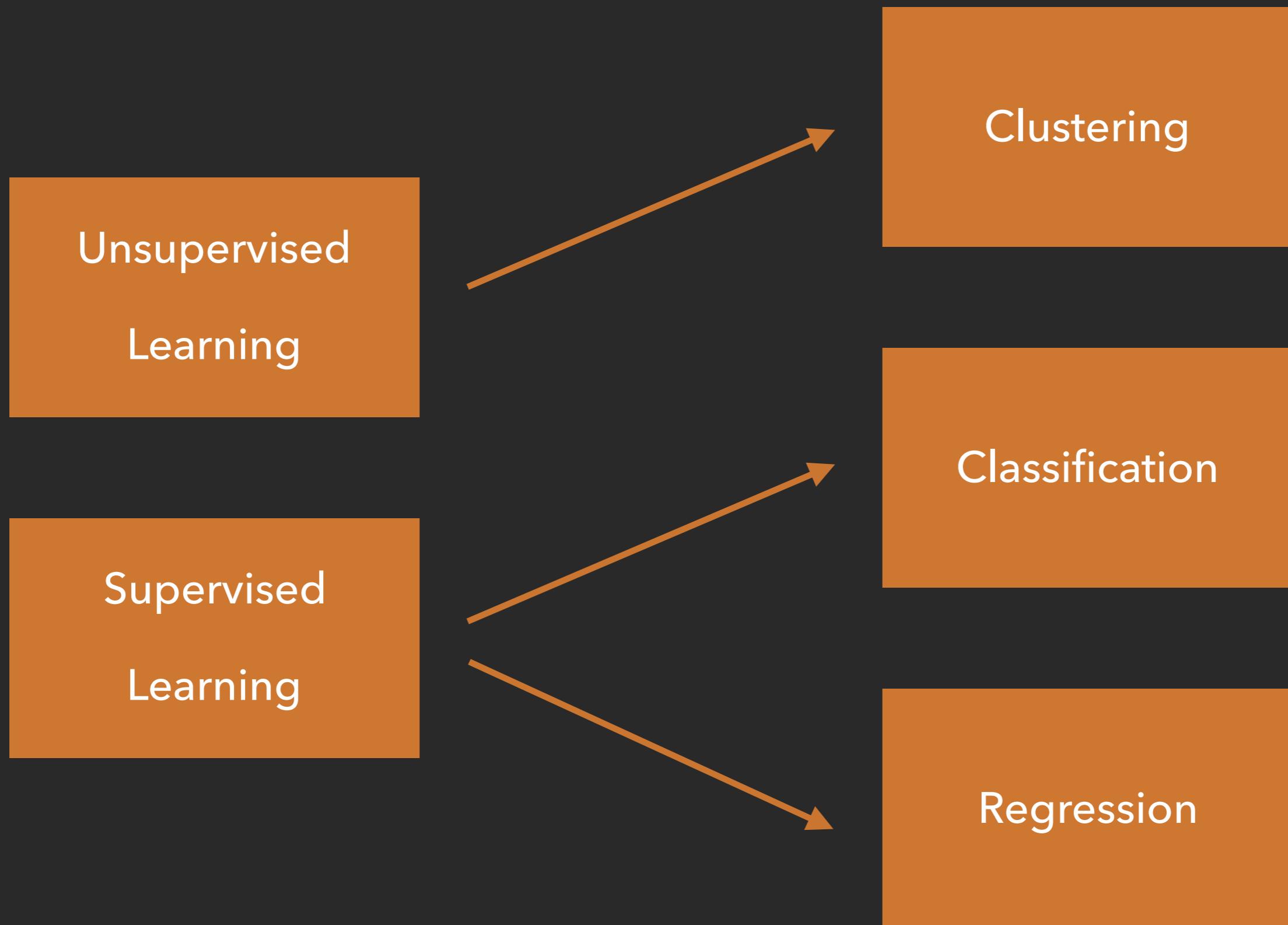
FOR ANDROID DEVELOPERS

JOE BIRCH - @HITHEREJOE - ANDROID LEAD @BUFFER - GDE @ANDROID

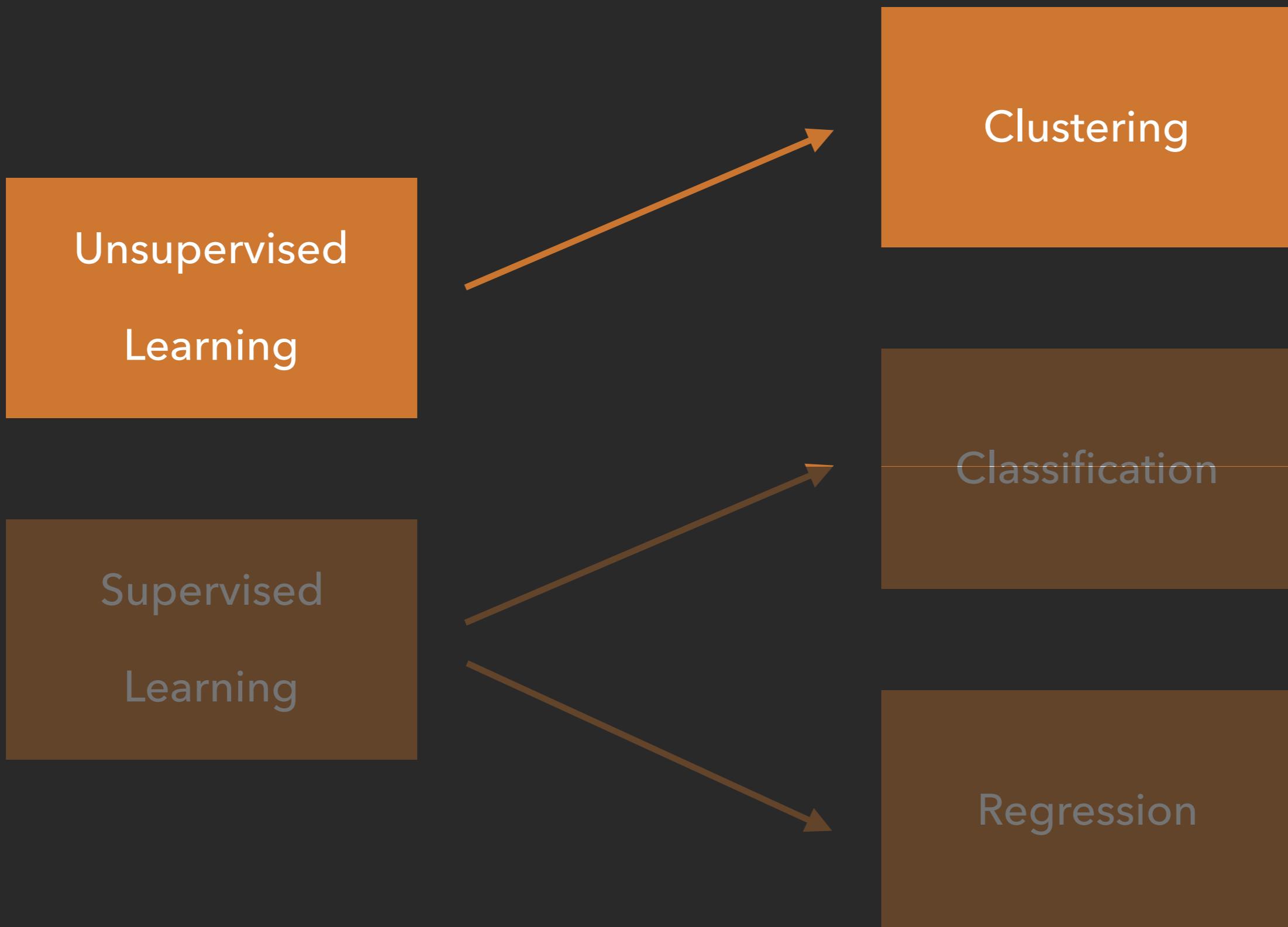
MACHINE LEARNING 101



MACHINE LEARNING 101



MACHINE LEARNING 101



MACHINE LEARNING 101

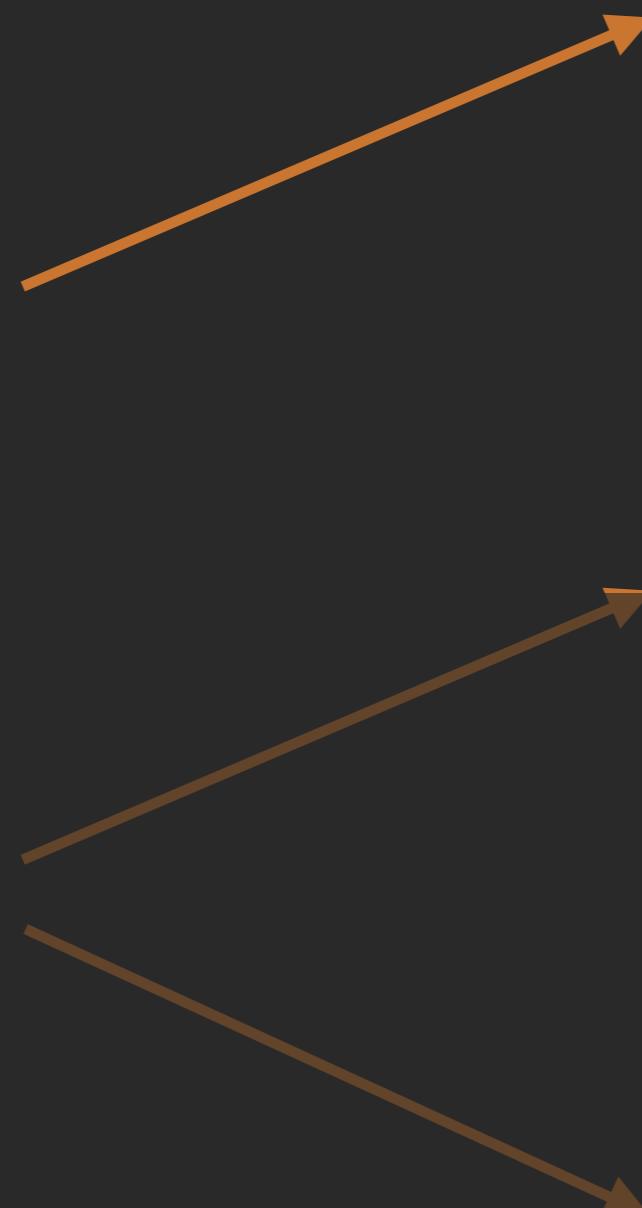
Unsupervised
Learning

Supervised
Learning

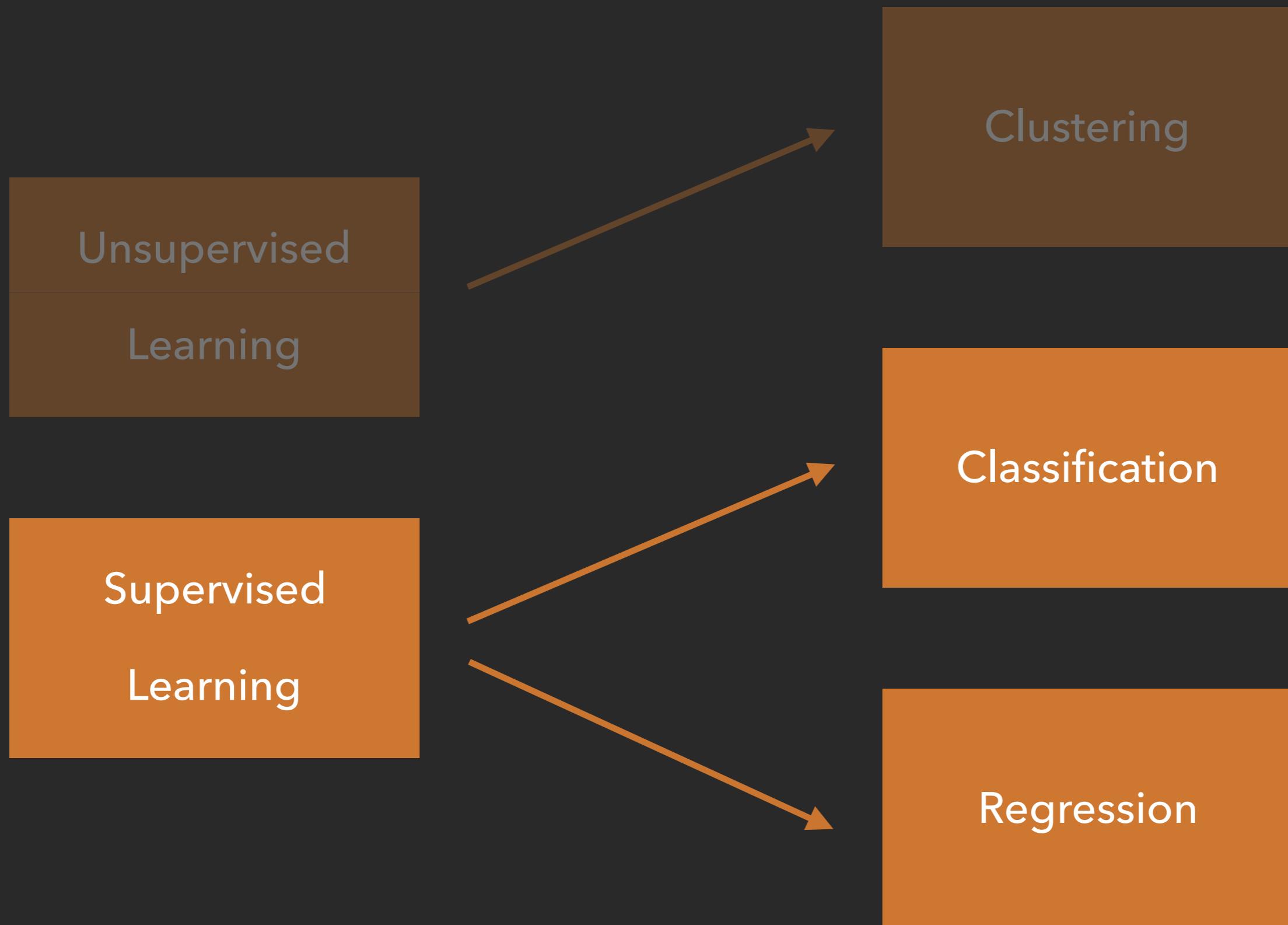
Clustering

Classification

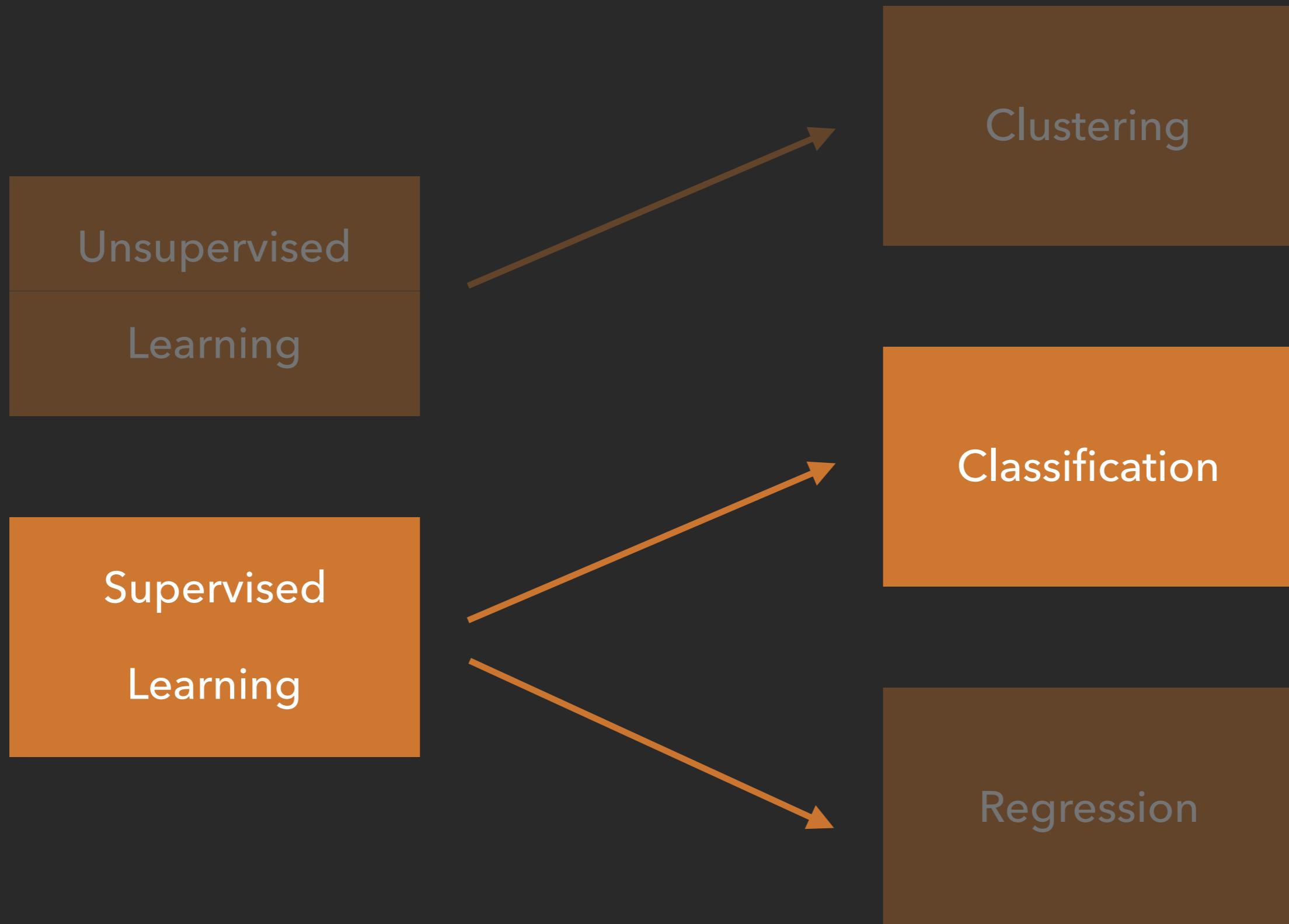
Regression



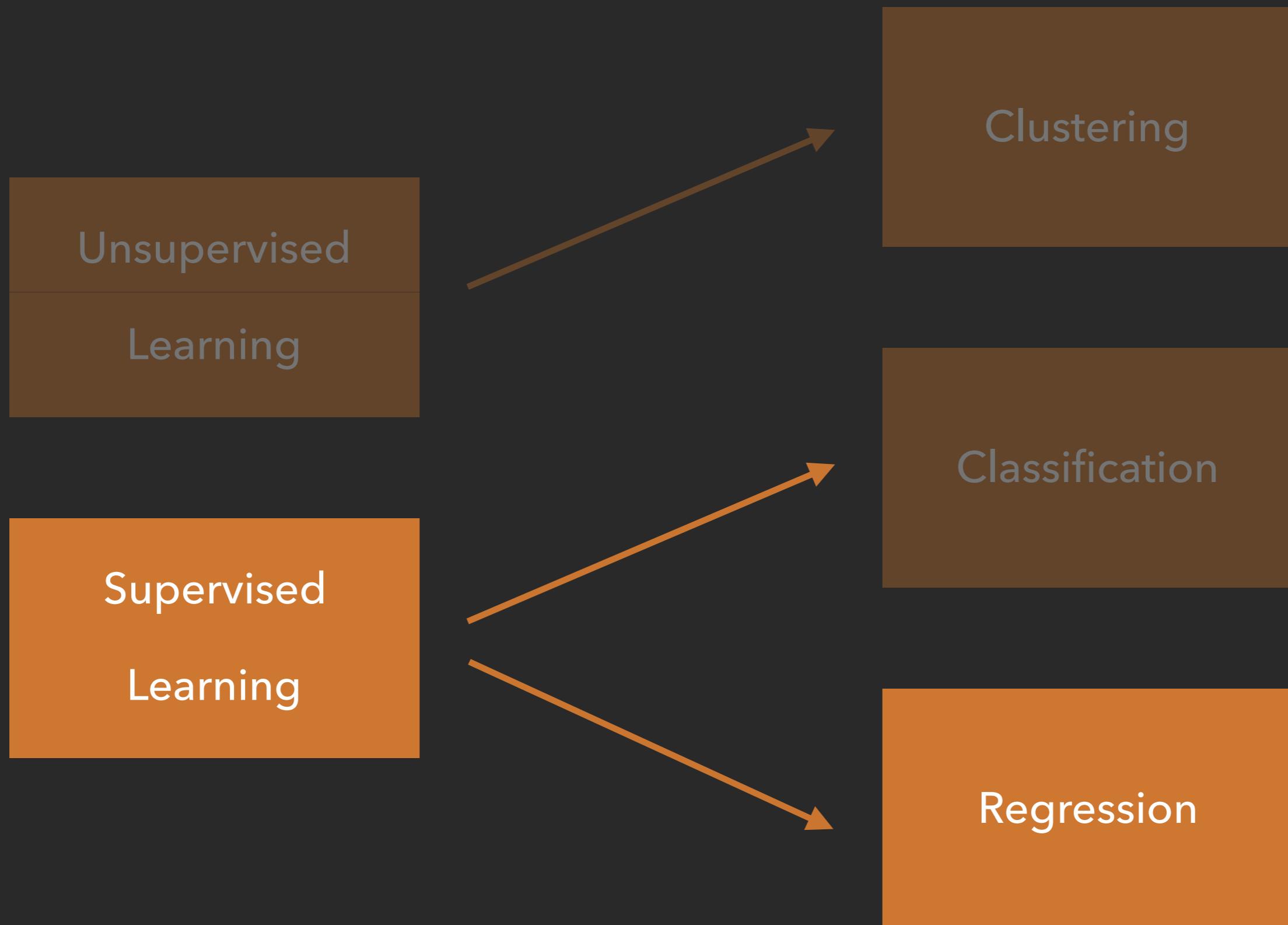
MACHINE LEARNING 101



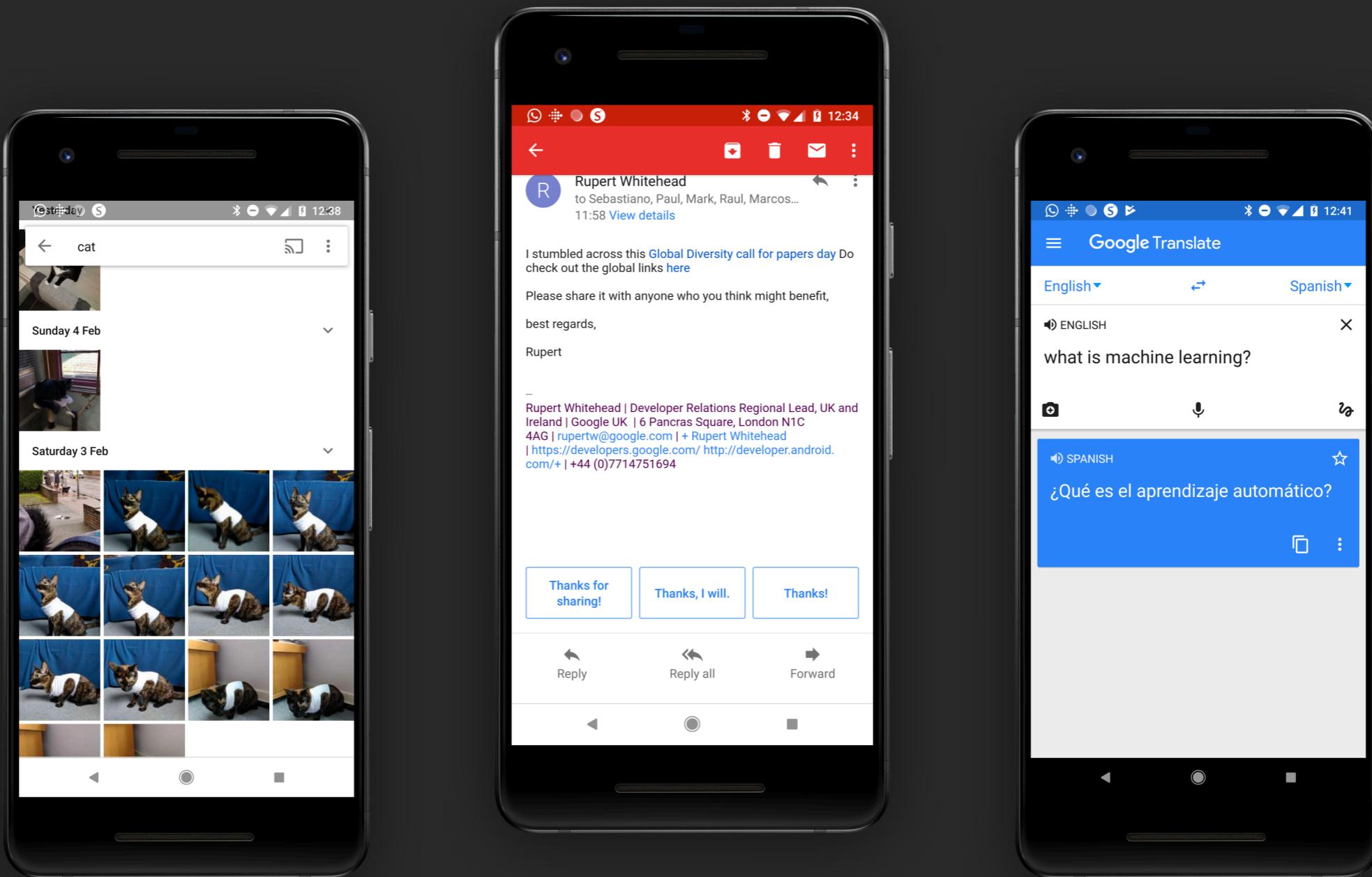
MACHINE LEARNING 101



