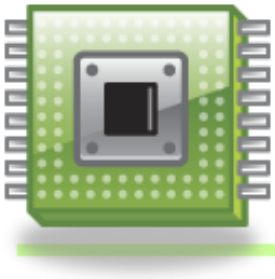


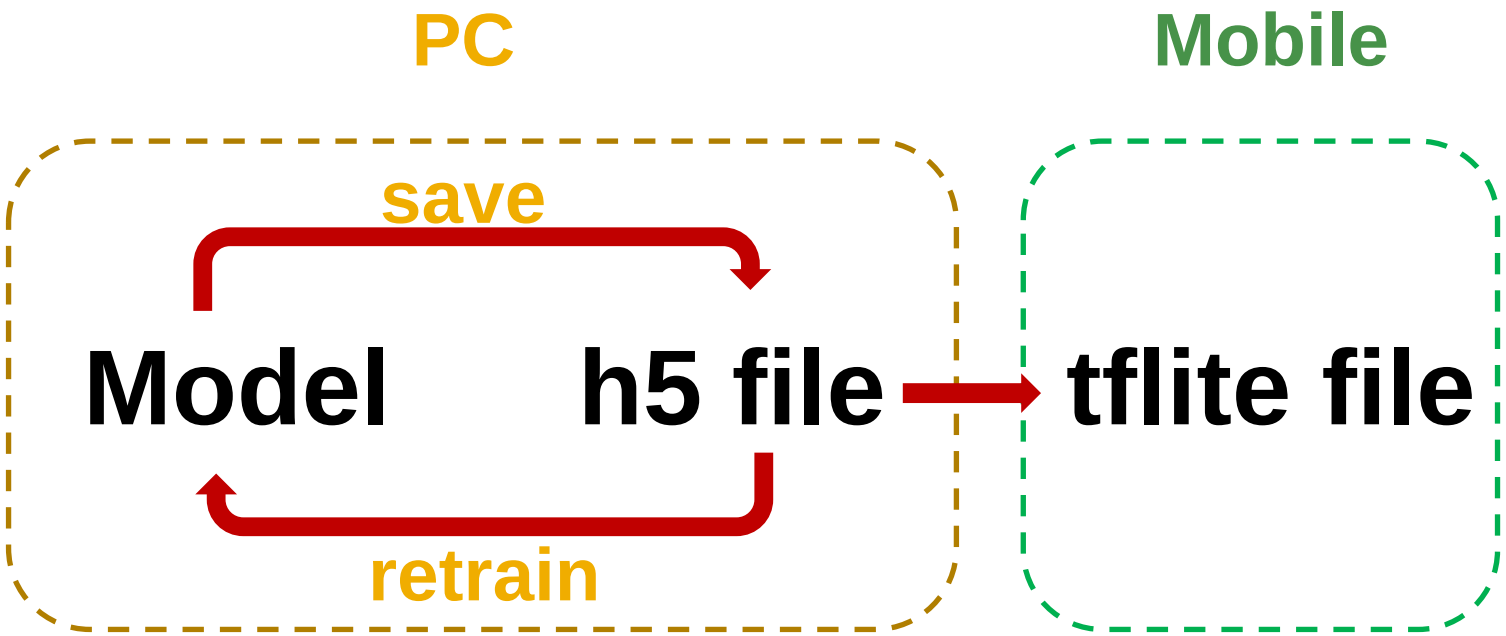
Model Transfer to TFLite

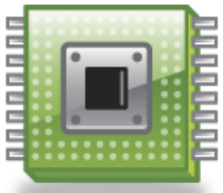
Speaker : Tse-Yu Chen

Advisor : Prof. Tong-Yu Hsieh

Convert to TFLite

- Use your PC to train and convert the model.





Environment Setting

1. conda create --name tensorflow2.0 python=3.7
2. conda activate tensoflow2.0
3. conda install tensorflow=2.1
4. conda install opencv
5. conda install keras
6. conda install matplotlib
7. conda install spyder

   or

   conda install jupyter

Save and Load Model

- **model.save('XXXX.h5')**
  - Save the keras model, including the output and input information.
  - You can also save the input or output result respectively.

- **model.load_model('XXXX.h5')**
  - Load the previous model we trained and saved.

Convert TFLite File

- You can use “tf.lite.TFLiteConverter.from_frozen_graph” for pb document
- We use “tf.lite.TFLiteConverter.from_keras_model” for h5 document

```
converter    = tensorflow.lite.TFLiteConverter.from_keras_model(model)
tflite_model = converter.convert()
# 儲存tflite檔案
open("converted_model.tflite", "wb").write(tflite_model)
```

- Convert and save model into tflite document

TFLite File Import (1/2)

- **The trained model tensors need to be loaded in mobile device.**

```
# 讀取模型還有相關模型張量
interpreter = tf.lite.Interpreter(model_path="converted_model.tflite")
interpreter.allocate_tensors()

# 取出模型的IO相關參數
input_details  = interpreter.get_input_details()
output_details = interpreter.get_output_details()
```

TFLite File Import (2/2)

- **Each model has different data shape and tensors. Check your model's data type to assign different shape and data.**

```
#將剛才進行完正規化的圖片作為輸入輸進我們準備好的模型
input_shape    = input_details[0]['shape']
input_data     = np.array(img_norm, dtype=np.float32)
interpreter.set_tensor(input_details[0]['index'], input_data)
interpreter.invoke()
```

Demo

- **MNIST model retrain and convert**
  - **MNIST_changeTFLite.py**
  - **TFLITE_test.py**
  - **10 number images**