

PDS Final

楊書暉 M132040015

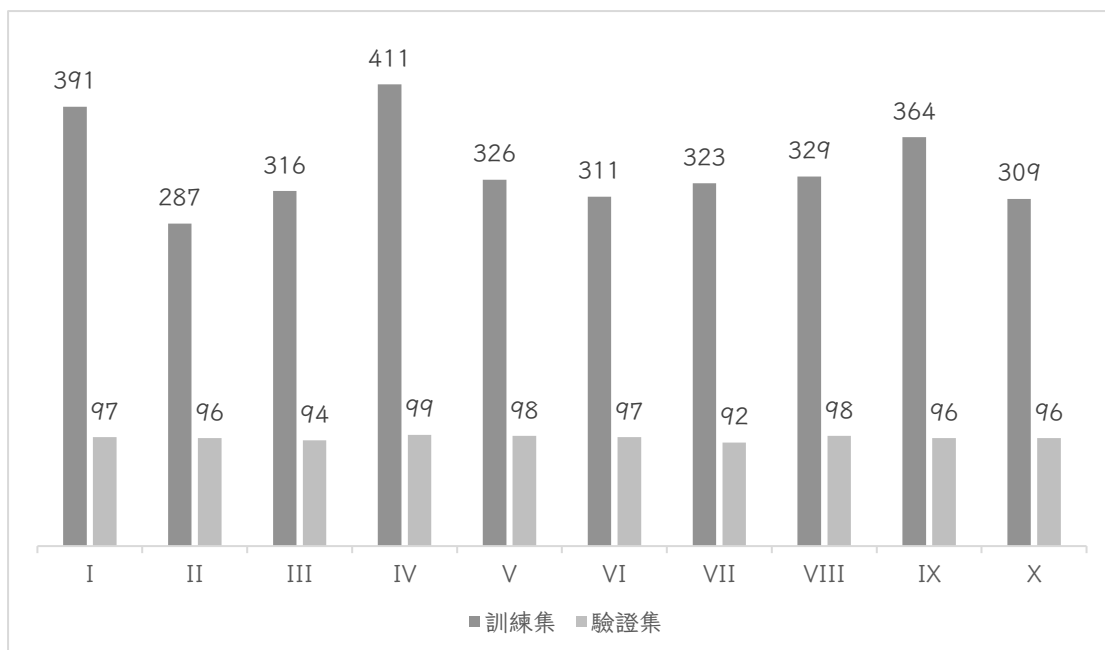
葉宇倫 M132040013

簡述

從原始資料集初始 0.65 的準確度，經過清理、移動以及增加資料、調整參數後達到了 0.85 的準確度。

原始資料

在原始資料裡，我們總共有 3367 張訓練圖像以及 963 張驗證圖像(圖一)。













(圖一)各類圖像數量

資料預處理

經過檢查後，發現資料有兩大問題：錯誤標籤、錯誤圖像(圖二)：

這兩個問題處處皆存在，為了解決問題，我們決定使用同樣的模型(競賽的固定模型)去做錯誤圖像識別，不過為了提高識別準確度，我們先使用了 CleanVision 套件觀察圖像問題(表一)，並根據圖像問題，來設定影像增強參數，以提高辨識準確度，並且我們將訓練集與驗證集融合，當成單一資料放入模型中訓練，在訓練完模型後，分別對訓練集和驗證集預測 30 次，對這 30 次的預測機率取平均值，以穩定預測。預測完後，使用 CleanLab 套件裡面的 find_label_issues 快速找出相關問題(*1)，找出問題後再決定是修改還是保留。

錯誤放置				
i_57.png	ii_7.png	v_201.png	viii_75.png	ix_5.png
				
錯誤圖像				
iii_15.png	iv_58.png	vi_307.png	vii_256	x_40.png
				

(圖二)資料問題

	訓練集問題	驗證集問題
Light	1088	*普遍性 > 0.5
Near duplicates	92	22
Odd size	62	5
Odd aspect ratio	1	0
備註	Low information 與 grayscale 普遍性 > 0.5	Low information 與 grayscale 普遍性 > 0.5