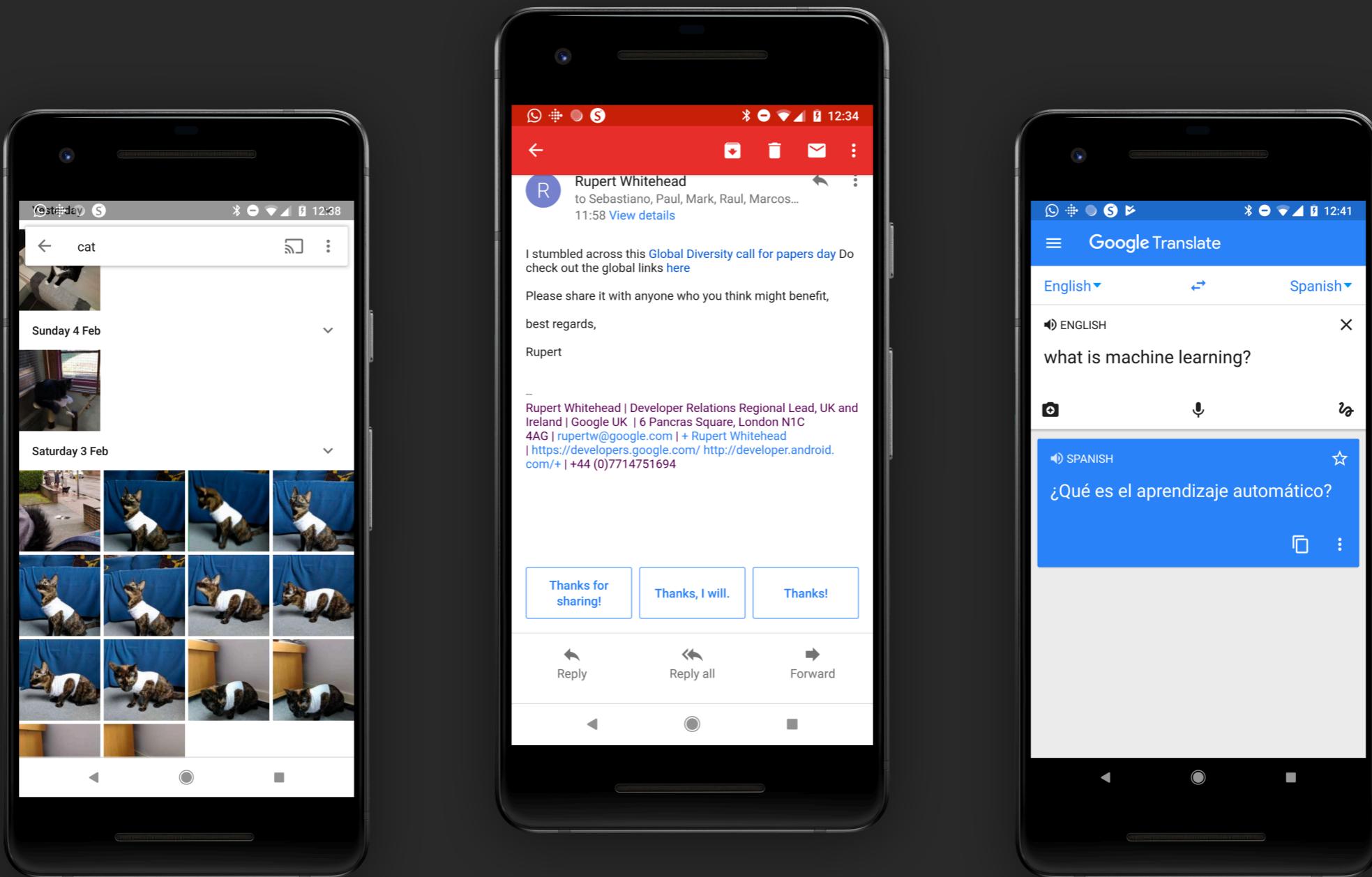
MACHINE LEARNING 101



MACHINE LEARNING AND MOBILE



MACHINE LEARNING AND MOBILE



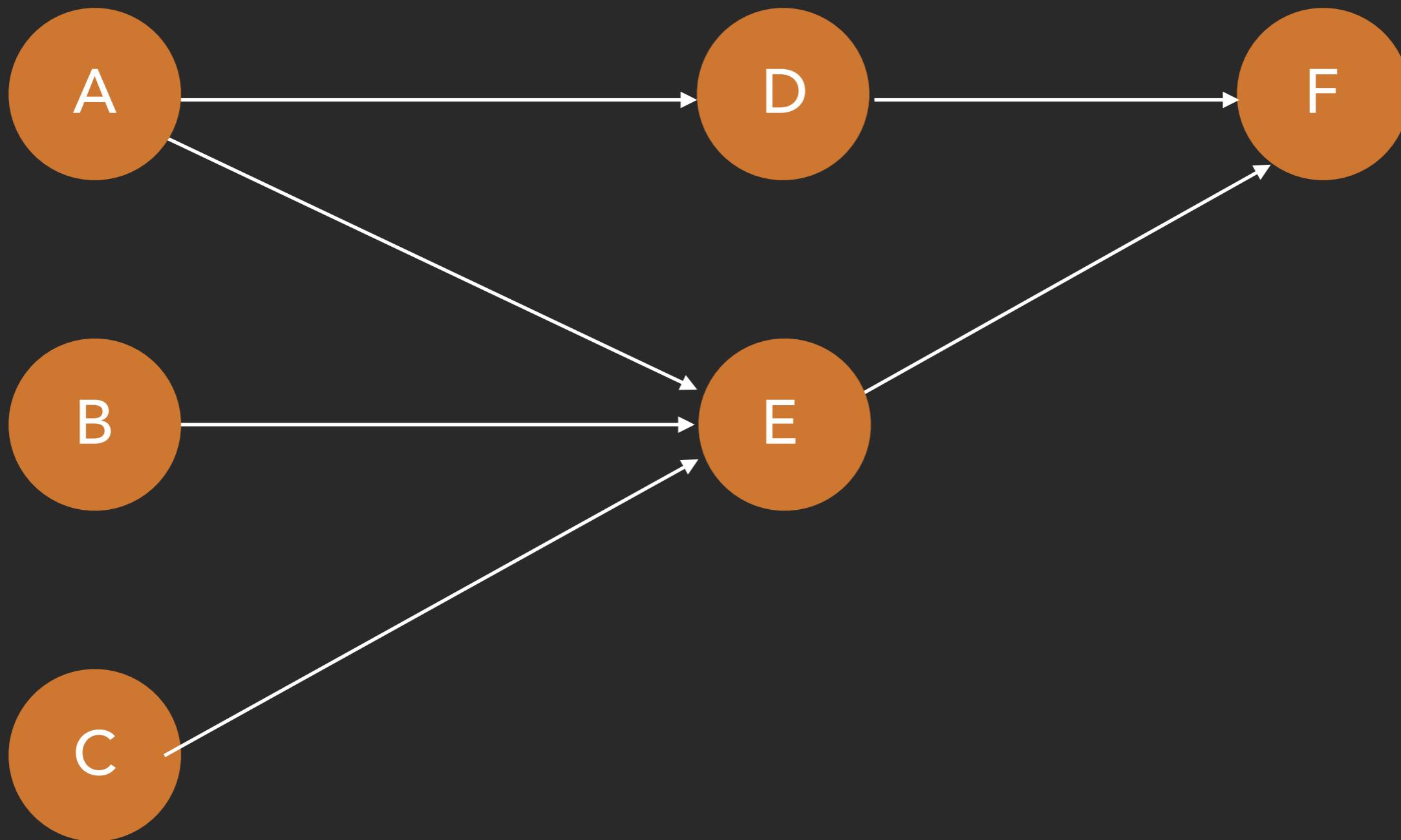
TENSORFLOW



TENSORFLOW

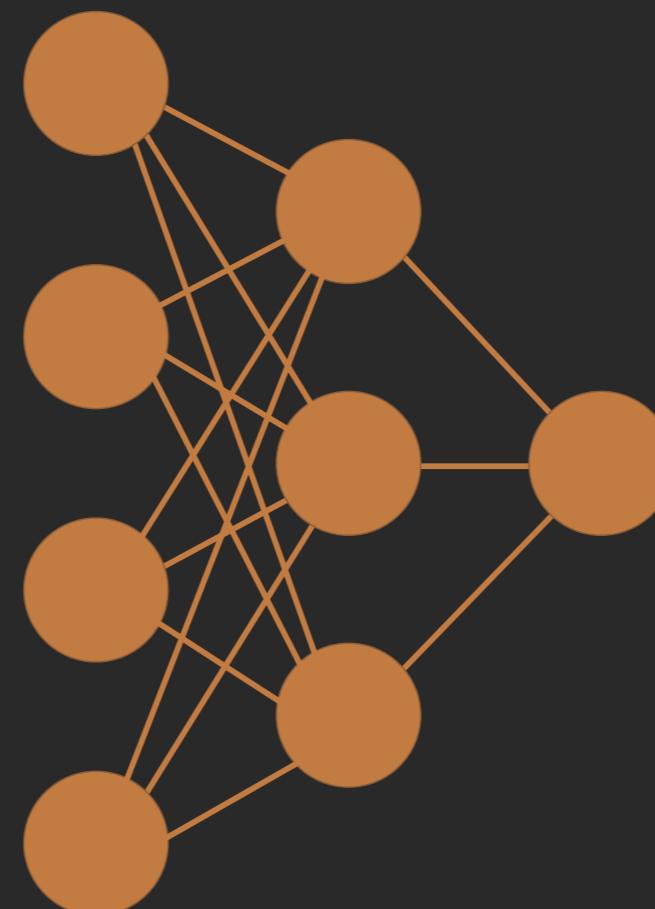


COMPUTATION GRAPHS



NEURAL NETWORKS

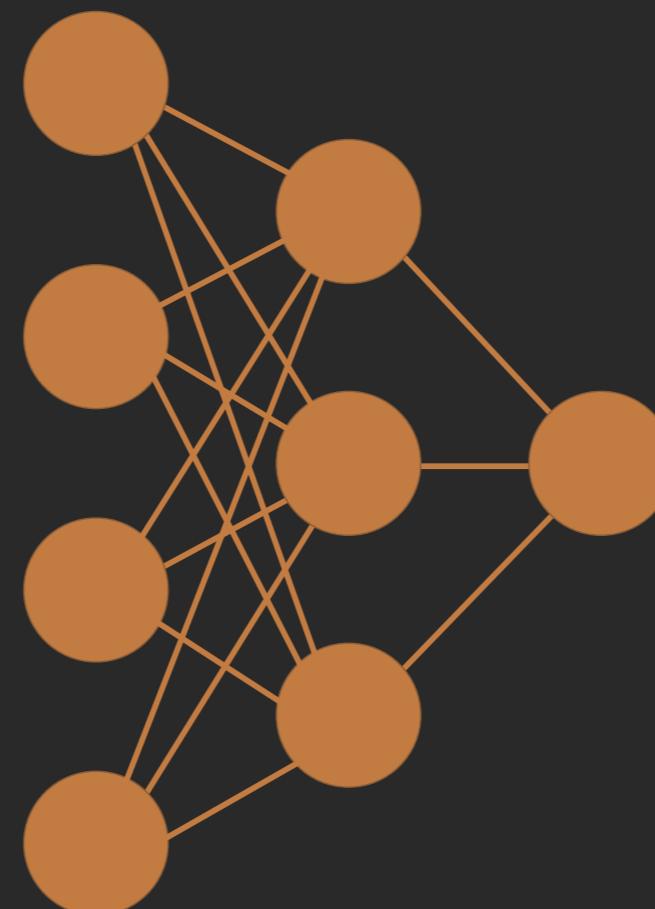
Some Image



Result

NEURAL NETWORKS

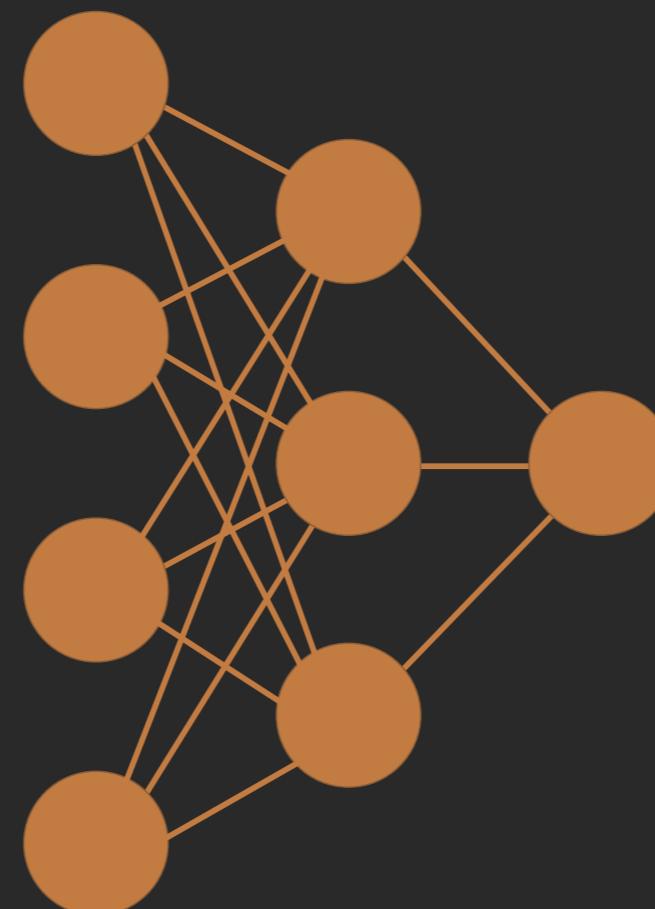
Some Image



Result

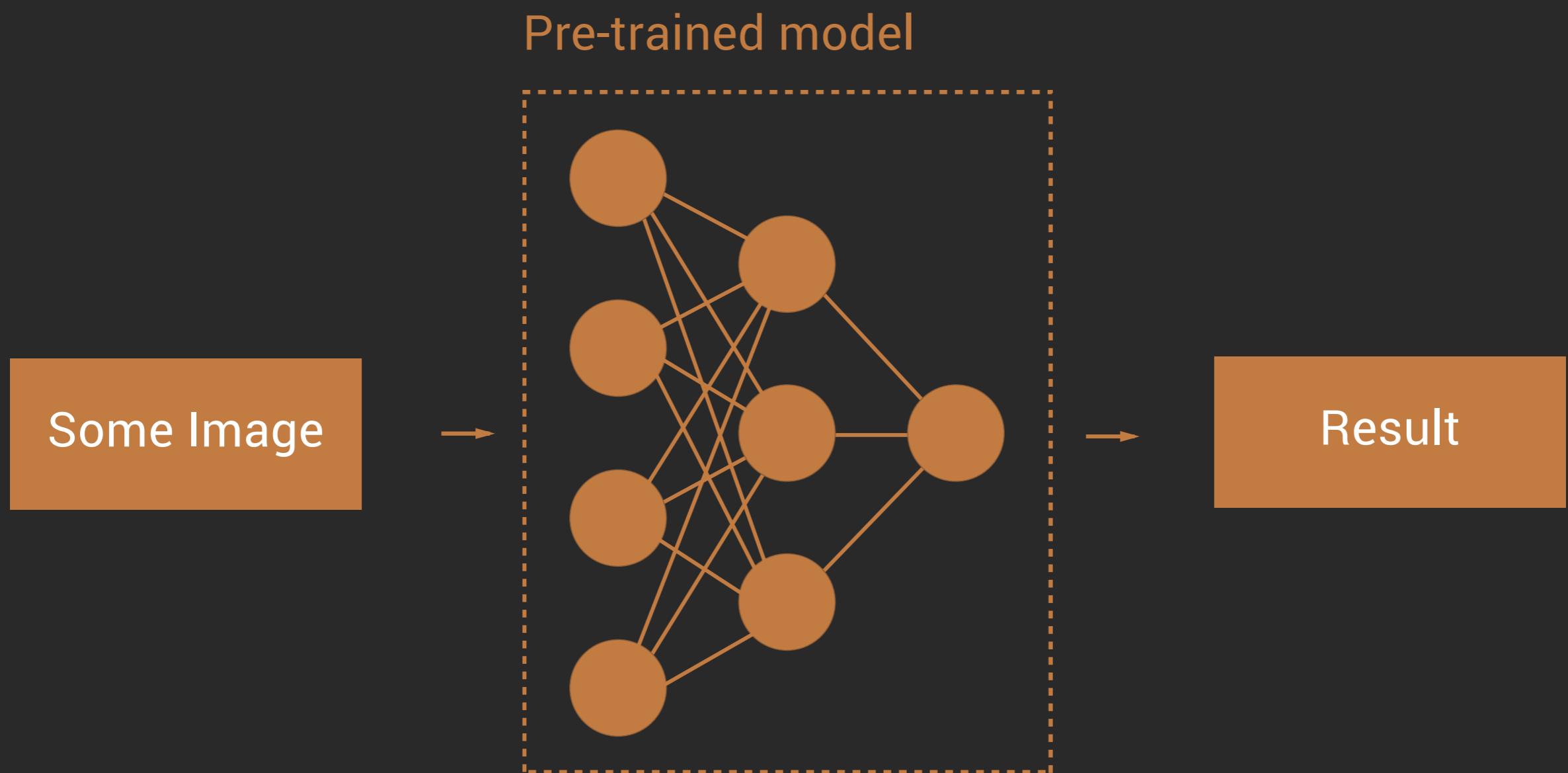
NEURAL NETWORKS

Some Image



Result

NEURAL NETWORKS



BUILDING OUR OWN IMAGE CLASSIFIER USING A MOBILE NET



TENSORBOARD

TensorBoard

SCALARS

GRAPHS

DISTRIBUTIONS

