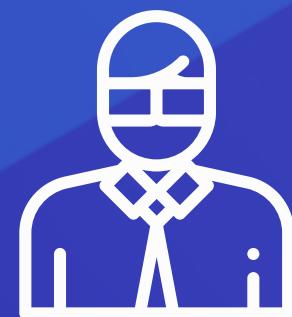




Day 88

初探深度學習使用 Keras

訓練神經網路的細節與技巧
撰寫自己的 callbacks 函數



游為翔

出題教練

知識地圖 深度學習訓練技巧

撰寫自己的 callbacks 函數

深度神經網路

Supervised Learning Deep Neural Network (DNN)

簡介 Introduction

套件介紹 Tools: Keras

組成概念 Concept

訓練技巧 Training Skill

應用案例 Application

卷積神經網路

Convolutional Neural Network (CNN)

簡介 introduction

套件練習 Practice with Keras

訓練技巧 Training Skill

電腦視覺 Computer Vision

深度學習訓練技巧

Training Skill of DNN

應注意的關鍵

防止過擬合 (Overfitting)

超參數 (Hyper-parameters)

學習率 (Learning Rate) 調整

相關訓練技巧

正規化
Regularization

批次標準化
Batch Normalization

回呼
Callback

隨機移除
Drop out

客製化損失函數
Customized Loss Function

提前終止
Early Stopping

本日知識點目標

- 學會如何使用自定義的 callbacks
- 知道 callbacks 的啟動時機

Callbacks

- Callback 在訓練時的呼叫時機
 - `on_train_begin`：在訓練最開始時
 - `on_train_end`：在訓練結束時
 - `on_batch_begin`：在每個 batch 開始時
 - `on_batch_end`：在每個 batch 結束時
 - `on_epoch_begin`：在每個 epoch 開始時
 - `on_epoch_end`：在每個 epoch 結束時

Custom callbacks in Keras

- 在 Keras 中，僅需要實作你想要啟動的部分即可
- 舉例來說，假如你想要每個 batch 都記錄 loss 的話

```
from keras.callbacks import Callback
class My_Callback(Callback):
    def on_train_begin(self, logs={}):
        return

    def on_train_end(self, logs={}):
        return

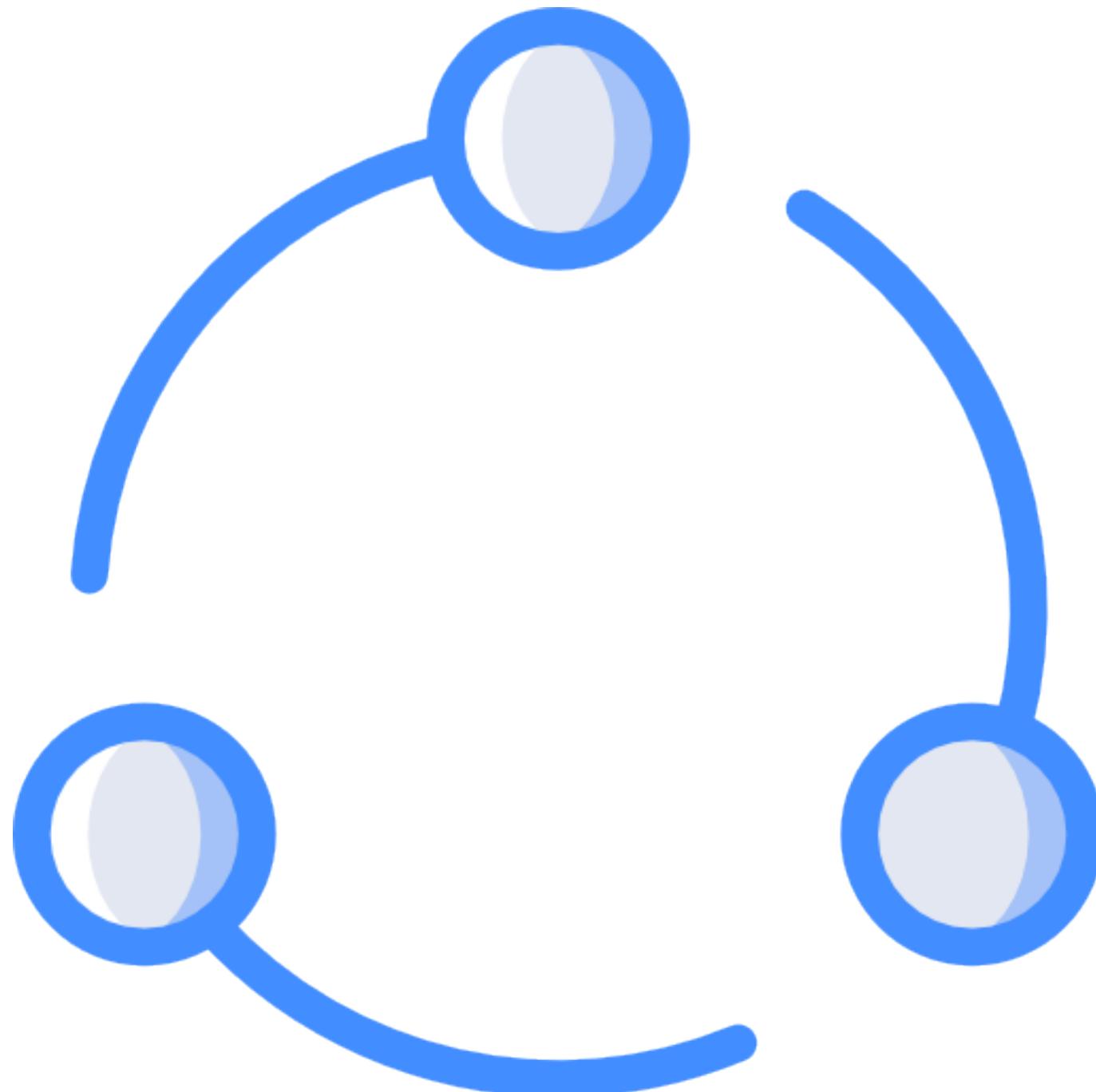
    def on_epoch_begin(self, logs={}):
        return

    def on_epoch_end(self, epoch, logs={}):
        return

    def on_batch_begin(self, batch, logs={}):
        return

    def on_batch_end(self, batch, logs={}):
        self.losses.append(logs.get('loss'))
        return
```

