TensorFlow Lite

The professional course

TensorFlow Lite

Week 3

Agenda

1. Introduction to model optimizations
2. Optimization techniques
   a. Post-training quantization
   b. Weight clustering
   c. Model pruning

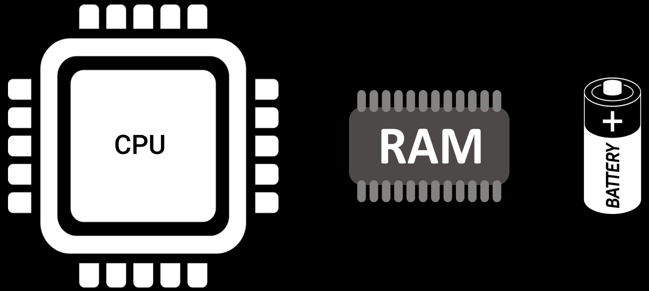
Introduction to model optimizations

Introduction to model optimizations

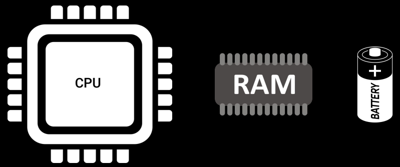
When we are talking about embedded applications, we have to remember that:

- The device that will run the application may have hardware limitations

PCs and Servers



Embedded systems

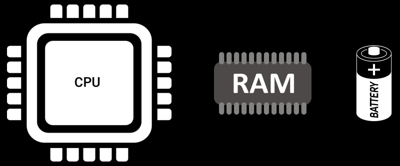


Introduction to model optimizations

When we are talking about embedded applications, we have to remember that:

- It’s important to improve resources usages, such as CPU, memory, and battery
- Some optimizations can be applied to our model that will run within hardware constraints

Embedded systems



Introduction to model optimizations

Some optimization benefits:

- **Model size reduction**
  - Smaller storage size and less memory usage
- **Latency reduction**
  - Faster inference time

Introduction to model optimizations

Some optimization harms:

- **Accuracy reduction**
    - These optimizations can lead to a slight decrease in model accuracy

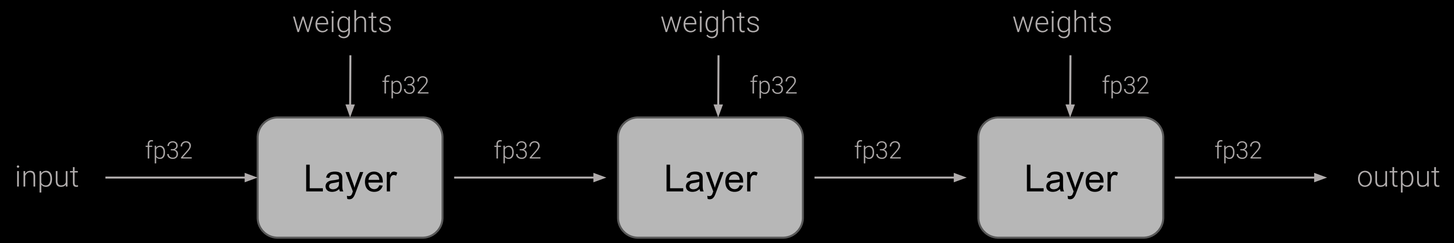
Optimization Techniques

Quantization

Optimization techniques

Post-training quantization

- Usually, the parameters (weights and inputs/outputs) of a neural network are represented by:



Optimization techniques

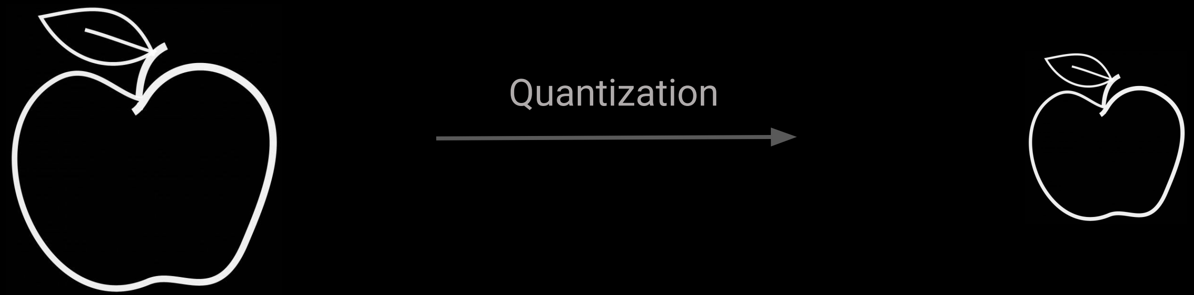
Post-training quantization

- But we can change these parameters bitwith:

Parameters in fp32

Quantization

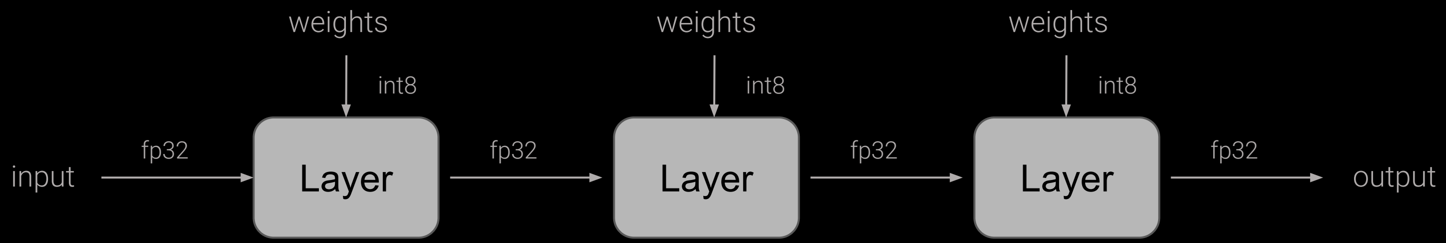
Parameters in int8 and fp16



Optimization techniques

Post-training quantization

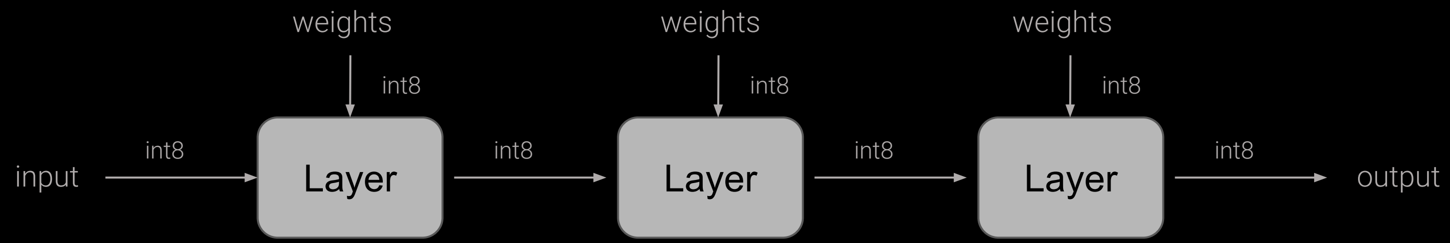
- Types:
  - **Weight quantization**: quantize just the weights



Optimization techniques

Post-training quantization

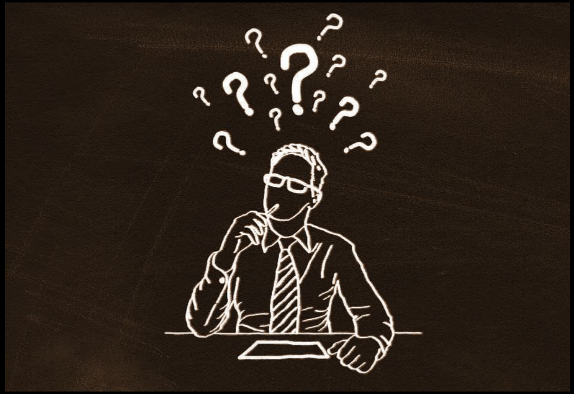
- Types:
  - **Full quantization**: quantize both the weights and activations



Optimization techniques

Post-training quantization

- **What is the best quantization type for me**?



Optimization techniques

Post-training quantization

- **What is the best quantization type for me**?
  - Consider the device specification
  - Consider your project constraints

Optimization techniques

Post-training quantization

- **What is the best quantization type for me?** Is the one that keeps satisfactory values to **size**, **accuracy**, and **latency** of the model, depends on your project requirements.