HISTOGRAMS

PROJECTOR

INACTIVE

▼

Show data download links

Ignore outliers in chart scaling

Tooltip sorting method: default ▾

Smoothing



Horizontal Axis

STEP

RELATIVE

WALL

Runs

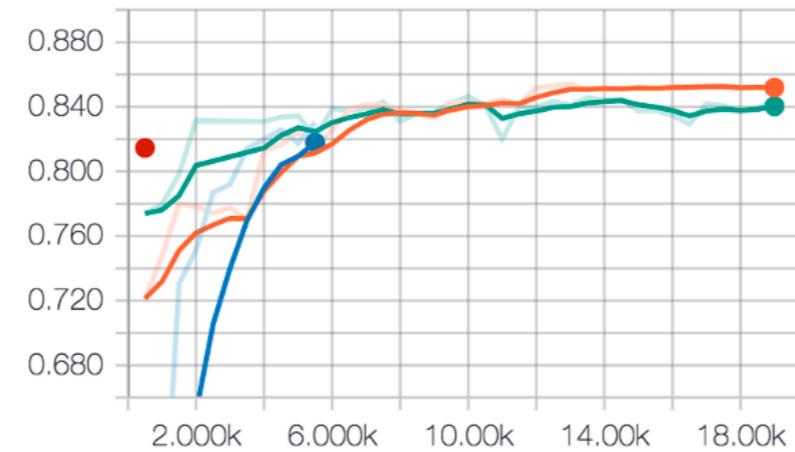
Write a regex to filter runs

- linear_1505895445
- linear_1505895445/eval
- deep01_1505895480
- deep01_1505895480/eval
- linear_1505895781
- linear_1505895781/eval
- deep01_1505896039

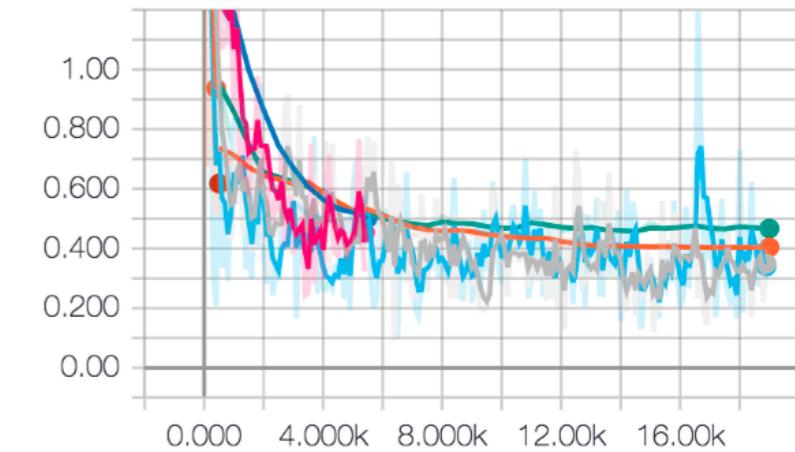


Tags matching ./*/ (all tags)

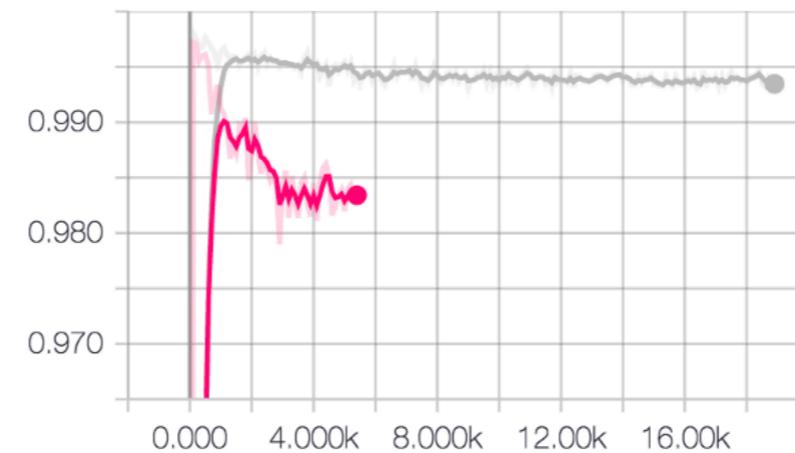
accuracy



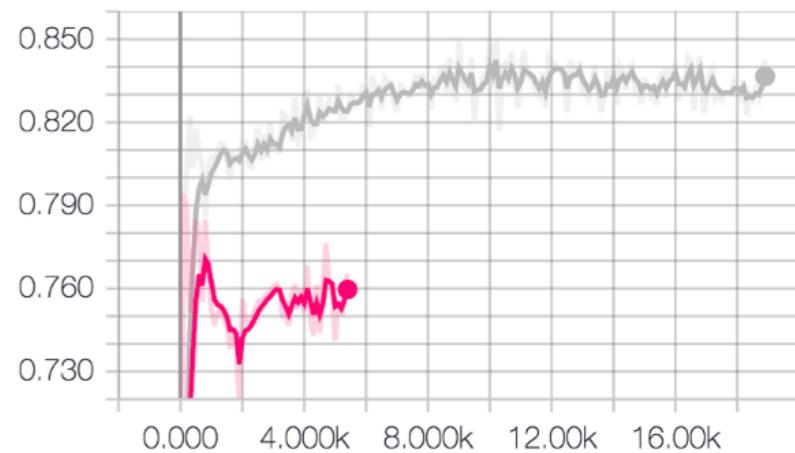
average_loss



