

# Homework 3 DeepQ Report

學號：r07942086 系級：電信碩二 姓名：顏宏宇

上傳的 resnet50\_best/variables/ 裡面的 variables.data-00000-of-00001檔案被我切成  
variables.data-00000-of-00001.aa, variables.data-00000-of-00001.ab,  
variables.data-00000-of-00001.ac, variables.data-00000-of-00001.ad  
共四個檔案。

執行 `cat variables.data-00000-of-00001* > variables.data-00000-of-00001` 指令可以cat回去

請分項說明你的 train.py 中所有的設定，包含

(a) 做了那些比較的實驗

(b) 最後選擇此值 / 此 scheduling 的原因

ResNet50:

`seed = None` :

不設定seed

實驗中發現其實各個seed差別不大，只有在要調整其他參數的時候才會  
固定seed

`"pretrained": True` :

是否pretrain都有試過，發現有pretrain訓練收斂速度快非常多

`"arg_scope_dict": {'weight_decay': 0.8}` :

測試過weight\_decay由0.001 至 0.9 的值

0.001的情況下模型幾乎不會被訓練到，後來越條越大，最後選了0.8

`lr = 0.0001` :

試過0.1到0.00005的值，發現0.0001可以到達比較好的accuracy

`optimizer = tf.train.AdamOptimizer(learning_rate=lr)` :

只試過Adam跟原始的，Adam效果好上不少

`get_eval_spec`

`transform = lambda image: tf.image.resize_images(image, [224, 224, 3])` :

有試過shape [224, 224, 3] 跟 [224, 224]，並沒有發現很明顯的差異

`"batchsize": 32` :

試過32跟64，最後是32效果稍微好一點

`before_epoch`

`transform = lambda image: tf.image.resize_images(image, [224, 224, 3])`

有試過shape [224, 224,3] 跟 [224, 224], 也沒有發現很明顯的差異

"batchsize": 32 :

試過32跟64, 最後是32效果稍微好一點

'feed\_dict': {lr: 0.0001} :

值一直都相等於lr, 測試過程跟lr一模一樣

'batch\_norm': 0.9 :

試過False、0.5~0.9, 發現有batch\_norm效果會比沒有好  
最後試到0.9會有比較好的效果

'drop\_out': None

ResNet 沒有dropout層, 固定為None

if n\_epoch == 30:

return 'save\_dir'

50會overfitting, 選了比較不會overfitting的30

mobilenetv2:

seed = None :

不設定seed

實驗中發現其實各個seed差別不大, 只有在要調整其他參數的時候才會  
固定seed

"pretrained": True :

是否pretrain都有試過, 發現有pretrain訓練收斂速度快非常多

"arg\_scope\_dict": {'weight\_decay':0.8} :

測試過weight\_decay由0.001 至 0.9 的值

0.001的情況下模型幾乎不會被訓練到, 後來越條越大, 最後選了0.8

lr = 0.001 :

試過0.1到0.00005的值, 發現在mobilenetv2中0.001可以到達比較好的  
accuracy, 收斂速度也比較合理

optimizer = tf.train.AdamOptimizer(learning\_rate=lr) :

只試過Adam跟原始的, Adam效果好上不少

get\_eval\_spec

transform = lambda image: tf.image.resize\_images(image, [244, 244,,3]) :

有試過shape [224, 224,3] 跟 [224, 224]還有[244, 244,3], 並沒有發現很明顯的差異

"batchsize": 32 :

試過32跟64, 最後是32效果稍微好一點

before\_epoch

transform = lambda image: tf.image.resize\_images(image, [224, 224,3])

有試過shape [224, 224,3] 跟 [224, 224], 也沒有發現很明顯的差異

"batchsize": 32 :

試過32跟64, 最後是32效果稍微好一點

'feed\_dict': {lr: 0.001} :

值一直都相等於lr, 測試過程跟lr一模一樣

'batch\_norm': 0.9 :

試過False、0.5~0.9, 發現有batch\_norm效果會比沒有好  
最後試到0.9會有比較好的效果

'drop\_out': 0.25

試過0.1~0.5之間的值, 對於overfitting來說都有不錯的效果, 最後才選了  
比較中間的0.25

if n\_epoch == 30:

return 'save\_dir'

50會overfitting, 選了比較不會overfitting的30