重要知識點複習：



- Callbacks 可以在模型訓練的過程中，進行監控或介入。Callbacks 的時機包含：
 - `on_training_begin`
 - `on_epoch_begin`
 - `on_batch_begin`
 - `on_batch_end`
 - `on_epoch_end`
 - `on_training_end`



延伸 閱讀

除了每日知識點的基礎之外，推薦的延伸閱讀能補足學員們對該知識點的了解程度，建議您解完每日題目後，若有
多餘時間，可再補充延伸閱讀文章內容。

推薦延伸閱讀

知乎 - 紀錄 F1-score 的 callback 實作（簡體）[連結](#)

原生的 Keras 並沒有提供紀錄 F1-score 的方法；在 Sklearn 中計算 F1-score 很容易，有沒有可能將兩個兜在一起？

- 寫一個 callback, 它的啟動時機是在每個 epoch 結束時
- 呼叫 sklearn 的 f1_score 計算
- 將數值寫入 self.val_f1s

最後在將這個 callback 與其它 callbacks 一起放入 model.fit 就好了。

```
def boolMap(arr):
    if arr > 0.5:
        return 1
    else:
        return 0

def on_epoch_end(self, epoch, logs=None):
    val_predict = list(map(boolMap, self.model.predict([self.validation_data[0], self.validation_data[1]])))
    val_targ = self.validation_data[2]
    _val_f1 = f1_score(val_targ, val_predict)
    _val_recall = recall_score(val_targ, val_predict)
    _val_precision = precision_score(val_targ, val_predict)
    self.val_f1s.append(_val_f1)
    self.val_recalls.append(_val_recall)
    self.val_precisions.append(_val_precision)
    print(_val_f1, _val_precision, _val_recall)
    print("max f1")
    print(max(self.val_f1s))
    if _val_f1 > self.best_val_f1:
        self.model.save_weights(self.file_path, overwrite=True)
        self.best_val_f1 = _val_f1
        print("best f1: {}".format(self.best_val_f1))
    else:
        print("val f1: {}, but not the best f1".format(_val_f1))
    return
```



解題時間

It's Your Turn

請跳出PDF至官網Sample Code & 作業
開始解題