dnn/dnn/hiddenlayer_0/fraction_of_zero_values



dnn/dnn/hiddenlayer_1/fraction_of_zero_values



TENSORBOARD

TensorBoard

SCALARS

GRAPHS

DISTRIBUTIONS

HISTOGRAMS

PROJECTOR

INACTIVE

▼

Show data download links

Ignore outliers in chart scaling

Tooltip sorting method: default ▾

Smoothing



Horizontal Axis

STEP RELATIVE WALL

Runs

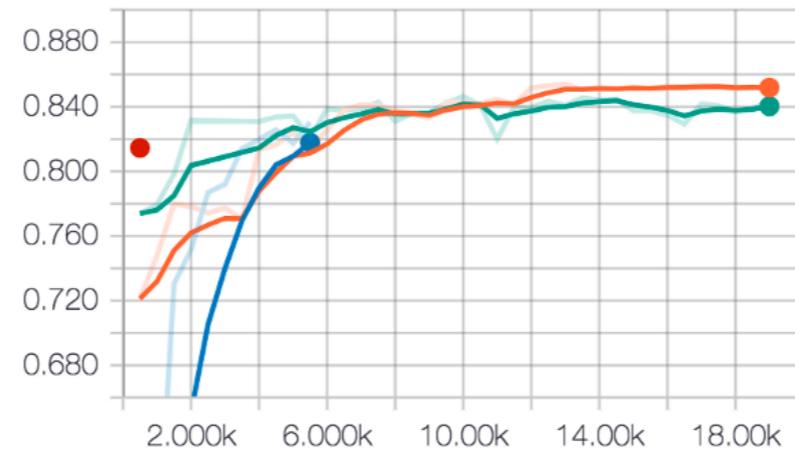
Write a regex to filter runs

- linear_1505895445
- linear_1505895445/eval
- deep01_1505895480
- deep01_1505895480/eval
- linear_1505895781
- linear_1505895781/eval
- deep01_1505896039

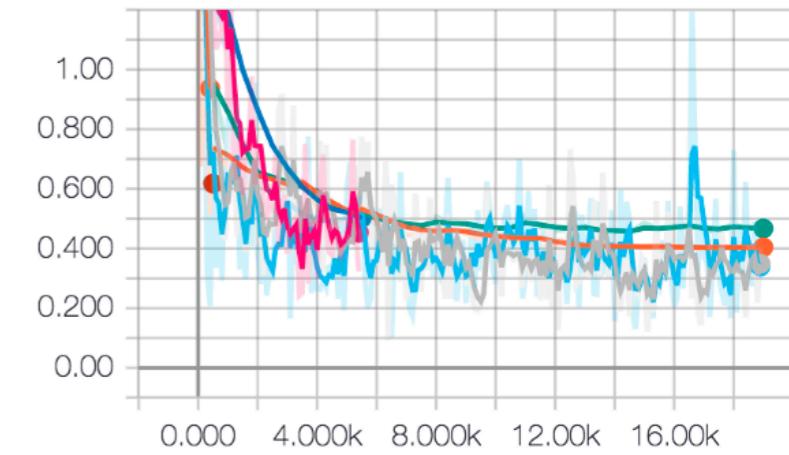


Tags matching ./*/(all tags)

