

Lab 2 結報

姓名：賴昱凱

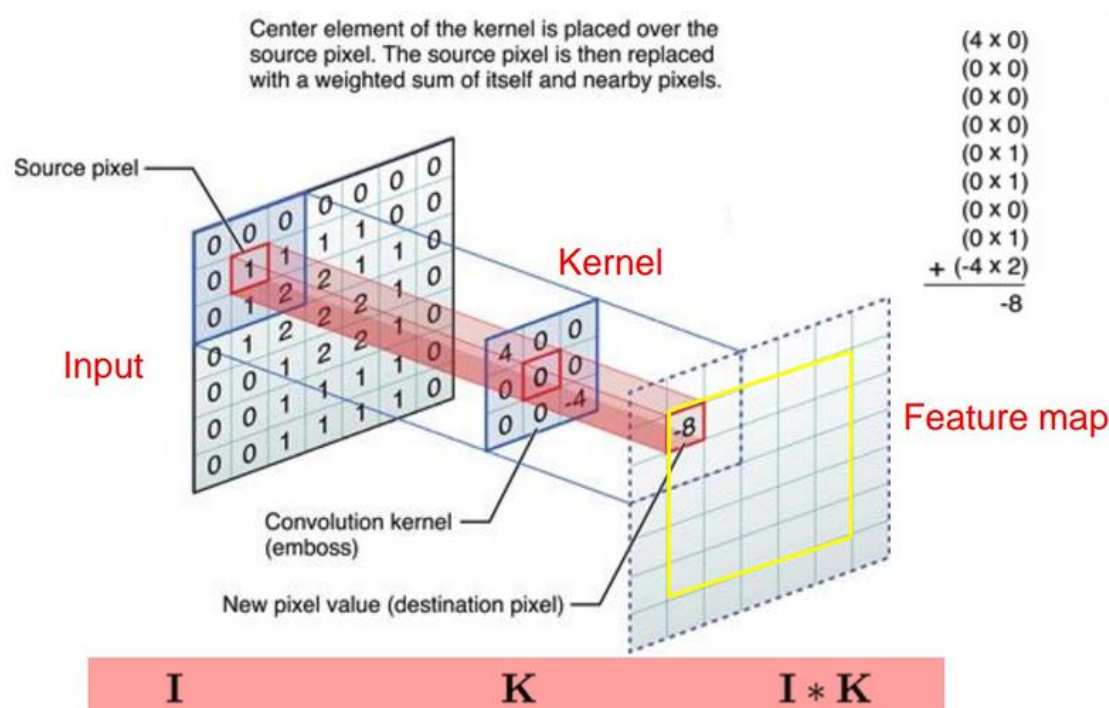
學號：111511141

1. 請敘述在課堂上實作之神經網路的架構

Model: "sequential"		
Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 56, 32)	416
max_pooling1d (MaxPooling1D)	(None, 28, 32)	0
batch_normalization (Batch Normalization)	(None, 28, 32)	128
dropout (Dropout)	(None, 28, 32)	0
conv1d_1 (Conv1D)	(None, 28, 64)	6208
max_pooling1d_1 (MaxPooling1D)	(None, 14, 64)	0
batch_normalization_1 (Batch Normalization)	(None, 14, 64)	256
dropout_1 (Dropout)	(None, 14, 64)	0
conv1d_2 (Conv1D)	(None, 14, 128)	24704
max_pooling1d_2 (MaxPooling1D)	(None, 7, 128)	0
batch_normalization_2 (Batch Normalization)	(None, 7, 128)	512
dropout_2 (Dropout)	(None, 7, 128)	0
conv1d_3 (Conv1D)	(None, 7, 128)	49280
max_pooling1d_3 (MaxPooling1D)	(None, 3, 128)	0
batch_normalization_3 (Batch Normalization)	(None, 3, 128)	512
dropout_3 (Dropout)	(None, 3, 128)	0
flatten (Flatten)	(None, 384)	0
dense (Dense)	(None, 256)	98560
dense_1 (Dense)	(None, 64)	16448
dense_2 (Dense)	(None, 6)	390
Total params: 197,414		
Trainable params: 196,710		
Non-trainable params: 704		

