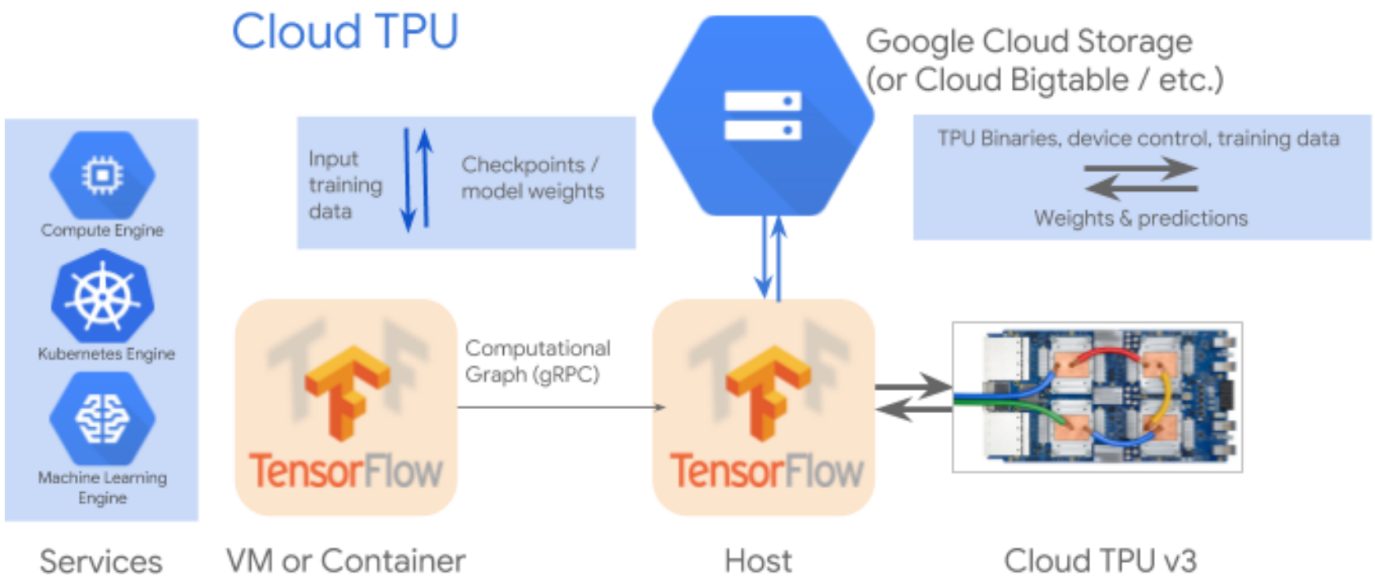
1. Configuration

1.1 Define some constants

```
GCS_PATH = 'gs://tf-everywhere-col'
DATASET_NAME = 'rock_paper_scissors'
BATCH_SIZE = 128
NUM_CLASSES = 3
DENSE_UNITS = 128
DROPOUT = 0.4
```

1.2 Authenticate

Authenticates the Colab machine and the TPU using your credentials so you can access the private GCS bucket.




```
from google.colab import auth

auth.authenticate_user()
```

1.3 Setup TPU

```
!pip install tensorflow_addons
```

```
import tensorflow as tf
import tensorflow_addons as tfa
import numpy as np
import matplotlib.pyplot as plt
import tensorflow_datasets as tfds
from tensorflow.keras import Model, layers, callbacks
from datetime import datetime
```

```
Collecting tensorflow_addons
  Downloading https://files.pythonhosted.org/packages/74/e3/56d2fe76f0bb7c88ed9b2a6a557e
     |████████████████████████████████| 706kB 6.2MB/s
Requirement already satisfied: typeguard>=2.7 in /usr/local/lib/python3.7/dist-packages
Installing collected packages: tensorflow-addons
Successfully installed tensorflow-addons-0.12.1
```

```
tpu = tf.distribute.cluster_resolver.TPUClusterResolver()
tf.config.experimental_connect_to_cluster(tpu)
tf.tpu.experimental.initialize_tpu_system(tpu)

strategy = tf.distribute.experimental.TPUStrategy(tpu)
```