Optimization techniques

Post-training quantization

- **How is quantization done in practice?**

$$X_{\text{quantized}} = X_{\text{real}}/\mathit{scale} + X_{\text{zero\_point}}$$

Optimization techniques

Post-training quantization

1. Compute *scale* and the X_{zero_point} by finding min and max value of weight tensor:

$$X_{real} \in [X_{real\_min}, X_{real\_max}]$$

$$scale = (X_{real\_max} - X_{real\_min}) / (X_{quantized\_max} - X_{quantized\_min})$$

$$X_{zero\_point} = X_{quantized\_max} - X_{real\_max} / scale$$

Optimization techniques

Post-training quantization

Ex: X_{real} = 0.85 in FP32 ∈ [-1 , 1] -> $X_{quantized}$ in INT8 ∈ [0 , 255]

$$scale = (X_{real\_max} - X_{real\_min}) / (X_{quantized\_max} - X_{quantized\_min}) = (1 -(- 1)) / (255 - 0) = 2/255$$

$$X_{zero\_point} = X_{quantized\_max} - X_{real\_max}/scale = 255 - 1/(2/255) \approx 127$$

$$X_{quantized} = X_{real}/scale + X_{zero\_point} = 0.85/(2/255) + 127 \approx 235$$

Clustering

Optimization techniques

Weight clustering

- Weight clustering is a technique to reduce the storage and transfer size of your model by replacing many unique parameter values with a smaller number of unique values.

Optimization techniques

Weight clustering

- Layer weight matrix

2.21	0.86	-0.53	-1.25
-1.75	0.96	0.23	-1.11
-0.35	-2.89	2.51	-1.86
-1.52	2.71	1.69	0.56

Optimization techniques

Weight clustering

- Get centroids:

2.21	0.86	-0.53	-1.25
-1.75	0.96	0.23	-1.11
-0.35	-2.89	2.51	-1.86
-1.52	2.71	1.69	0.56

x1
x2
x3
x4

Optimization techniques

Weight clustering

- Get centroids indexes:

2.21	0.86	-0.53	-1.25
-1.75	0.96	0.23	-1.11
-0.35	-2.89	2.51	-1.86
-1.52	2.71	1.69	0.56

Centroids	Index
x1	0
x2	1
x3	2
x4	3

Optimization techniques

Weight clustering

- Assign Indexes:

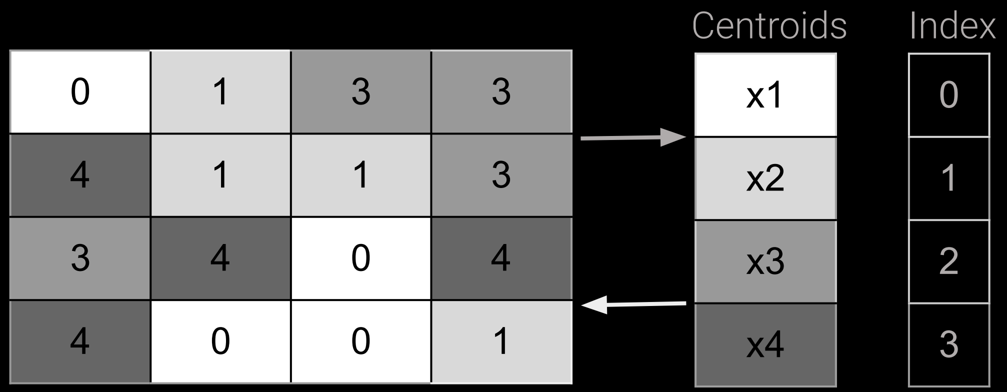
2.21	0.86	-0.53	-1.25
-1.75	0.96	0.23	-1.11
-0.35	-2.89	2.51	-1.86
-1.52	2.71	1.69	0.56

Centroids	Index
x1	0
x2	1
x3	2
x4	3

Optimization techniques

Weight clustering

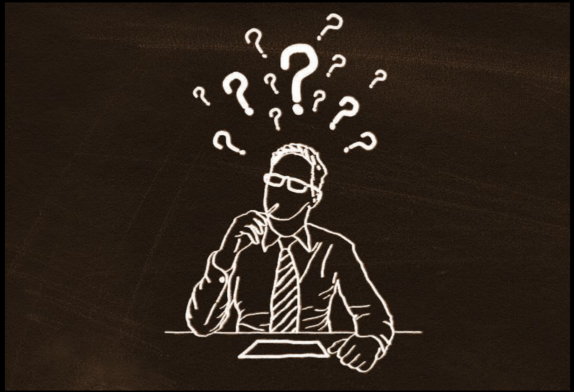
- Pull Indexes:



Optimization techniques

Weight clustering

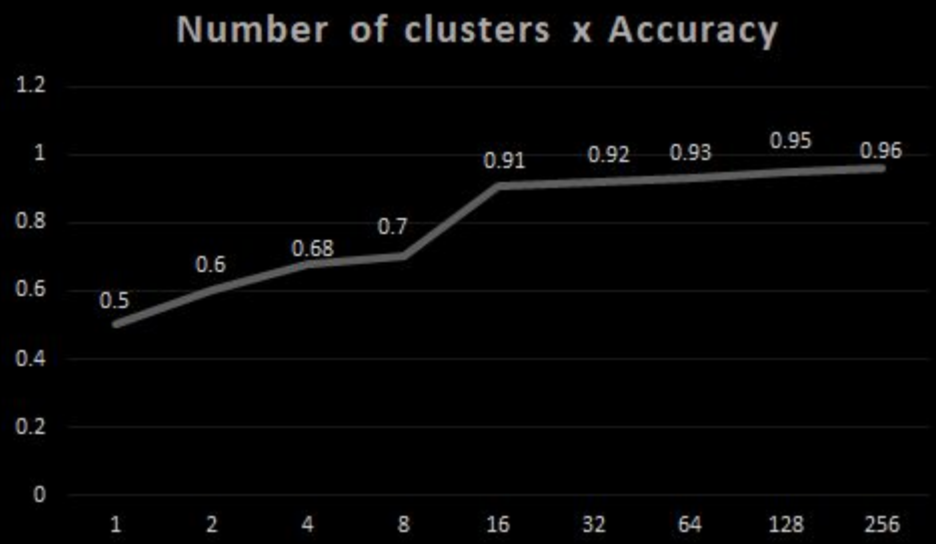
- **What is the best number of centroids (or clusters)**?



Optimization techniques

Weight clustering

- **Elbow method**



Pruning

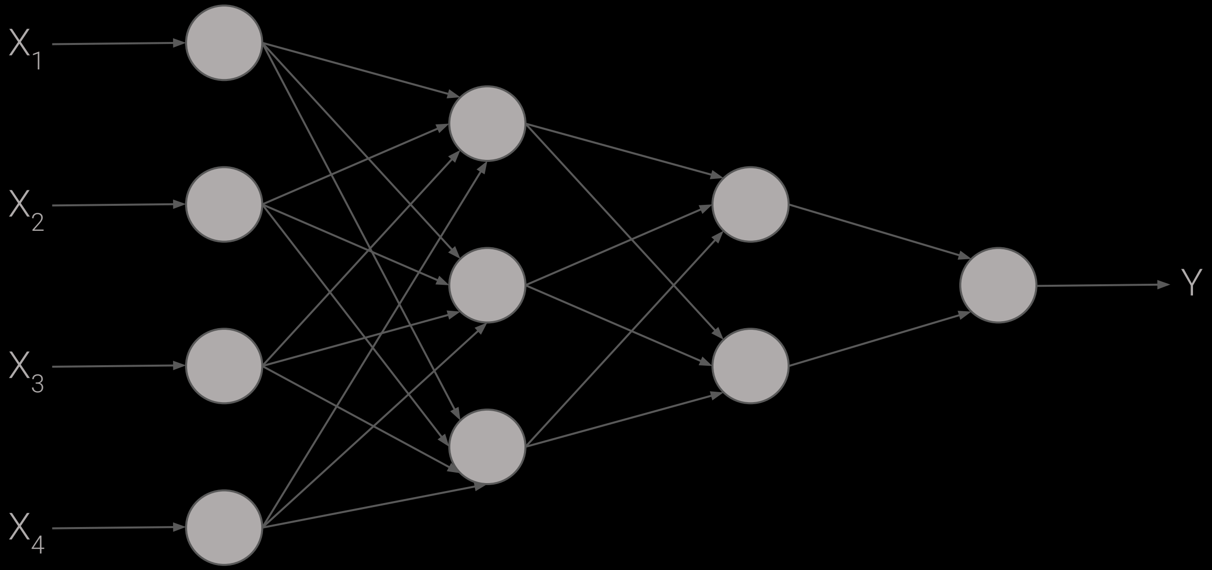
Optimization techniques

Model pruning

- Training optimization
- The goal of pruning is to reduce the number of parameters and operations in our neural network
- Sparse models are easier to compress, and we can skip the zeroes during inference for latency improvements