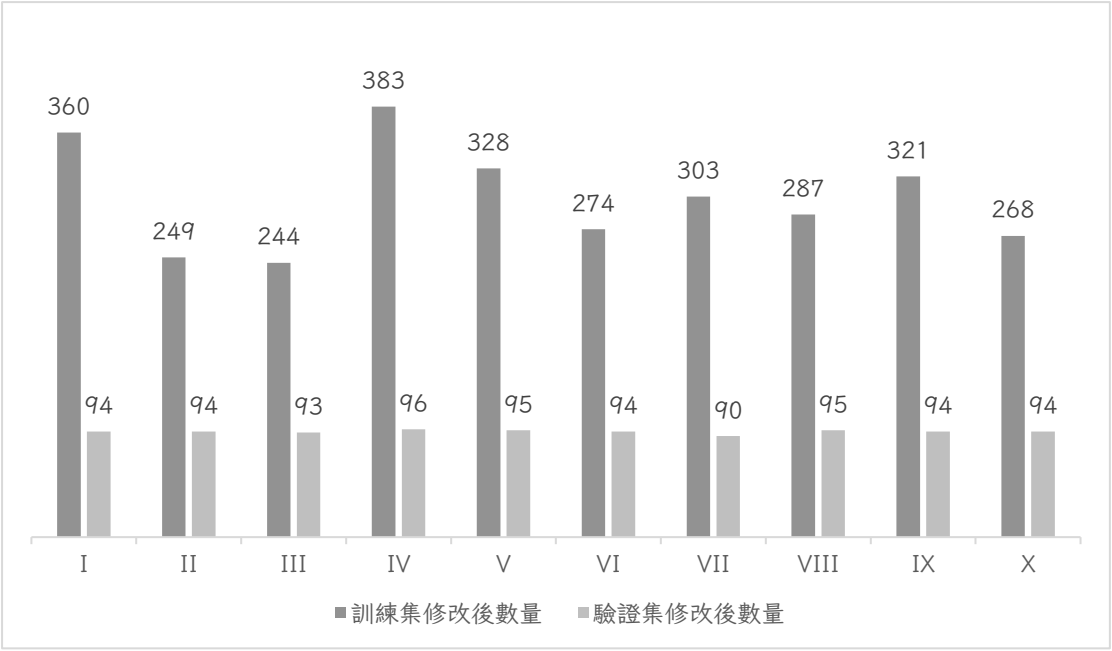
(表一) CleanVision 圖像問題

*普遍性 > 0.5：含有此類問題的圖像超過資料筆數的一半

(*1) 使用篩選器為 'both' 同時使用兩種篩選方法：

- prune by noise rate：高機率預測錯誤
- prune by class：在正確標籤上預測為最小機率

修改完原始資料後總圖像數量為訓練集：3017 筆、驗證集：939 筆



(圖三)各類圖像修改後數量

增加資料

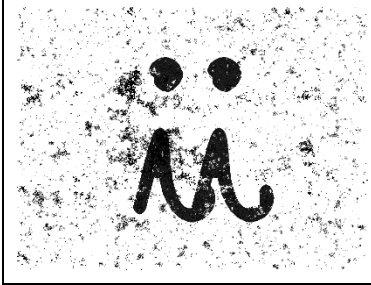
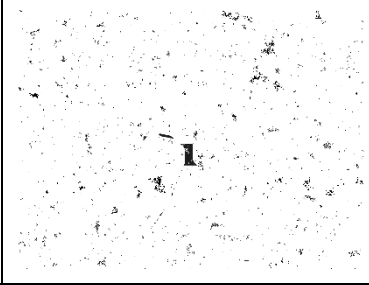
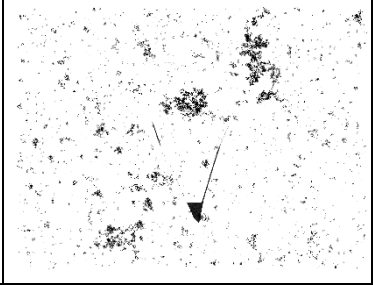
在修改完資料後經過初步的影像增強，能夠達到 0.7 左右的準確度，這離 0.85 的準確度還遠遠不足。另外我們知道任何羅馬數字都可以由 "I, V, X" 字母組合而成，為了增加圖像的多樣性，我們選擇了 Chars74k 這個資料集，資料集裡正好包含了以上的字母(*2)。