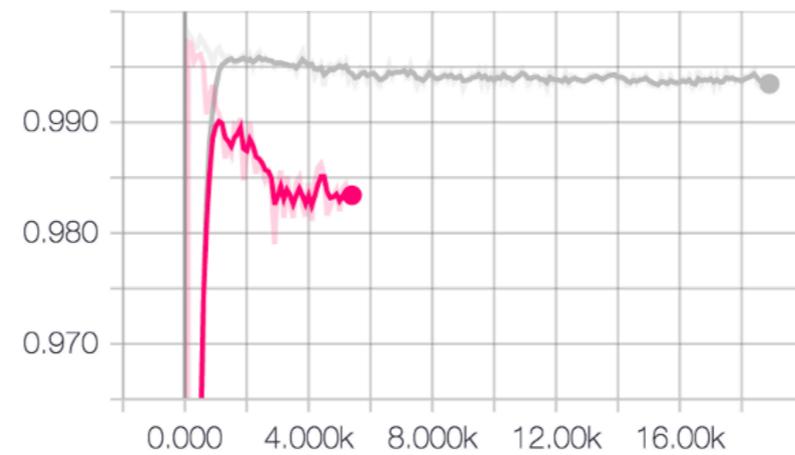
accuracy



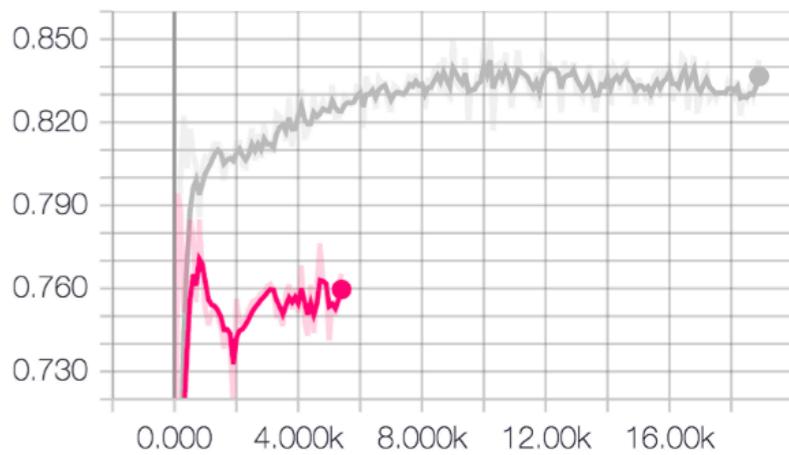
average_loss



dnn/dnn/hiddenlayer_0/fraction_of_zero_values



dnn/dnn/hiddenlayer_1/fraction_of_zero_values



TENSORBOARD

tensorboard --logdir tf_files/training_summaries &

○ ○ ○

```
1 TensorBoard 0.4.0rc3 at http://Joes-MacBook-Pro-2.local:6006 (Press CTRL+C to quit)
2
3 W0226 13:46:08.686425 Thread-7 program.py:285] EPIPE caused by 127.0.0.1:50538 in HTTP serving
4 W0226 13:46:08.698321 Thread-12 program.py:285] EPIPE caused by 127.0.0.1:50543 in HTTP serving
5 W0226 13:46:08.702178 Thread-8 program.py:285] EPIPE caused by 127.0.0.1:50539 in HTTP serving
6 W0226 13:46:08.801446 Thread-9 program.py:285] EPIPE caused by 127.0.0.1:50540 in HTTP serving
7 W0226 13:46:08.837674 Thread-10 program.py:285] EPIPE caused by 127.0.0.1:50541 in HTTP serving
8 W0226 13:46:08.843611 Thread-11 program.py:285] EPIPE caused by 127.0.0.1:50542 in HTTP serving
9 W0226 19:19:36.557772 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'inception_v3/train'
10 W0226 19:19:36.614542 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'inception_v3/validation'
11 W0226 19:25:46.444421 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'inception_v3/train'
12 W0226 19:25:46.444560 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'inception_v3/validation'
13 W0226 19:26:17.917988 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'mobilenet_0.50_224/train'
14 W0226 19:26:17.918148 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'inception_v3/train'
15 W0226 19:26:17.918269 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'inception_v3/validation'
16 W0226 19:26:17.918428 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'default/train'
17 W0226 19:26:17.918551 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'retrained'
18 W0226 19:26:17.918701 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'optimized'
19 W0226 19:26:17.918835 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'train'
20 W0226 19:26:17.918940 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'default/validation'
21 W0226 19:26:17.919131 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'mobilenet_0.50_224/validation'
22 W0226 19:26:17.919234 Reloader plugin_event_multiplexer.py:203] Deleting accumulator 'validation'
```

TRAINING DATA

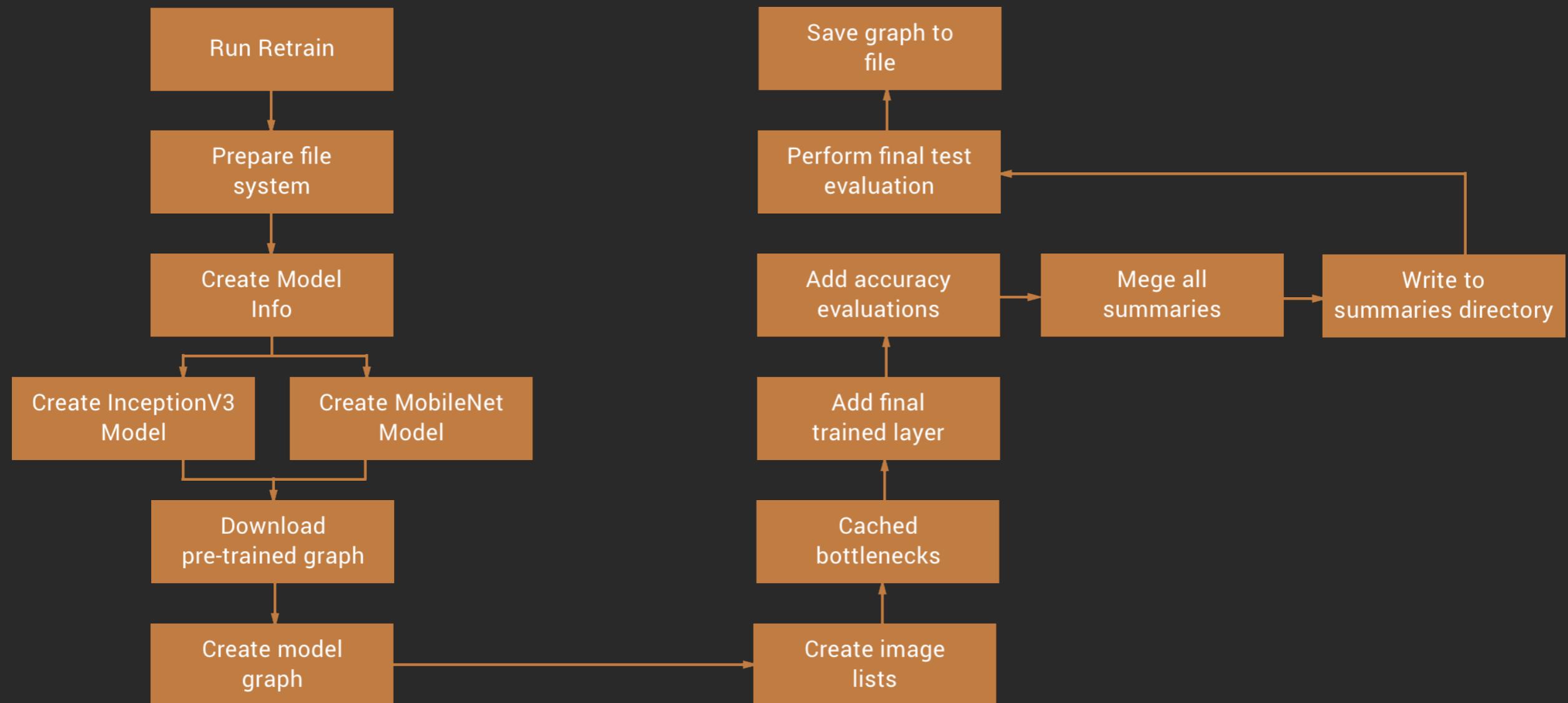
▶	📁 Apple Braeburn	8 Feb 2018, 00:23
▶	📁 Apple Granny Smith	8 Feb 2018, 00:23
▶	📁 Apple Red 1	8 Feb 2018, 00:23
▶	📁 Avocado	8 Feb 2018, 00:23
▶	📁 Banana	8 Feb 2018, 00:23
▶	📁 Cherry	8 Feb 2018, 00:23
▶	📁 Kiwi	8 Feb 2018, 00:23
▶	📁 Lemon	8 Feb 2018, 00:23
	LICENSE.txt	9 Feb 2016, 02:59
▶	📁 Limes	8 Feb 2018, 00:23
▶	📁 Peach	8 Feb 2018, 00:23
▶	📁 Pineapple	8 Feb 2018, 00:23

RETRAINING THE MODEL



```
1 python -m scripts.retrain \
2   --bottleneck_dir=tf_files/bottlenecks \
3   --how_many_training_steps=500 \
4   --model_dir=tf_files/models/ \
5   --summaries_dir=tf_files/training_summaries/mobilenet_0.50_224 \
6   --output_graph=tf_files/retrained_graph.pb \
7   --output_labels=tf_files/retrained_labels.txt \
8   --architecture="mobilenet_0.50_224" \
9   --image_dir=tf_files/fruit_photos
```

RETRAINING THE MODEL



RETRAINING THE MODEL



```
1 python -m scripts.retrain \
2   --bottleneck_dir=tf_files/bottlenecks \
3   --how_many_training_steps=500 \
4   --model_dir=tf_files/models/ \
5   --summaries_dir=tf_files/training_summaries/mobilenet_0.50_224 \
6   --output_graph=tf_files/retrained_graph.pb \
7   --output_labels=tf_files/retrained_labels.txt \
8   --architecture="mobilenet_0.50_224" \
9   --image_dir=tf_files/fruit_photos
```

RETRAINING THE MODEL



```
1 python -m scripts.retrain \
2   --bottleneck_dir=tf_files/bottlenecks \
3   --how_many_training_steps=500 \
4   --model_dir=tf_files/models/ \
5   --summaries_dir=tf_files/training_summaries/mobilenet_0.50_224 \
6   --output_graph=tf_files/retrained_graph.pb \
7   --output_labels=tf_files/retrained_labels.txt \
8   --architecture="mobilenet_0.50_224" \
9   --image_dir=tf_files/fruit_photos
```

RETRAINING THE MODEL



```
1 python -m scripts.retrain \
2   --bottleneck_dir=tf_files/bottlenecks \
3   --how_many_training_steps=500 \
4   --model_dir=tf_files/models/ \
5   --summaries_dir=tf_files/training_summaries/mobilenet_0.50_224 \
6   --output_graph=tf_files/retrained_graph.pb \
7   --output_labels=tf_files/retrained_labels.txt \
8   --architecture="mobilenet_0.50_224" \
9   --image_dir=tf_files/fruit_photos
```

RETRAINING THE MODEL



```
1 python -m scripts.retrain \
2   --bottleneck_dir=tf_files/bottlenecks \
3   --how_many_training_steps=500 \
4   --model_dir=tf_files/models/ \
5   --summaries_dir=tf_files/training_summaries/mobilenet_0.50_224 \
6   --output_graph=tf_files/retrained_graph.pb \
7   --output_labels=tf_files/retrained_labels.txt \
8   --architecture="mobilenet_0.50_224" \
9   --image_dir=tf_files/fruit_photos
```

RETRAINING THE MODEL



```
1 python -m scripts.retrain \
2   --bottleneck_dir=tf_files/bottlenecks \
3   --how_many_training_steps=500 \
4   --model_dir=tf_files/models/ \
5   --summaries_dir=tf_files/training_summaries/mobilenet_0.50_224 \
6   --output_graph=tf_files/retrained_graph.pb \
7   --output_labels=tf_files/retrained_labels.txt \
8   --architecture="mobilenet_0.50_224" \
9   --image_dir=tf_files/fruit_photos
```

RETRAINING THE MODEL



```
1 python -m scripts.retrain \
2   --bottleneck_dir=tf_files/bottlenecks \
3   --how_many_training_steps=500 \
4   --model_dir=tf_files/models/ \
5   --summaries_dir=tf_files/training_summaries/mobilenet_0.50_224 \
6   --output_graph=tf_files/retrained_graph.pb \
7   --output_labels=tf_files/retrained_labels.txt \
8   --architecture="mobilenet_0.50_224" \
9   --image_dir=tf_files/fruit_photos
```

RETRAINING THE MODEL



```
1 python -m scripts.retrain \
2   --bottleneck_dir=tf_files/bottlenecks \
3   --how_many_training_steps=500 \
4   --model_dir=tf_files/models/ \
5   --summaries_dir=tf_files/training_summaries/mobilenet_0.50_224 \
6   --output_graph=tf_files/retrained_graph.pb \
7   --output_labels=tf_files/retrained_labels.txt \
8   --architecture="mobilenet_0.50_224" \
9   --image_dir=tf_files/fruit_photos
```