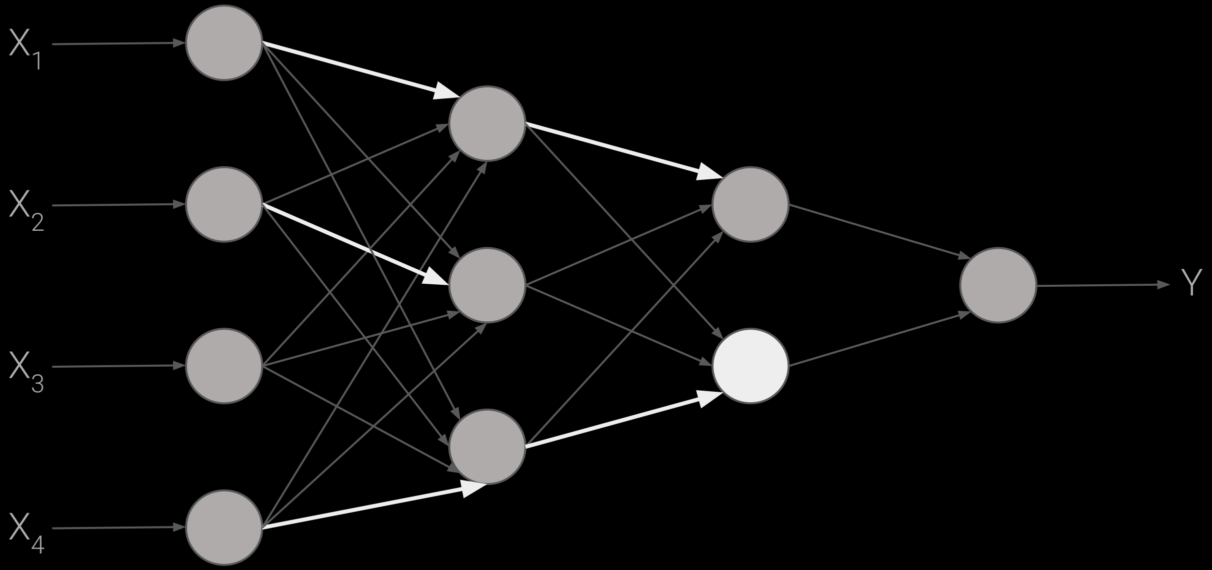
Optimization techniques

Model pruning



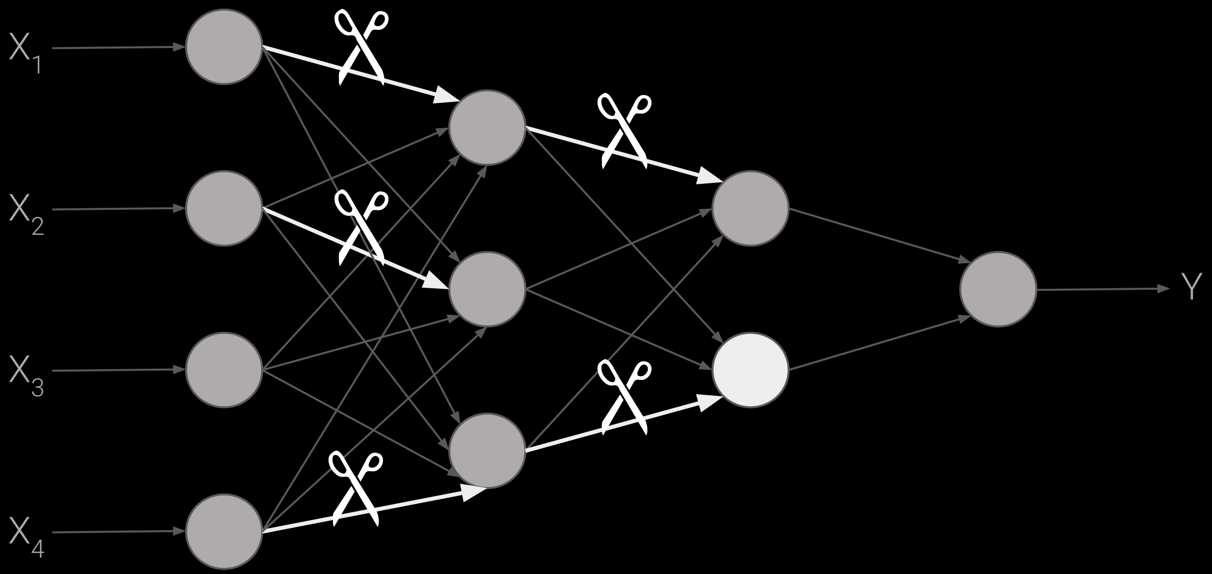
Optimization techniques

Model pruning



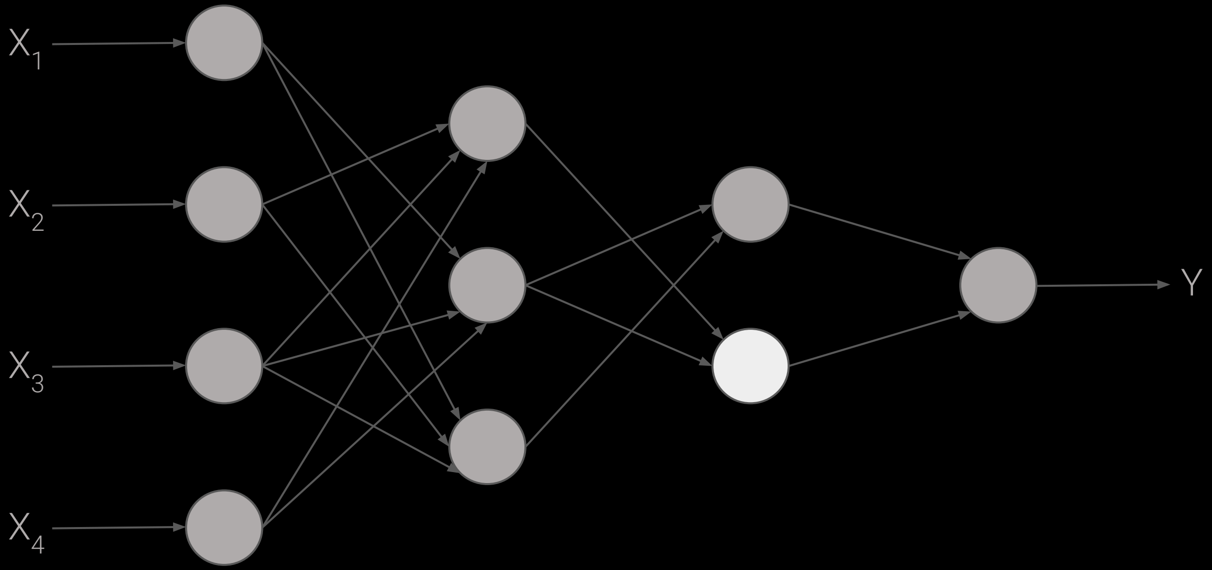
Optimization techniques

Model pruning



Optimization techniques

Model pruning



Optimization techniques

Weight clustering

- The pruning is done by adding zeros to some parameters
- You can specify the sparsity you want
  - E.g: 50%

0.02	0.86	-0.53	-0.01
-1.75	0.10	0.23	-0.03
-0.35	-2.89	0.15	-1.86
-1.52	0.17	1.69	-0.22

→

0	0.86	-0.53	0
-1.75	0	0.23	0
0	-2.89	0	-1.86
-1.52	0	1.69	0

Hands-On

Hands-on Project

Step:

1. Let’s apply all these optimizations to our model using the TensorFlow Lite !

Wrap-up

Wrap-Up

During this week we have learned:

1. An overview of the model optimizations
2. Importance to apply different optimization techniques
3. How to apply in practice these optimization techniques

References

To learn more, please, take a look:

- Model optimization: https://www.tensorflow.org/lite/performance/model_optimization
- Post-training quantization: https://www.tensorflow.org/lite/performance/post_training_quantization
- Weight clustering: https://blog.tensorflow.org/2020/08/tensorflow-model-optimization-toolkit-weight-clustering-api.html