透過 CV2 套件，使用 "add weighted" 我們能夠將 "I, V, X" 組合而成，並且透過將圖像加入噪聲(表二)的方式模擬原始圖像。

加入噪聲	特性	範圍
Guassain Noise	值分佈接近於平均值，波動小	平均：128、變異數：20
Uniform Noise	值在範圍內均勻隨機分佈	0, 255
Impulse Noise	部分像素的值被隨機置為最大值或最小值	250, 255
備註：我們為值取了(0, 0.2)的隨機權重。		

(表二)加入噪聲

在為圖像添加完上述方法後，我們得到了每組各 110 張的圖片(大小寫各 55 張)，由於圖像有好有壞，有時圖片會有易被混淆的問題；另外，噪聲加入方式也是隨機的，圖像有時也會被破壞(圖四)。因此，此次資料添加提升準確度只能到達 0.72 左右。

易被辨識為 V；標籤為 II	被破壞圖像；標籤為 X	過細不易辨識；標籤為 V
		

(圖四)生成不佳的圖像

(*2) 這個資料集裡包含了數字與字母大小寫的手寫資料與圖片，這裡我們只使用手寫資料。

分類增強

為了提高準確度，更多的圖像是必要的，我們決定使用現有的圖像進行隨機增強加入資料集中。為了確保圖像數量保持在 12000 張的限制中，我們將訓練集與驗證集混合，將每個類別增加到 1200 張後打亂放回，以確保圖像品質一致。

在增強時，為了適應各數字不同的特徵，我們將所有數字的參數分開來設定，以達到最好的參數(表三)；並將訓練集與驗證集採 8000 張與 4000 張的初始配置。

使用參數	解釋
*Resize	將圖像大小調整為指定高度和寬度
Fliplr	將圖像水平翻轉
Flipud	將圖像垂直翻轉
Crop	裁剪圖像
*GaussianBlur	高斯模糊
*AdditiveGaussianNoise	高斯雜訊
LinearContrast	將每個像素縮放來調整對比度
Multiply	將圖像中的所有像素乘以特定值

Affine - scale	將圖像縮放
Affine - translate_percent	將圖像平移
Affine - rotate	將圖像旋轉
Affine - shear	將圖像剪切

(表三)初始使用參數

*Resize：設定為 400*400 以保留更多細節







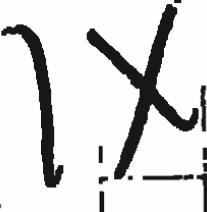
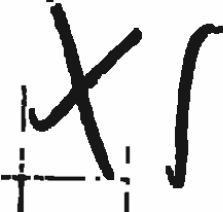
*高斯模糊、雜訊：互為拮抗關係，模糊有降低噪聲影響效果。

調整參數

在設定完初始參數後，調整參數使其 F1-score 達到 0.85 就是最後的工作，由於每個類別都有不同的優缺點，並且水平翻轉以及垂直翻轉無法適用到所有類別上(表四、圖五)，為了更加細緻的微調參數，我們決定為每個類別分開進行調整。並且，為了在修改後方便辨別準確度，我們使用了人工標註後的*測試集進行評分比較(*3)，在大量嘗試後，我們將訓練集與驗證集的數量改成 11000 張與 1000 張，並且將每個類別都進行了大量的修改，最終我們得到了能夠達到 0.85 準確度以上的參數。