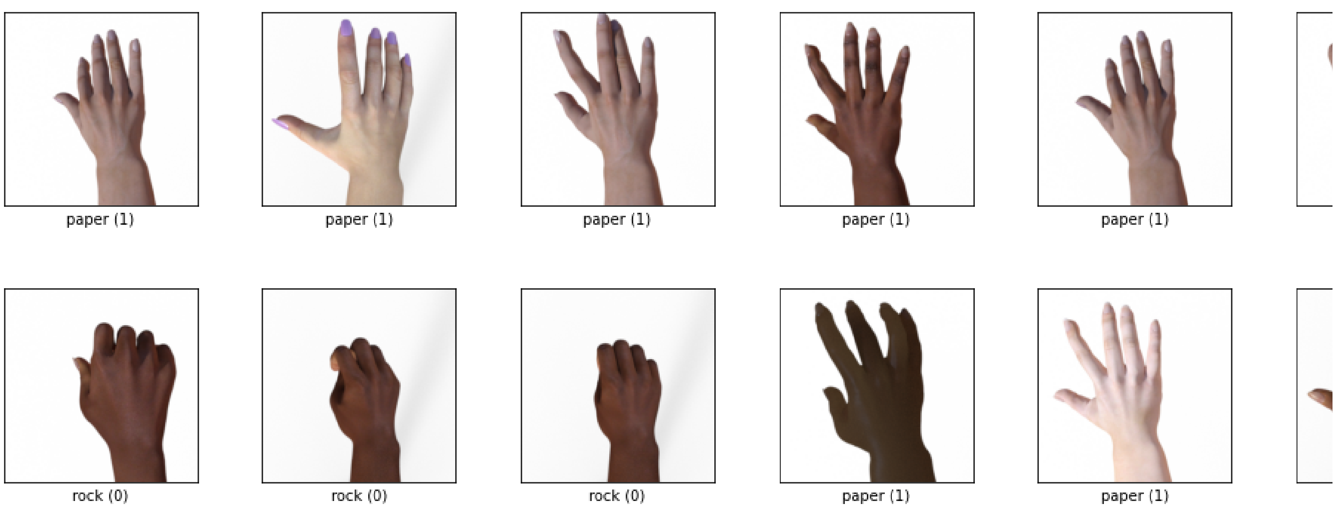
```
INFO:tensorflow:Initializing the TPU system: grpc://10.75.133.202:8470
INFO:tensorflow:Initializing the TPU system: grpc://10.75.133.202:8470
INFO:tensorflow:Clearing out eager caches
INFO:tensorflow:Clearing out eager caches
INFO:tensorflow:Finished initializing TPU system.
INFO:tensorflow:Finished initializing TPU system.
WARNING:absl:`tf.distribute.experimental.TPUStrategy` is deprecated, please use  the nor
INFO:tensorflow:Found TPU system:
INFO:tensorflow:Found TPU system:
INFO:tensorflow:*** Num TPU Cores: 8
INFO:tensorflow:*** Num TPU Cores: 8
INFO:tensorflow:*** Num TPU Workers: 1
INFO:tensorflow:*** Num TPU Workers: 1
INFO:tensorflow:*** Num TPU Cores Per Worker: 8
INFO:tensorflow:*** Num TPU Cores Per Worker: 8
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:localhost/replica:0/task:0/
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:localhost/replica:0/task:0/
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
```

```
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
INFO:tensorflow:*** Available Device: _DeviceAttributes(/job:worker/replica:0/task:0/dev
```

2. Dataset definition

```
train_ds, train_ds_info = tfds.load(DATASET_NAME,
                                    split="train",
                                    with_info=True,
                                    shuffle_files=True,
                                    data_dir=f'{GCS_PATH}/{DATASET_NAME}/train')
val_ds, val_ds_info = tfds.load(DATASET_NAME,
                                split="test",
                                with_info=True,
                                shuffle_files=False,
                                data_dir=f'{GCS_PATH}/{DATASET_NAME}/val')
```

```
tfds.visualization.show_examples(train_ds, train_ds_info, rows=2, cols=6);
```