本次實驗主題為 CNN，全名為 Convolutional Neural Network。通常 CNN 適用於圖像相關應用，因此這邊先以最常見的 2D CNN 來做解釋。

CNN 是針對具有空間結構的資料（如圖片等）進行特徵擷取的神經網路架構。它的主要運作方式是透過卷積核（filter 或 kernel）在圖片上滑動，進行所謂的 Convolution 的計算，得到局部區域特徵擷取的作用，得知物體邊緣、紋理、形狀等特徵。卷積層後常接續使用激活函數（如 ReLU 等）來引入非線性，這與上一次實驗的 DNN 中的激活函數功能是相同的，另外還有 Padding、Pooling 等功能協助 CNN 的訓練。



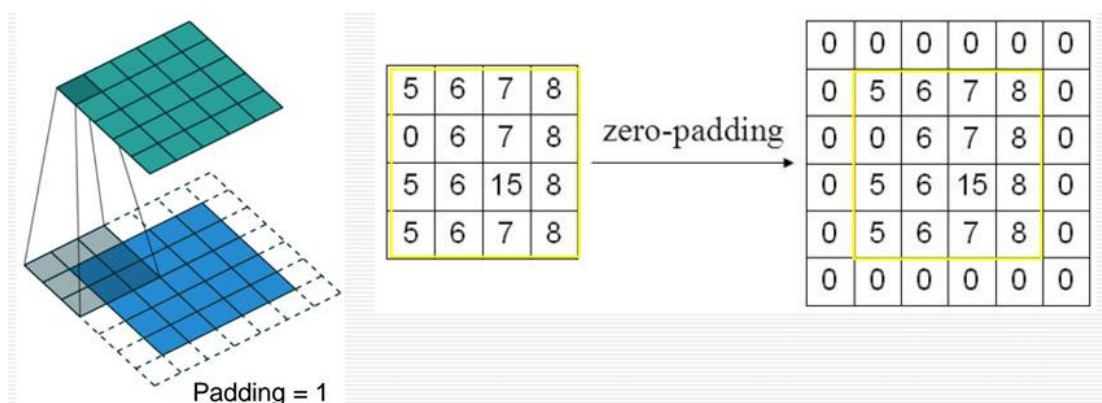
上面提到的 Convolution 計算如上圖所示，他利用一個 kernel 在圖片上滑動，而對應到 kernel 的 pixel 將與 kernel 做位置上一對一的相乘，並加總即為該層 output 對應的 pixel，也稱為 feature map。而其計算量為：

$$D_k \times D_k \times M \times (N \times D_f \times D_f)$$

因為 convolution 計算方式就是將 kernel 大小的部分一對一相乘再相加

得到 output 的一個像素，當 kernel 像素總數 $D_k \times D_k$ ，且 input 有 M 個 channel 時，每計算一個 output 像素的計算量就為 $D_k \times D_k \times M$ 。若我們希望 output 的大小為 $D_f \times D_f$ ，有 N 個 output channel 所需總計算量就為 $D_k \times D_k \times M \times (N \times D_f \times D_f)$ 。

而圖中可以發現，若直接對圖片做 Convolution 計算，將會導致 output 的尺寸變小，若多做幾次 Convolution 將會導致 feature map 小到無法代表特徵，因此這裡就會用到剛剛提到的 padding，這是在 input 的最外圍加上額外的 pixel，可以使 feature map 保持相同的尺寸，讓 CNN 可以有更深的架構，常見的有 Zero Padding 等。Pooling 則是在下一題中解釋。



透過多層 Convolution 的堆疊，CNN 能夠學習越來越抽象、複雜的圖片特徵。最終，透過 flatten 連接全連接層（Dense Layer，如 Lab1），將特徵圖轉換為向量表示，並進行分類、回歸等任務。Flatten 就是將 2D 的 feature map 直接攤平變成一維的向量，讓後續的 MLP 可以連接計算。