可被水平翻轉	I, II, III, V, X
可被垂直翻轉	I, II, III, IX, X

(表四)可被翻轉類別

成功範例		失敗範例	
翻轉前	翻轉後	翻轉前	翻轉後
			
			

(圖五)翻轉範例圖像

*測試集：testing_data ，並未加入至訓練集中
(*3)人工標註不是 100%準確，會有 ± 0.02 的誤差。

問題與討論

在調整參數時，有遇到幾項常見問題：

- 參數隨機性過高：在設置參數時，起初為了更微小的調節，在隨機性上，我們為高斯模糊加上了只有 50%機率起作用的設置，後續因不易使用，故關閉此項。
- I, III, V 圖像不佳：在與真實標籤的比較中，這三種的預測錯誤是最常發生的，因此參數調整，也著重在這三類圖像的調整。
- V、X 相似性過高：在觀察 f1-score 中，可以觀察到 V 與 X 會互相影響：V 的精確率(Precision)與 X 的召回率(Recall)兩者同時偏低(圖六)，在最後將 X 的高斯模糊拉高，解決此問題。

	precision	recall	f1-score	support
1	0.80	0.94	0.87	52
2	0.87	0.81	0.84	59
3	0.73	0.70	0.72	54
4	0.89	0.94	0.92	52
5	0.71	0.88	0.78	41
6	0.86	0.76	0.81	50
7	0.71	0.78	0.74	41
8	0.91	0.91	0.91	47
9	0.86	0.78	0.82	49
10	0.83	0.69	0.75	55
accuracy			0.82	500
macro avg	0.82	0.82	0.82	500
weighted avg	0.82	0.82	0.82	500

(圖六)調整中報表

- 可能的設備問題：使用同一參數進行調整時，在 Kaggle 平台上，我們兩人的準確度(valid accuracy)差距能達到 0.1 以上，後葉宇倫轉到 Colab 平台上解決此問題。
-

嘗試與結果

在這章節，我將會列出我們嘗試過的一些事情與結果

- 資料預處理

- i. 錯誤導向學習(Error-driven learning)：

- ◆ 概念：根據模型預測結果與真實值的差異來調整模型參數。在這裡，我們將方式改為，模型預測不符正確標籤的圖像取出進行增強後放回。
 - ◆ 表現：使用過後能夠提升模型的預測表現，簡化預處理過程。
 - ◆ 問題點：增強參數過於保守，效果提升有限、嘗試次數不足。礙於時間關係，故放棄使用。

- ii. Wasserstein GAN：

- ◆ 概念：這是一種改進後的生成對抗網路，與傳統 GAN 相比能提高學習的穩定性。
 - ◆ 表現：提升效果待確定
 - ◆ 問題點：迭代次數不足，使得生成圖像不佳，嘗試次數不足。礙於時間關係，故放棄使用。

- 調整參數

- i. 根據 f1-score 改變權重：

- ◆ 在改變訓練集與驗證集數量時，我們有嘗試過每個類別分開調整數量，與調整圖像類別總數。
 - ◆ 例：將 VIII 的總數降低至 500 張，將多出圖像平分增加至 V 與 VI。提升效果不佳，與原始相同。
 - ◆ 問題點：在圖像原本就不好的情況下，調整圖像數量無法帶來幫助。

- ii. 根據模型預測做出混淆矩陣：

- ◆ 放棄：過度為測試集調整會帶來過擬合的風險，不使用。

結論

在這次競賽中，我們首先清理和移動了大量的錯誤圖像，並在後續增加了額外的手寫資料集，增加圖像多樣性。在之後，我們採取分類增強的方法將資料增加到最大限制的 12000 張，採取了良好的參數設置，並且將訓練集的數量從 8000 張提升至 11000 張，部分解決了上述提到的問題，將 f1-score 的評分成功的保持在 0.84 以上，並且有機會達到 0.87 的分數。