2.1 Data Preprocessing

```
from tensorflow.data.experimental import AUTOTUNE
```

```
from tensorflow.data.experimental import AUTOTUNE

@tf.function
def preprocess(example, height=224, width=224):
    image = example['image']
    label = example['label']
    image = tf.image.resize_with_crop_or_pad(image, height, width)

    image = tf.keras.applications.resnet.preprocess_input(image)
    label = tf.one_hot(label, depth=NUM_CLASSES)
    return image,label

def get_datasets(train_ds, val_ds, batch_size):
    norm_train_ds = train_ds.map(preprocess, num_parallel_calls=AUTOTUNE)\
            .shuffle(500).batch(batch_size).prefetch(AUTOTUNE)
    norm_val_ds = val_ds.map(preprocess, num_parallel_calls=AUTOTUNE)\
            .batch(batch_size).cache()
    return norm_train_ds, norm_val_ds
```

2.3 Create train/val datasets

Note the number of replicas is **8** then we need to multiply the original batch size with the number of replicas, to optimize the usage or the TPU.

```
norm_train_ds, norm_val_ds = get_datasets(train_ds, val_ds,
                              batch_size=strategy.num_replicas_in_sync*BATCH_SIZE)

strategy.num_replicas_in_sync
```

```
8
```

3. Model definition

```
def create_model(num_classes, units, dropout):

  input_layer = layers.Input(shape=(224,224,3))

  feature_extractor = tf.keras.applications.ResNet50(include_top=False, weights='imagenet', i
  feature_extractor.trainable = False

  features = layers.GlobalAveragePooling2D()(feature_extractor.output)
  x = layers.Dense(units, 'relu')(features)
  x = layers.Dropout(dropout)(x)
  x = layers.Dense(num_classes, 'softmax', dtype='float32')(x)

  model = Model(inputs=[input_layer], outputs=[x], name='feature_extractor')
  model.summary()
```

```
    model.summary()
    return model
```

3.1 Create & Compile

Using the TPU distribution strategy create and compile the model

```
with strategy.scope():
    model = create_model(NUM_CLASSES, DENSE_UNITS, DROPOUT)
    model.compile(optimizer='adam',
                  loss=tf.keras.losses.CategoricalCrossentropy(label_smoothing=0.2),
                  metrics=[tf.keras.metrics.CategoricalAccuracy(),
                           tfa.metrics.F1Score(num_classes=NUM_CLASSES, average='macro')])
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/r
94773248/94765736 [==============================] - 1s 0us/step
Model: "feature_extractor"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            [(None, 224, 224, 3) 0
__________________________________________________________________________________________________
conv1_pad (ZeroPadding2D)       (None, 230, 230, 3)  0           input_1[0][0]
__________________________________________________________________________________________________
conv1_conv (Conv2D)             (None, 112, 112, 64) 9472        conv1_pad[0][0]
__________________________________________________________________________________________________
conv1_bn (BatchNormalization)   (None, 112, 112, 64) 256         conv1_conv[0][0]
__________________________________________________________________________________________________
conv1_relu (Activation)         (None, 112, 112, 64) 0           conv1_bn[0][0]
__________________________________________________________________________________________________
pool1_pad (ZeroPadding2D)       (None, 114, 114, 64) 0           conv1_relu[0][0]
__________________________________________________________________________________________________
pool1_pool (MaxPooling2D)       (None, 56, 56, 64)   0           pool1_pad[0][0]
__________________________________________________________________________________________________
conv2_block1_1_conv (Conv2D)    (None, 56, 56, 64)   4160        pool1_pool[0][0]
__________________________________________________________________________________________________
conv2_block1_1_bn (BatchNormali (None, 56, 56, 64)   256         conv2_block1_1_conv
__________________________________________________________________________________________________
conv2_block1_1_relu (Activation (None, 56, 56, 64)   0           conv2_block1_1_bn[0
__________________________________________________________________________________________________
conv2_block1_2_conv (Conv2D)    (None, 56, 56, 64)   36928       conv2_block1_1_relu
__________________________________________________________________________________________________
conv2_block1_2_bn (BatchNormali (None, 56, 56, 64)   256         conv2_block1_2_conv
__________________________________________________________________________________________________
conv2_block1_2_relu (Activation (None, 56, 56, 64)   0           conv2_block1_2_bn[0
__________________________________________________________________________________________________
conv2_block1_0_conv (Conv2D)    (None, 56, 56, 256)  16640       pool1_pool[0][0]
__________________________________________________________________________________________________
conv2_block1_3_conv (Conv2D)    (None, 56, 56, 256)  16640       conv2_block1_2_relu
__________________________________________________________________________________________________
conv2_block1_0_bn (BatchNormali (None, 56, 56, 256)  1024        conv2_block1_0_conv
__________________________________________________________________________________________________
conv2_block1_3_bn (BatchNormali (None, 56, 56, 256)  1024        conv2_block1_3_conv
__________________________________________________________________________________________________
```

```
conv2_block1_add (Add)          (None, 56, 56, 256)  0           conv2_block1_0_bn[0
                                                                 conv2_block1_3_bn[0
__________________________________________________________________________________
conv2_block1_out (Activation)   (None, 56, 56, 256)  0           conv2_block1_add[0]
__________________________________________________________________________________
conv2_block2_1_conv (Conv2D)    (None, 56, 56, 64)   16448       conv2_block1_out[0]
__________________________________________________________________________________
conv2_block2_1_bn (BatchNormali (None, 56, 56, 64)   256         conv2_block2_1_conv
__________________________________________________________________________________
conv2_block2_1_relu (Activation (None, 56, 56, 64)   0           conv2_block2_1_bn[0
__________________________________________________________________________________
conv2_block2_2_conv (Conv2D)    (None, 56, 56, 64)   36928       conv2_block2_1_relu
__________________________________________________________________________________
conv2_block2_2_bn (BatchNormali (None, 56, 56, 64)   256         conv2_block2_2_conv
__________________________________________________________________________________
conv2_block2_2_relu (Activation (None, 56, 56, 64)   0           conv2_block2_2_bn[0
__________________________________________________________________________________
conv2_block2_3_conv (Conv2D)    (None, 56, 56, 256)  16640       conv2_block2_2_relu
```