RETRAINING THE MODEL



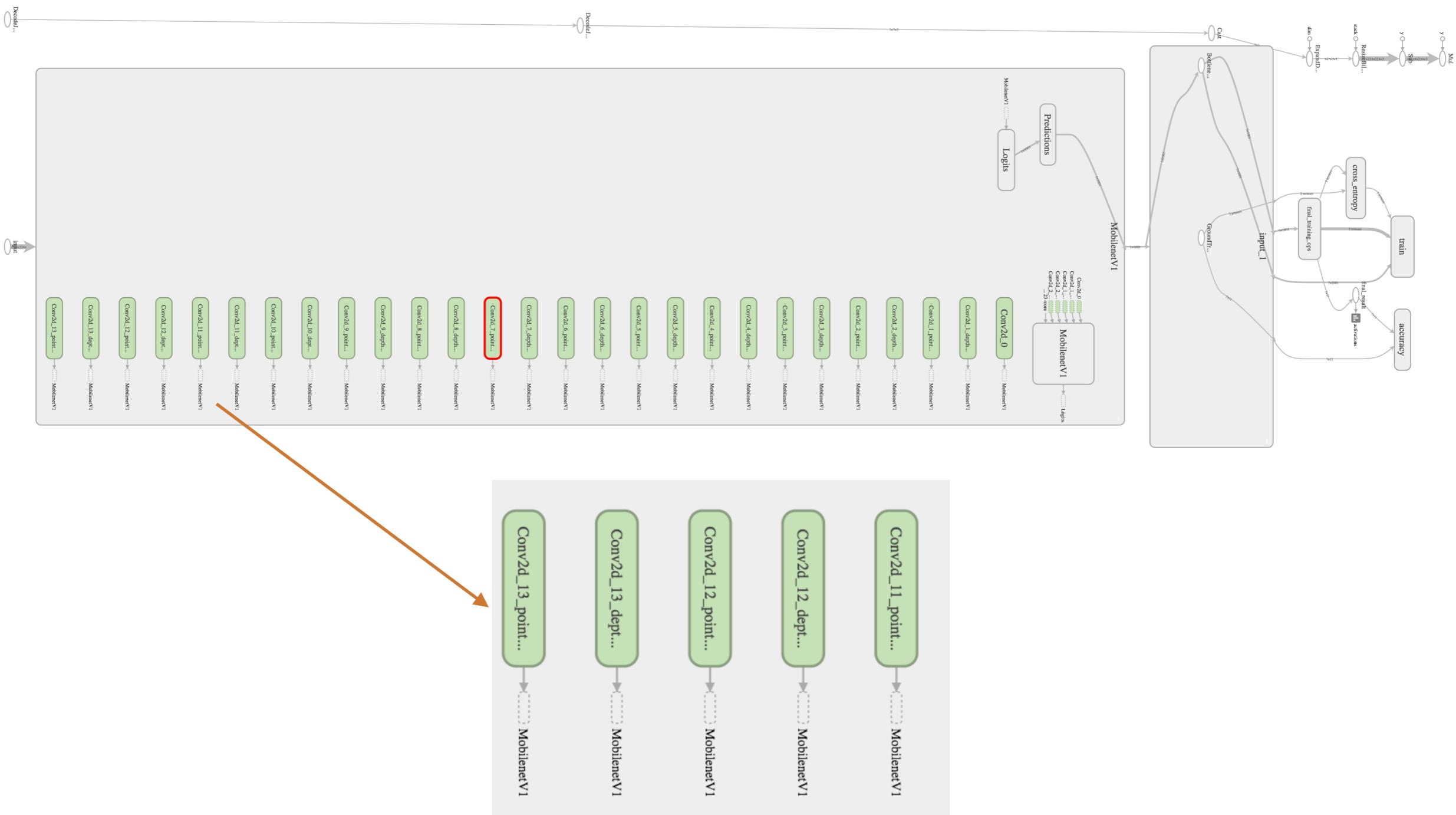
```
1 python -m scripts.retrain \
2   --bottleneck_dir=tf_files/bottlenecks \
3   --how_many_training_steps=500 \
4   --model_dir=tf_files/models/ \
5   --summaries_dir=tf_files/training_summaries/mobilenet_0.50_224 \
6   --output_graph=tf_files/retrained_graph.pb \
7   --output_labels=tf_files/retrained_labels.txt \
8   --architecture="mobilenet_0.50_224" \
9   --image_dir=tf_files/fruit_photos
```

RETRAINING THE MODEL

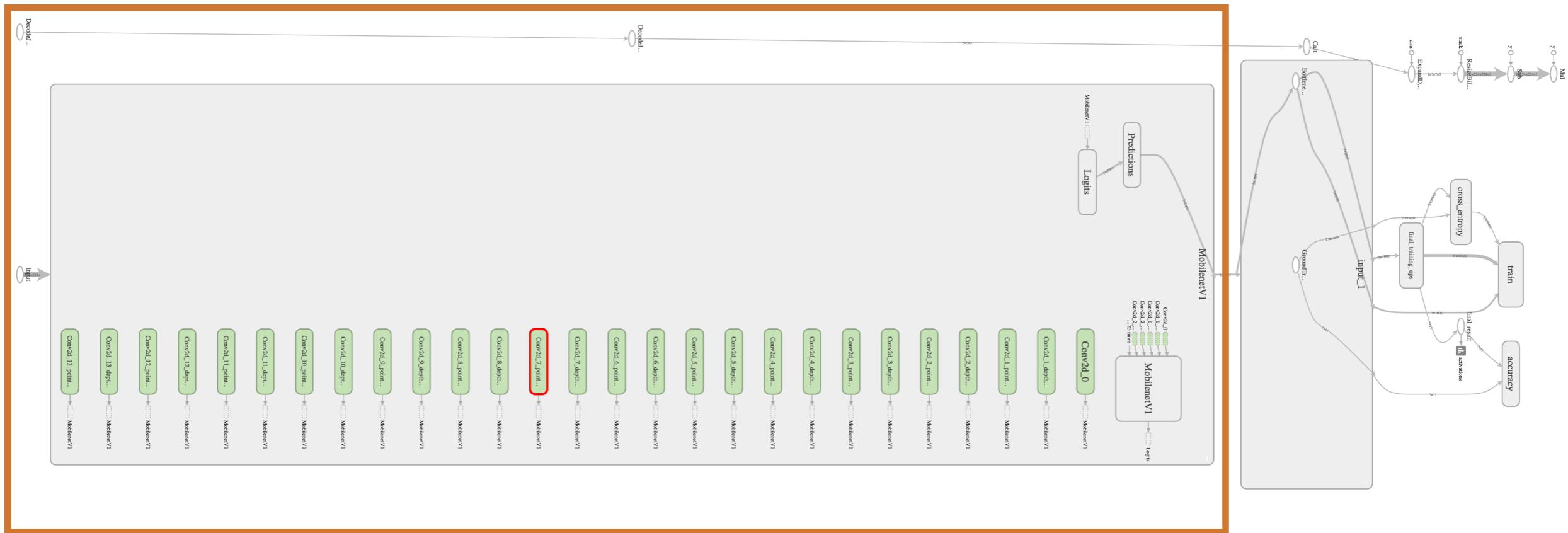


```
1 python -m scripts.retrain \
2   --bottleneck_dir=tf_files/bottlenecks \
3   --how_many_training_steps=500 \
4   --model_dir=tf_files/models/ \
5   --summaries_dir=tf_files/training_summaries/mobilenet_0.50_224 \
6   --output_graph=tf_files/retrained_graph.pb \
7   --output_labels=tf_files/retrained_labels.txt \
8   --architecture="mobilenet_0.50_224" \
9   --image_dir=tf_files/fruit_photos
```

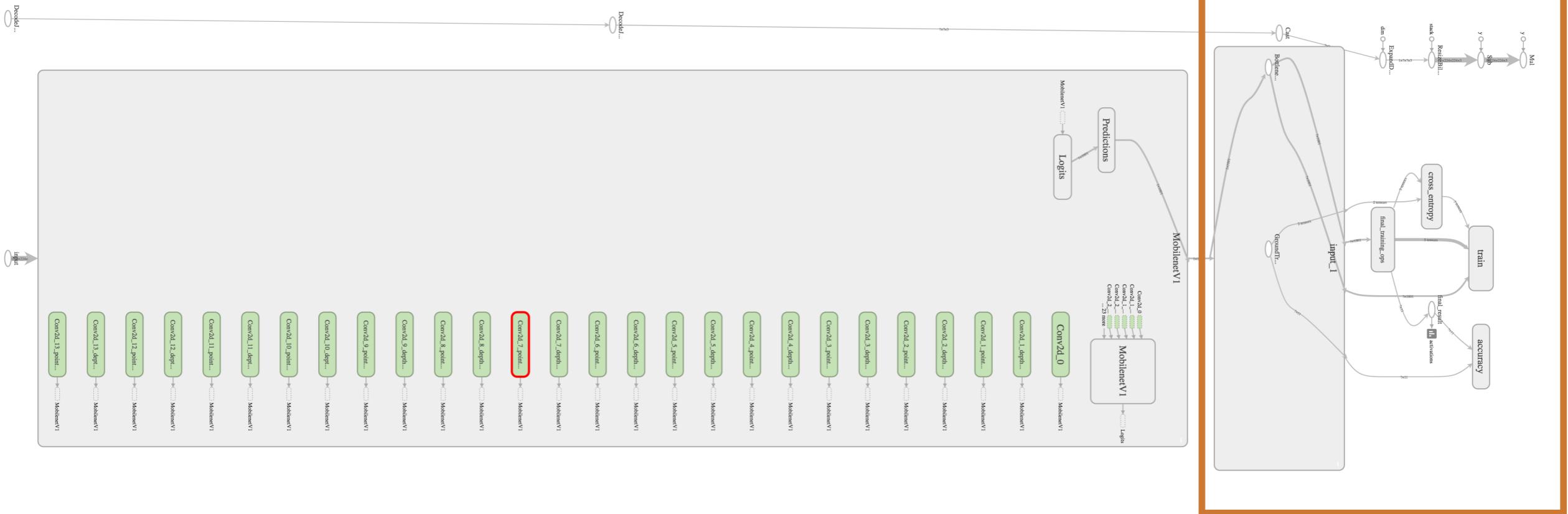
HOW TRAINING WORKS?



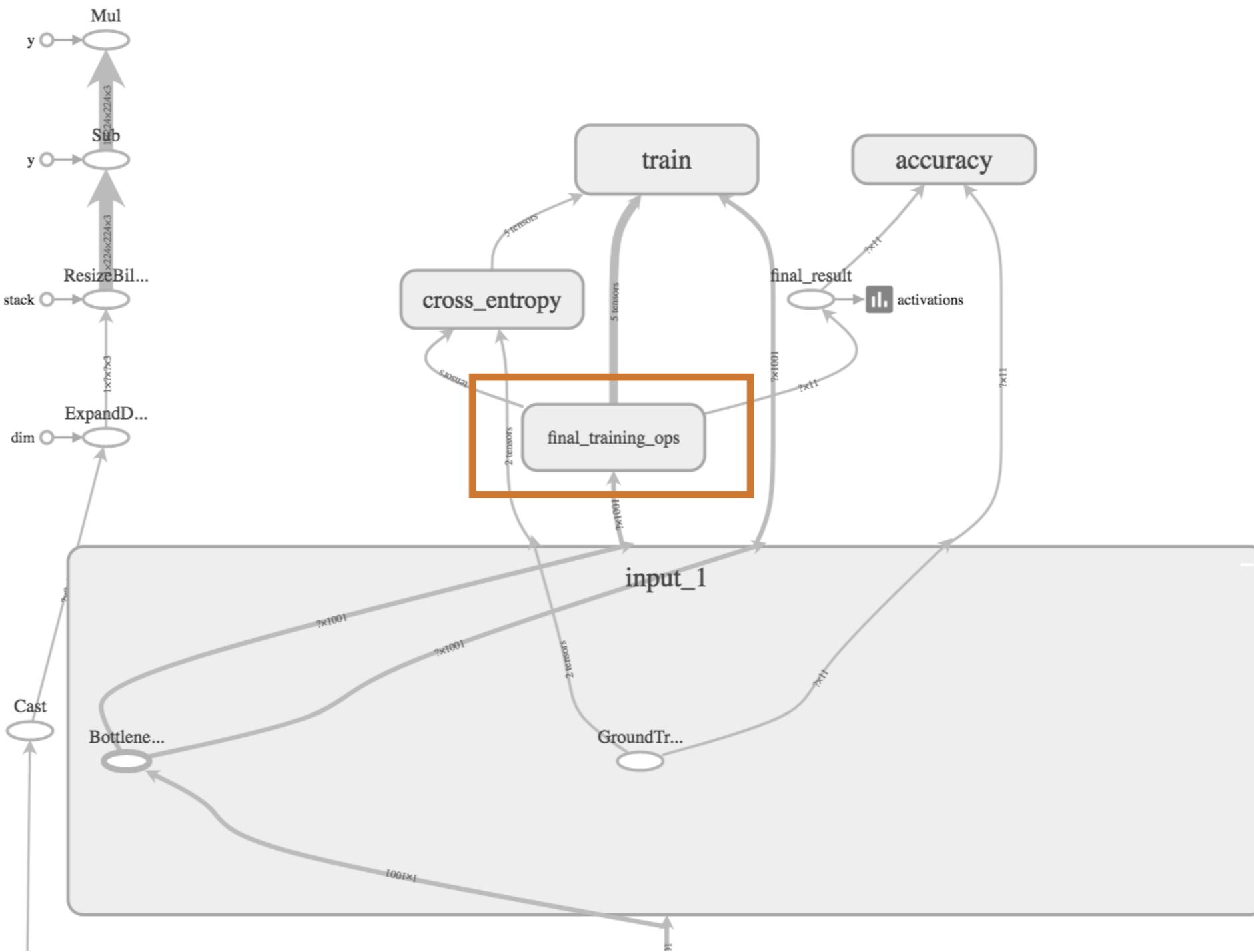
HOW TRAINING WORKS?