貢獻與感謝

我與葉宇倫都是一起討論及共同編寫程式碼，我們決定兩人貢獻相同。

特別感謝：

- 薄育文：提高訓練集數量的提示幫大忙了。
 - 阮柏誠：一起熬夜調參數，讓我不孤單。
 - 我媽：給我電話號碼，讓我多出 30 小時調整參數。
-

參考資料

- [GitHub - kennethleungty](#)
- [The Beginning! | Agneev's DS/ML lab book](#)
- [The Chars74K image dataset](#)
- [eriklindernoren/Keras-GAN: Keras implementations of Generative Adversarial Networks.](#)
- [李宏毅 ATDL Lecture 15 - HackMD](#)

附錄 – 最終參數與原始參數比較

	Fliplr	Flipud	Crop	GaussianBlur	AdditiveGaussianNoise	LinearContrast	Multiply	Affine
原始	0.5	0.5	(0, 0.05)	sigma = (0, 0.5)	scale = (0, 0.05*255)	(1.35, 1.75)	(0.8, 1.2)	縮放 : (0.8, 1.2) 平移 : (-0.06, 0.06) 剪切 : (-3, 3) 旋轉 : (-20, 20)
I	0.5	0.2	(0, 0.05)	sigma = (0, 3)		(1.2, 1.5)	(0.85, 1.15)	縮放 : (0.9, 1.1) 平移 : (-0.05, 0.05) 剪切 : (-3, 3) 旋轉 : (-15, 15)
II	0.5		(0, 0.05)		scale = (0, 0.05*255)	(1.2, 1.4)	(0.9, 1.1)	縮放 : (0.9, 1.1) 平移 : (-0.04, 0.04) 剪切 : (-2, 2) 旋轉 : (-10, 10)
III	0.5	0.2	(0, 0.03)	sigma = (0, 2)	scale = (0, 0.05*255)	(1.1, 1.4)	(0.9, 1.2)	縮放 : (0.85, 1.15) 平移 : (-0.04, 0.04) 剪切 : (-3, 3) 旋轉 : (-10, 10)
IV			(0, 0.03)	sigma = (0, 1)		(1.2, 1.5)	(0.85, 1.15)	縮放 : (0.9, 1.1) 平移 : (-0.05, 0.05) 剪切 : (-4, 4) 旋轉 : (-15, 15)
V	0.25		(0, 0.05)	sigma = (0, 4)		(0.7, 1.4)	(0.55, 1.2)	縮放 : (0.9, 1.15) 平移 : (-0.06, 0.06) 剪切 : (-2, 2) 旋轉 : (-10, 10)
VI			(0, 0.03)	sigma = (0, 0.5)		(1.1, 1.4)	(0.9, 1.1)	縮放 : (0.85, 1.15) 平移 : (-0.08, 0.08) 剪切 : (-5, 5) 旋轉 : (-20, 20)
VII			(0, 0.03)	sigma = (0, 1)		(1.1, 1.4)	(0.9, 1.1)	縮放 : (0.85, 1.15) 平移 : (-0.08, 0.08) 剪切 : (-5, 5) 旋轉 : (-20, 20)

VIII			(0, 0.05)	sigma = (0, 1.5)		(1.2, 1.6)	(0.85, 1.15)	縮放 : (0.85, 1.15) 平移 : (-0.06, 0.06) 剪切 : (-4, 4) 旋轉 : (-15, 15)
IX		0.5	(0, 0.05)	sigma = (0, 0.7)	scale = (0, 0.05*255)	(1.2, 1.6)	(0.85, 1.15)	縮放 : (0.85, 1.15) 平移 : (-0.06, 0.06) 剪切 : (-4, 4) 旋轉 : (-15, 15)
X	0.5	0.5	(0, 0.05)		scale = (0, 0.85*255)	(1.2, 1.6)	(0.85, 1.15)	縮放 : (0.85, 1.15) 平移 : (-0.06, 0.06) 剪切 : (-4, 4) 旋轉 : (-15, 15)

*紅字為有改變參數

*打叉為未使用