3.2 Train & Validate model

```
%%time
history = model.fit(norm_train_ds, validation_data=norm_val_ds, epochs=20)
```

```
Epoch 1/20
3/3 [==============================] - 25s 6s/step - loss: 2.0653 - categorical_accuracy
Epoch 2/20
3/3 [==============================] - 2s 705ms/step - loss: 1.0916 - categorical_accura
Epoch 3/20
3/3 [==============================] - 2s 693ms/step - loss: 0.8546 - categorical_accura
Epoch 4/20
3/3 [==============================] - 2s 670ms/step - loss: 0.7435 - categorical_accura
Epoch 5/20
3/3 [==============================] - 2s 683ms/step - loss: 0.6876 - categorical_accura
Epoch 6/20
3/3 [==============================] - 2s 653ms/step - loss: 0.6498 - categorical_accura
Epoch 7/20
3/3 [==============================] - 2s 849ms/step - loss: 0.6254 - categorical_accura
Epoch 8/20
3/3 [==============================] - 2s 662ms/step - loss: 0.6039 - categorical_accura
Epoch 9/20
3/3 [==============================] - 2s 708ms/step - loss: 0.5957 - categorical_accura
Epoch 10/20
3/3 [==============================] - 2s 676ms/step - loss: 0.5924 - categorical_accura
Epoch 11/20
3/3 [==============================] - 2s 684ms/step - loss: 0.5794 - categorical_accura
Epoch 12/20
3/3 [==============================] - 2s 672ms/step - loss: 0.5722 - categorical_accura
Epoch 13/20
3/3 [==============================] - 2s 847ms/step - loss: 0.5663 - categorical_accura
Epoch 14/20
3/3 [==============================] - 2s 666ms/step - loss: 0.5603 - categorical_accura
Epoch 15/20
```

```
3/3 [==============================] - 2s 676ms/step - loss: 0.5569 - categorical_accura
Epoch 16/20
3/3 [==============================] - 2s 649ms/step - loss: 0.5505 - categorical_accura
Epoch 17/20
3/3 [==============================] - 2s 650ms/step - loss: 0.5502 - categorical_accura
Epoch 18/20
3/3 [==============================] - 2s 649ms/step - loss: 0.5441 - categorical_accura
Epoch 19/20
3/3 [==============================] - 2s 827ms/step - loss: 0.5446 - categorical_accura
Epoch 20/20
3/3 [==============================] - 2s 653ms/step - loss: 0.5411 - categorical_accura
CPU times: user 35.7 s, sys: 3.01 s, total: 38.7 s
Wall time: 1min 9s
```