

Day 68

Sequential API

序列模型搭建網路




本日知識點目標



目標
知識點

了解Keras Sequential API



獲得
知識點

完成今日課程後你應該可以了解

- 了解Keras Sequential API 與其應用的場景

Sequential API

- sequential是一系列模型的簡單線性疊加，可以在構造函數中傳入一些列的網路層：

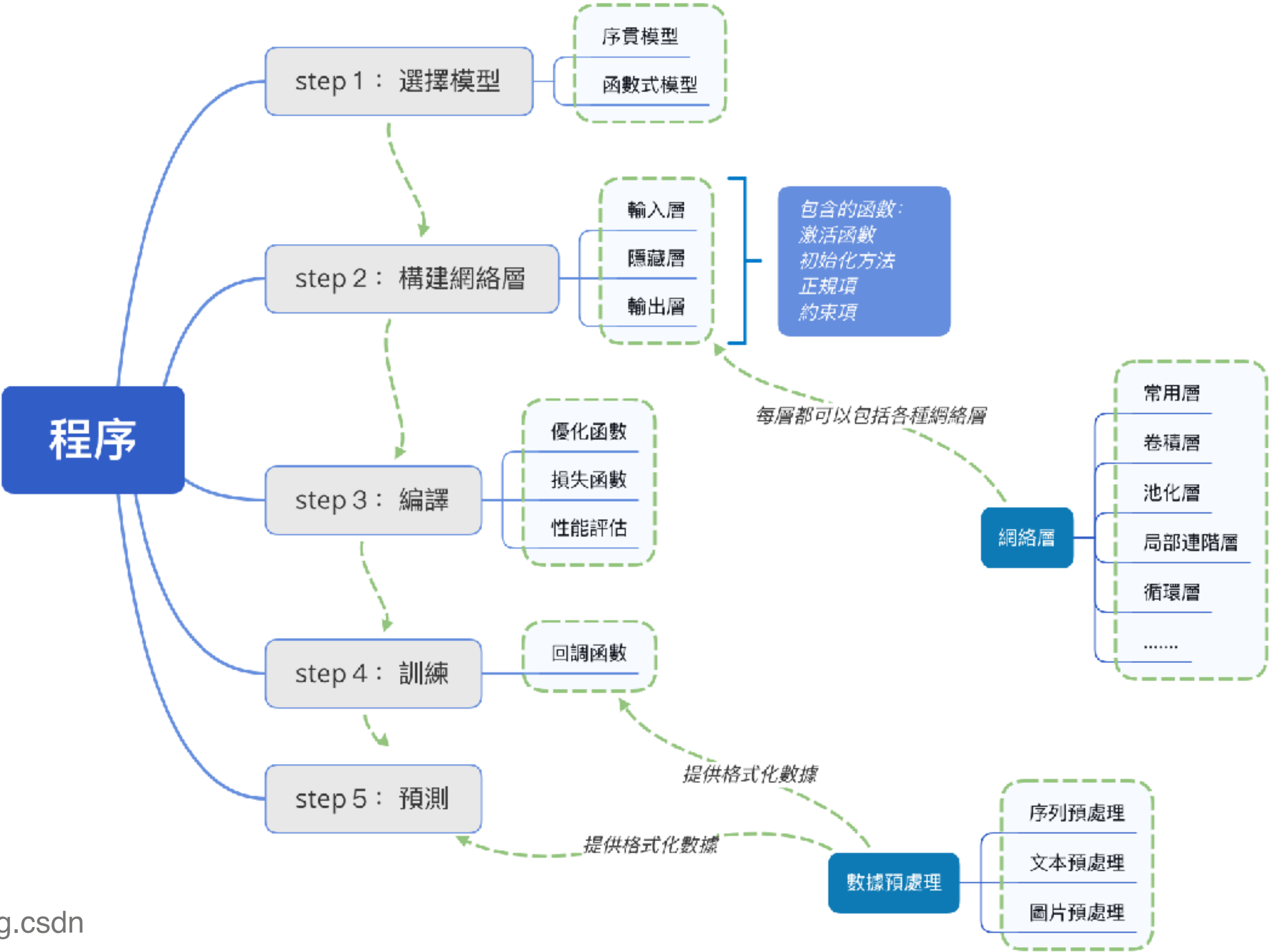
```
from keras.models import Sequential  
from keras.layers import Dense, Activation
```

```
model = Sequential([Dense(32, _input_shape=(784,)), Activation("relu")])
```

- 也可以透過 .add

```
model = Sequential()  
model.add(Dense(32, _input_dim=784))  
model.add(Activation("relu"))
```

如何設定



指定模型的輸入維度

- Sequential的第一層(只有第一層，後面的層會自動匹配)需要知道輸入的 shape
 - 在第一層加入一個input_shape參數，input_shape應該是一個shape的 tuple資料類型。
 - input_shape是一系列整數的tuple，某些位置可為None
 - input_shape中不用指明batch_size的數目。

指定模型的輸入維度

- 2D的網路層，如Dense，允許在層的構造函數的input_dim中指定輸入的維度。
- 對於某些3D時間層，可以在構造函數中指定input_dim和input_length來實現。
- 對於某些RNN，可以指定batch_size。這樣後面的輸入必須是(batch_size, input_shape)的輸入

常用參數說明

名稱	作用	原型參數
Dense	實現全連接層	Dense(units,activation,use_bias=True, kernel_initializer='glorot_uniform', bias_initializer='zeros')
Activation	對上層輸出應用激活函數	Activation(activation)
Dropout	對上層輸出應用 dropout 以防止過擬合	Dropout(ratio)
Flatten	對上層輸出一維化	Flatten()
Reahape	對上層輸出reshape	Reshape(target_shape)

前述流程 / python程式 對照

```
# build our CNN model, 多加幾層
model = Sequential()
model.add(Conv2D(64, (3, 3), padding='same',
                 input_shape=x_train.shape[1:]))
model.add(Flatten())
model.add(Dense(512))
model.add(Activation('relu'))
model.add(Dropout(0.5))
model.add(Dense(num_classes))
model.add(Activation('softmax'))
```


重要知識點複習

- Keras函數式模型接口是用戶定義多輸出模型、非循環有向模型或具有共享層的模型等複雜模型的途徑
- 所有的模型都是可調用的，就像層一樣
- 使用函數式模型的一個典型場景是搭建多輸入、多輸出的模型
- 另一個使用函數式模型的場合是使用共享層的時候

解題時間 It's Your Turn

請跳出PDF至官網Sample Code & 作業
開始解題