在本實驗中，雖然資料型態為一維，但我們仍可套用 CNN 架構，使用 1D 的 CNN 對序列進行特徵擷取，其原理與 2D CNN 類似，只是卷積的操作改為在單一維度上進行。

2. 請敘述 Maxpooling、Dropout 與 Batch Normalization 的作用

(1) MaxPooling

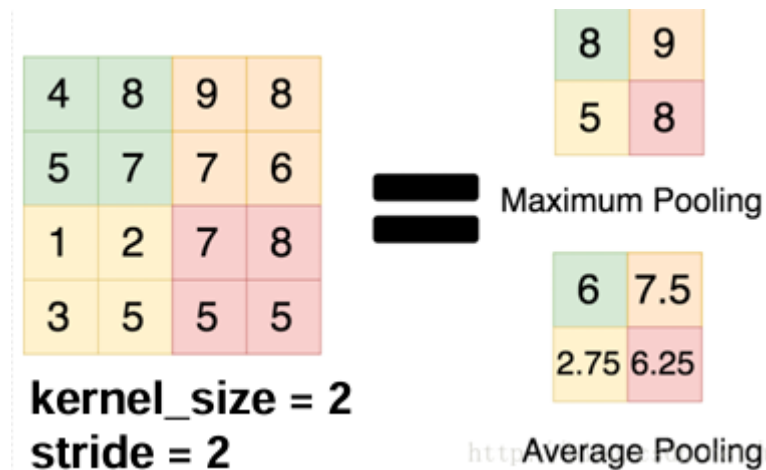
MaxPooling 是一種下採樣 (downsampling) 操作，主要目的是降低特徵圖的空間維度，同時保留最具代表性的特徵。它的運作方式是在一個固定大小的區塊內 (如 2x2 範圍) 取最大值，它的優點有：

- 減少參數量與計算成本

- 提高模型對於平移與局部扭曲的穩定性

- 在視覺任務中幫助模型聚焦於最突出的特徵

除了 MaxPooling，Average Pooling 也是常見的 Pooling 方式之一。



(2) Dropout

Dropout 是一種常用的正則化 (regularization) 技術，也就是代表可以讓 Testing 的表現更好，對模型來說更泛化。在訓練時會隨機「關閉」一部分神經元，這樣做的目的為：

- 避免模型過度依賴某些特徵 (防止 overfitting)

- 強迫網路學會更有彈性、更具泛化能力的特徵表示

- 提高測試時的穩定性與表現

與其他方法不同的是，Dropout 僅於訓練階段作用，測試時會自動關閉這個功能，讓所有的神經元都啟用。

(3) Batch Normalization

Batch Normalization 是一種用於穩定與加速訓練的技術，它的作用是在每一層輸出中，對一個 batch 的資料進行標準化，也就是平均為 0，標準差為 1，並且該層會有可學習的縮放與偏移參數。Batch Normalization 主要優點包括：

穩定梯度，減少梯度爆炸或消失問題

加速收斂速度，減少對參數初始化與對 learning rate 的敏感性

正則化效果，有助於降低 overfitting

3. 心得

這次的實驗進階到 CNN 的模型，這也是在 Transformer 之前最主流的視覺模型，就算到現在，許多不需要太大量計算量的模型也是使用 CNN-Based 的模型，因此這堂課對於 AI 新手來說是一個十分重要的課程。這次實驗過程中，讓我們很明顯的感受到 Shuffle、Dropout 與 Batch Normalization 的作用，因為在一開始沒有加上這些功能前，模型訓練出來的內容就是 training accuracy 超級高，但 validation 以及 testing accuracy 很慘，明顯就是 overfitting，但當我們一加上這些功能後，模型立刻變得非常優秀，不但 Training 表現仍然極佳，validation 馬上就跟上，表現也與 Training accuracy 差不多，讓我們親自動手感受到正則化（regularization）相關功能的影響力以及重要性。