HOW TRAINING WORKS?

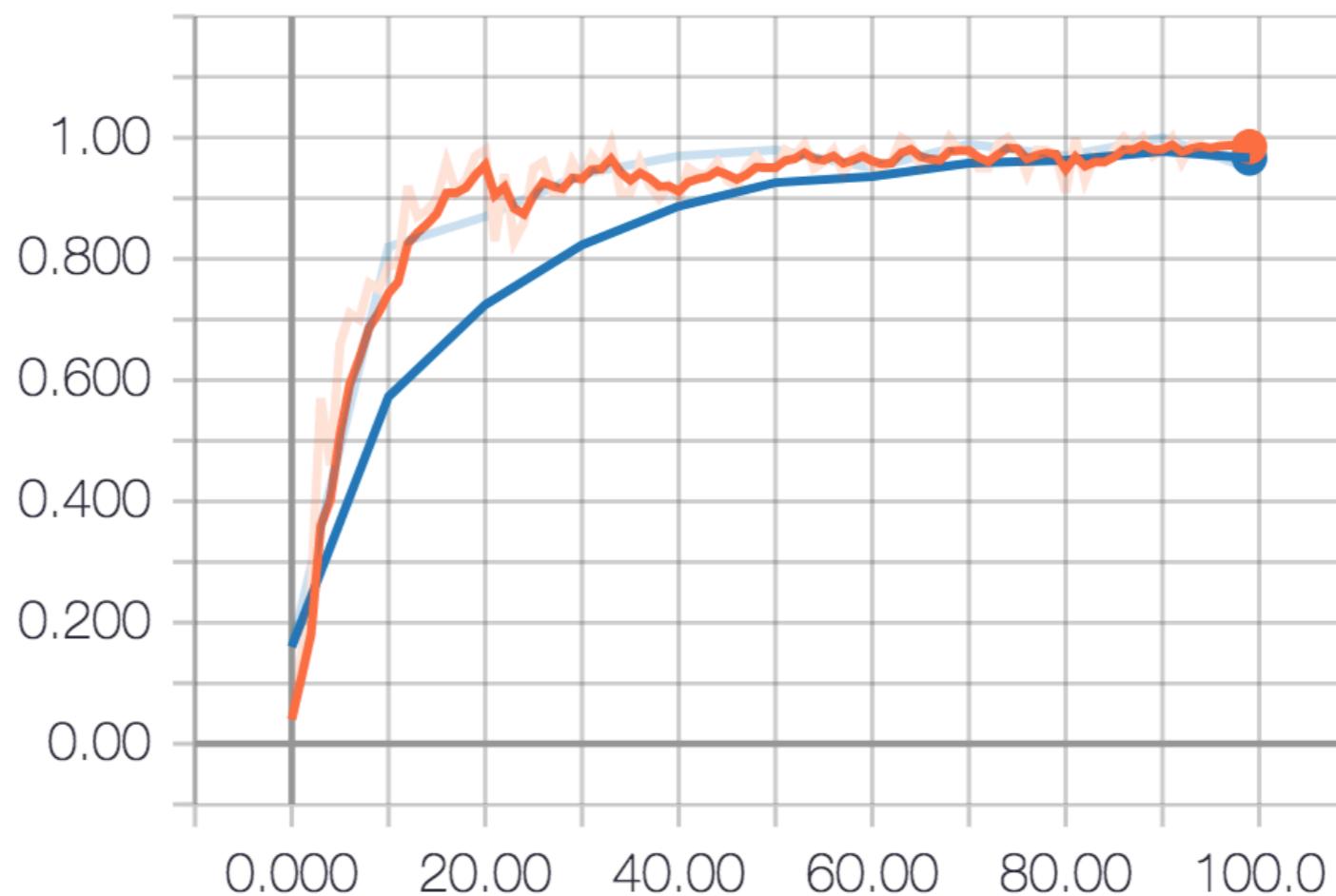


HOW TRAINING WORKS?



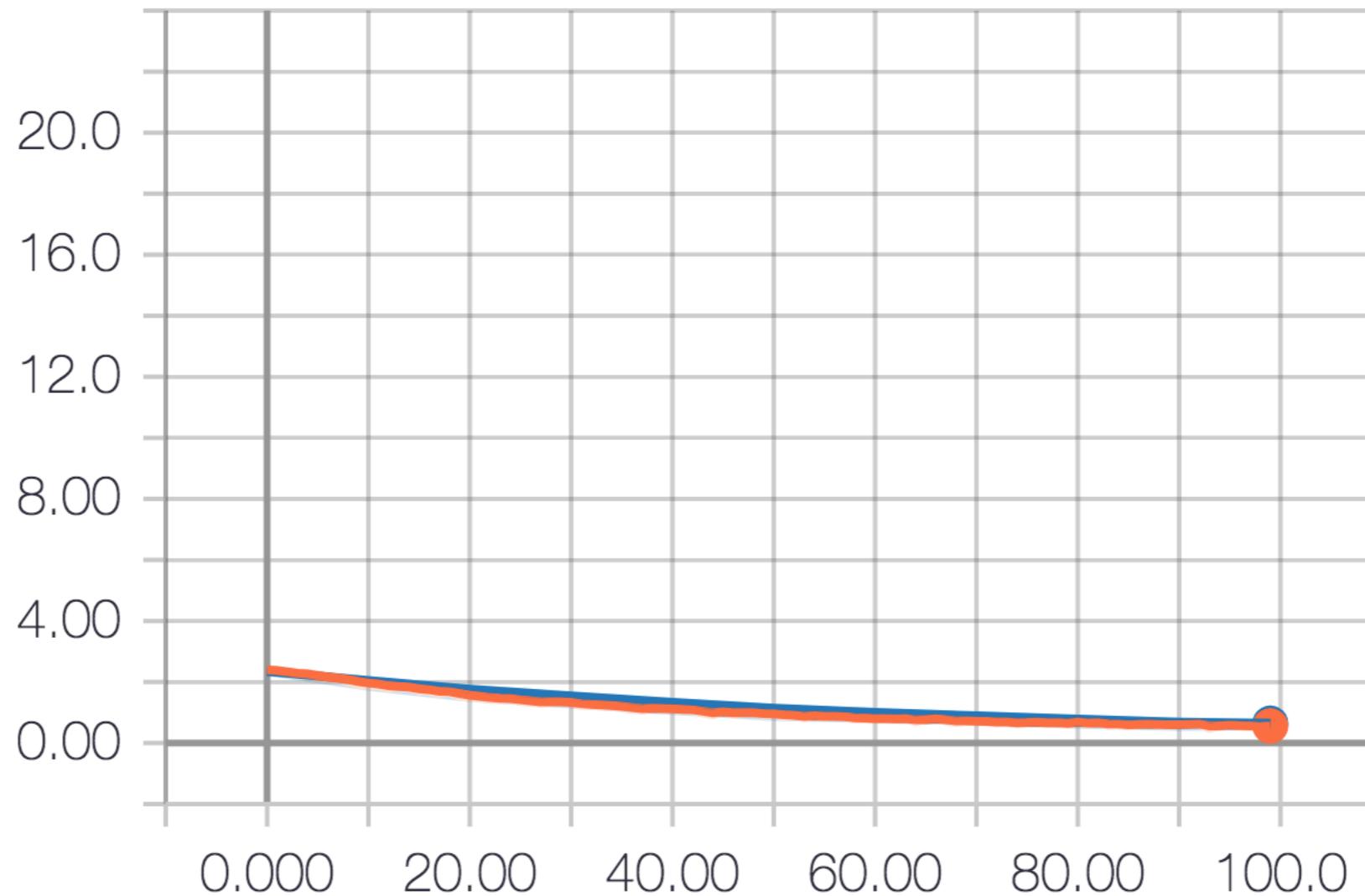
HOW TRAINING WORKS?

accuracy_1



HOW TRAINING WORKS?

`cross_entropy_1`



MODEL ACCURACY



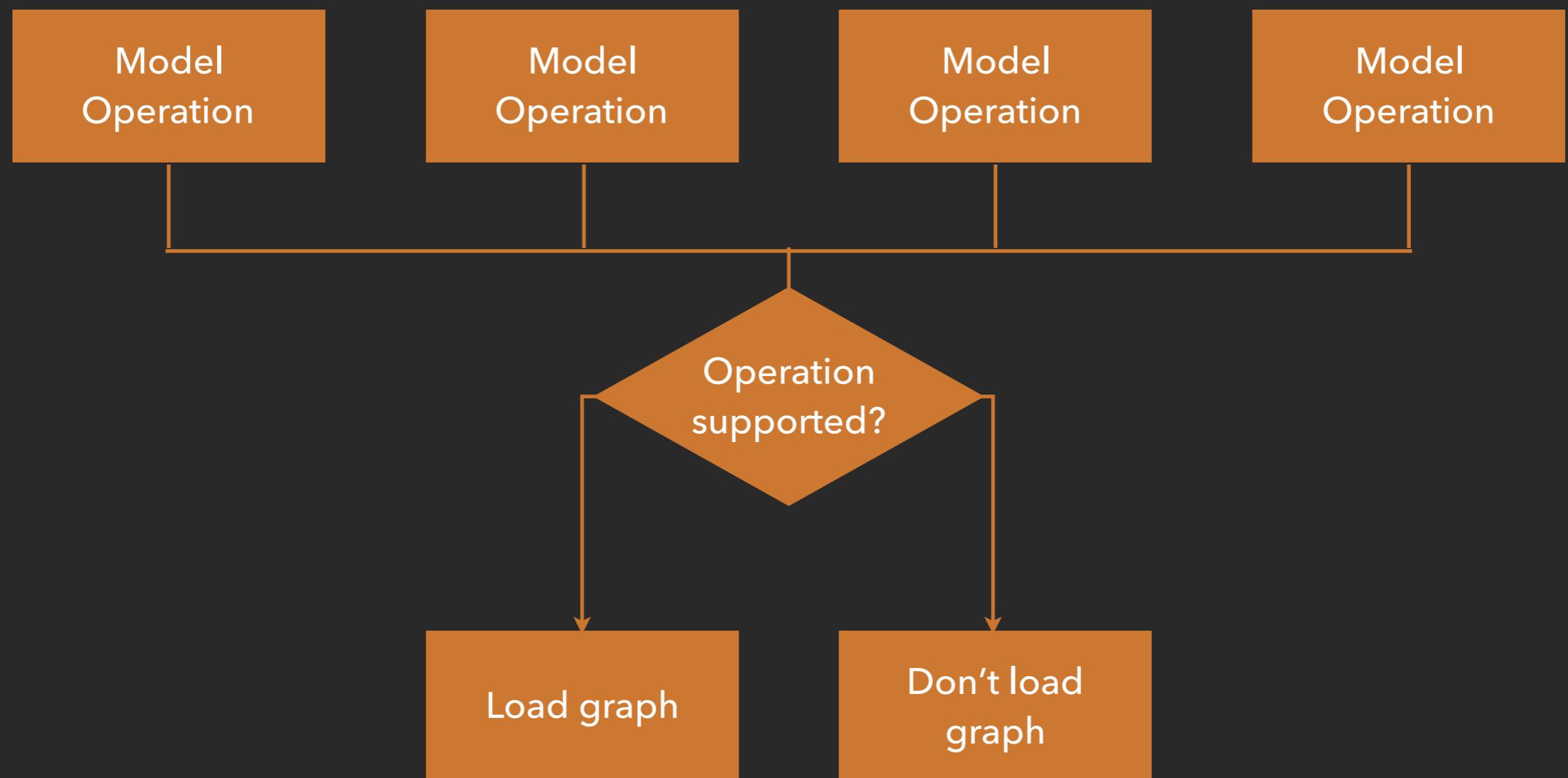
```
1 python -m scripts.label_image  
2     --graph=tf_files/retrained_graph.pb  
3     --image=tf_files/banana.jpg
```

MODEL ACCURACY



```
1 Evaluation time (1-image): 0.344s
2
3 banana 0.940712
4 lemon 0.0558581
5 peach 0.003358
6 apple granny smith 7.17525e-05
7 apple braeburn 6.50068e-08
```

OPTIMISING THE MODEL



OPTIMISING THE MODEL



```
1 python -m tensorflow.python.tools.optimize_for_inference \
2   --input=tf_files/retrained_graph.pb \
3   --output=tf_files/optimized_graph.pb \
4   --input_names="input" \
5   --output_names="final_result"
```

OPTIMISING THE MODEL



```
1 python -m scripts.label_image \  
2       --image=tf_files/banana.jpg \  
3       --graph=tf_files/optimized_graph.pb
```

OPTIMISING THE MODEL



```
1 Evaluation time (1-image): 0.236s
2
3 banana 0.999999
4 apple braeburn 6.97637e-07
5 peach 1.57967e-07
6 lemon 1.45191e-07
7 cherry 1.42835e-08
```

OPTIMISING THE MODEL

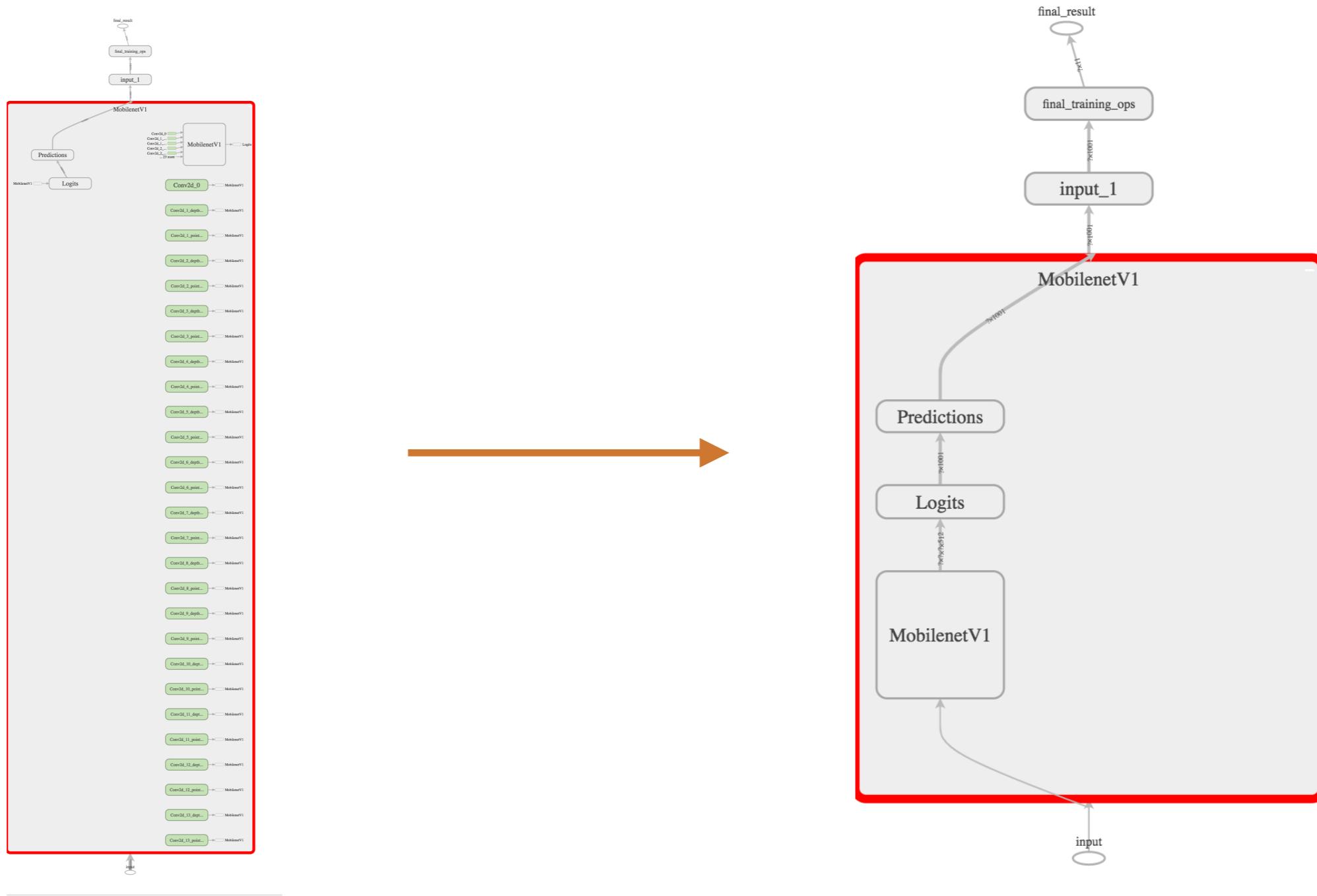


```
1 Evaluation time (1-image): 0.344s  
2  
3 banana 0.999999  
4 apple braeburn 6.97637e-07  
5 peach 1.57967e-07  
6 lemon 1.45191e-07  
7 cherry 1.42835e-08
```



```
1 Evaluation time (1-image): 0.236s  
2  
3 banana 0.999999  
4 apple braeburn 6.97637e-07  
5 peach 1.57967e-07  
6 lemon 1.45191e-07  
7 cherry 1.42835e-08
```

OPTIMISING THE MODEL



QUANTISATION



```
1 gzip -c tf_files/optimized_graph.pb > tf_files/optimized_graph.pb.gz
2 gzip -l tf_files/optimized_graph.pb.gz
3
4 compressed    uncompressed   ratio    uncompressed_name
5          5051787           5484061   7.8%  tf_files/optimized_graph.pb
```

QUANTISATION



```
1 python -m scripts.quantize_graph \
2   --input=tf_files/optimized_graph.pb \
3   --output=tf_files/rounded_graph.pb \
4   --output_node_names=final_result \
5   --mode=weights_rounded
```

QUANTISATION



```
1 gzip -c tf_files/optimized_graph.pb > tf_files/optimized_graph.pb.gz
2 gzip -l tf_files/optimized_graph.pb.gz
3
4 compressed uncompressed ratio uncompressed_name
5 1638526 5484080 70.1% tf_files/rounded_graph.pb
```

QUANTISATION



```
1 python -m scripts.label_image \  
2   --image=tf_files/fruit_photos/Banana/0_100.jpg \  
3   --graph=tf_files/rounded_graph.pb  
4  
5 Evaluation time (1-image): 0.234s  
6  
7 banana 0.999951  
8 lemon 3.05915e-05  
9 apple braeburn 1.38778e-05  
10 apple granny smith 1.62907e-06  
11 peach 1.21448e-06
```

OPTIMISING THE MODEL

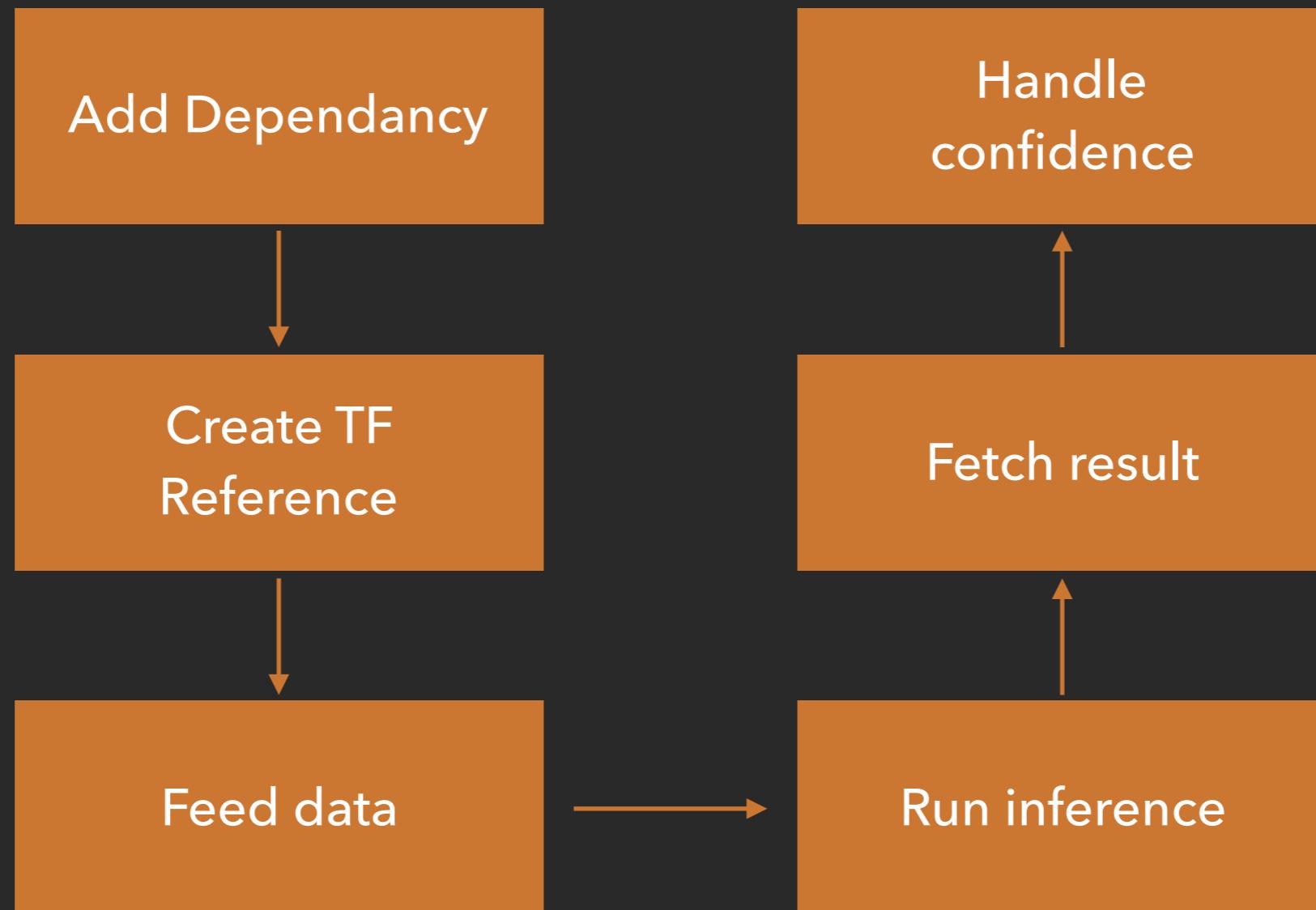


```
1 Evaluation time (1-image): 0.236s  
2  
3 banana 0.999999  
4 apple braeburn 6.97637e-07  
5 peach 1.57967e-07  
6 lemon 1.45191e-07  
7 cherry 1.42835e-08
```



```
1 Evaluation time (1-image): 0.234s  
2  
3 banana 0.999951  
4 lemon 3.05915e-05  
5 apple braeburn 1.38778e-05  
6 apple granny smith 1.62907e-06  
7 peach 1.21448e-06
```

ADDING THIS TO AN APP



ADDING THIS TO AN APP



```
1 val inferenceInterface =  
2     TensorFlowInferenceInterface(assetManager, modelName)
```

ADDING THIS TO AN APP



```
1 bitmap.getPixels(intValues, 0, bitmap.width, 0, 0, bitmap.width, bitmap.height)  
2  
3  
4     // convert to 3d array (width / height / color)  
5  
6  
7
```

ADDING THIS TO AN APP



```
1 bitmap.getPixels(intValues, 0, bitmap.width, 0, 0, bitmap.width, bitmap.height)
2 for (i in intValues.indices) {
3     val values = intValues[i]
4     floatValues[i * 3] = ((values shr 16 and 0xFF) - imageMean) / imageStd
5     floatValues[i * 3 + 1] = ((values shr 8 and 0xFF) - imageMean) / imageStd
6     floatValues[i * 3 + 2] = ((values and 0xFF) - imageMean) / imageStd
7 }
```

ADDING THIS TO AN APP



```
1 val inputName = "input"
2 inferenceInterface.feed(inputName, floatValues, 1, inputSize, inputSize, 3);
```

ADDING THIS TO AN APP



```
1 val inputName = "input"  
2 inferenceInterface.feed(inputName, floatValues, [1, inputSize, inputSize, 3]);
```



Shape of our input

ADDING THIS TO AN APP



```
1 val outputNames = arrayOf("MobilenetV1/Predictions/Softmax")
2
3 inferenceInterface.run(outputNames, false)
```

ADDING THIS TO AN APP



```
1 val outputName = "MobilenetV1/Predictions/Softmax"
2 val operation = inferenceInterface.graphOperation(outputName)
3 val numClasses = operation.output[0].shape.size[1]
4 val outputs = FloatArray(numClasses)
5
6 inferenceInterface.fetch(outputName, outputs)
```

ADDING THIS TO AN APP



```
1 for (i in outputs.indices) {  
2     if (outputs[i] > THRESHOLD) {  
3         val label = labels.get(i)  
4         val confidence = outputs[i]  
5     }  
6 }
```

ADDING THIS TO AN APP



<https://github.com/tensorflow/tensorflow>

CONCLUSION