DLCV Final: Long Tail Annihilator

Hung-Yu Shu, Yi-Hsien Lin, Cheng-You Tsai, Chih-Ying Liu, Pin-Hua Lee
舒泓諭*, 林奕憲, 蔡承佑, 劉知穎, 李品樺
R09993021, D06943006, R10943014, B07901039, B07303024
Team: NO QQ NO LIFE (*: Leader)
{r09993021, d06943006, r10943014, b07901039, b07303024}@ntu.edu.tw

1 Methodology and Model Architecture

1.1 Backbone Model

我們依循原始資料分布規則取樣(Uniform Sampler) 測試了四種模型架構，包括兩種CNN跟兩種Transformer，實驗結果發現Swin Transformer[1]的效果最好，它因為會將圖拆成不同的大小階層做Self-Attention，相較於ViT，可以更全面的抓出各種細微的特徵，所以我們所有實驗的Backbone模型都使用它。

Backbone	Main	Frequent	Common	Rare
ResNest269	67.53%	86.68%	64.52%	20.46%
EfficientNet-B7	64.99%	82.65%	63.03%	21.79%
ViT	60.56%	83.61%	56.38%	8.34%
Swin Transformer	**73.76%**	**90.68%**	**72.93%**	**31.20%**

Table 1: Result of different backbones

1.2 Train Decoupling [2]

Decoupling會將模型訓練分成兩階段，第一階段會用規則取樣來訓練，而在第二階段時，它會換不同的取樣方法來finetune模型，均衡取樣(Balanced Sample 會讓每種類別抽到的機率都相同，而反轉取樣(Reversed Sampler)則是將趨勢反過來，讓原本最稀少的類別抽出機率最高，我們實驗依據這兩種取樣跟需要訓練的架構分成六種情形來研究其效果。

1.3 Bilateral-Branch Network(BBN) [3]

為了嘗試結合上述第二階段兩種取樣方法的優點，我們就以BBN為出發點，改良了第二階段的訓練方式，BBN的核心概念是兩條訓練分支共享同個特徵擷取器，但使用不同的取樣方法，測試的時候，兩條分支各自訓練出來的分類器會把結果相加平均當作分類的結果，不過它們在訓練時，兩條分支並非都維持相同的權重，隨著Epoch增加會有不同變化，拋物線狀的權重設計會讓準確度最高。原始的BBN架構是用Resnet當作特徵擷取器，我們改用Swin Transformer，把BBN當作Decoupling訓練的第二階段，但後面的分類器參數設為亂數，其次將第一條分支，從規則取樣改成均衡取樣，我們認為這樣應該更能穩定三種不同分類的準確率。

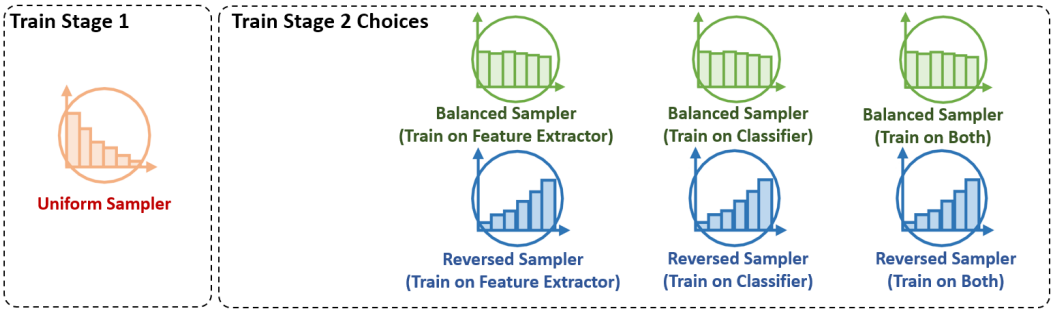


Figure 1: The proposed decoupling pipeline.

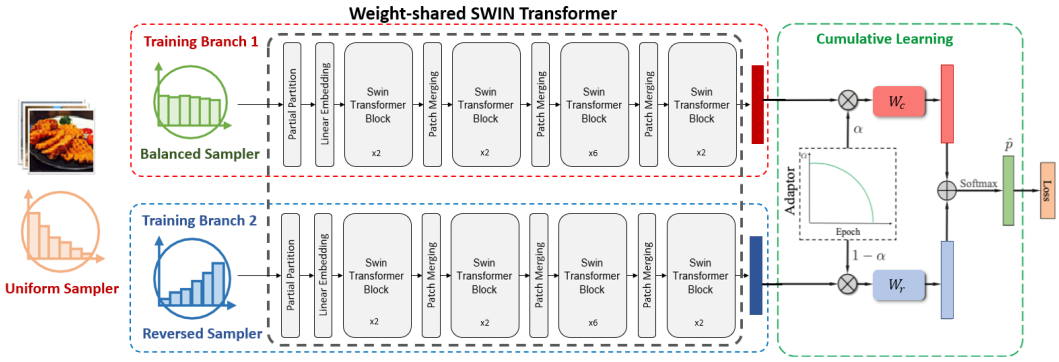


Figure 2: The proposed BBN pipeline.

2 Implementation Details

2.1 Data Preprocessing

除了先前提到的三種不同的資料取樣方式外，我們也在所有訓練時做以下的影像資料增强(Data Augmentation)，包括縮放、隨機水平翻轉($p = 0,5$)、隨機角度旋轉(10度)、正規化(Imagenet常用參數)。

2.2 Hyperparameter Choices

這邊列出我們所使用Ensemble各模型的參數。

Backbone	Image size	Learning Rate	Batchsize	Max Epoch
Decoupling Stage 1 (Swin)	384	10^{-5}	4	10
Decoupling Stage 2 (Swin)	384	10^{-5}	4	10
Decoupling Stage 1 (Resnest269)	320	10^{-5}	4	3
Decoupling Stage 2 (Resnest269)	320	10^{-5}	4	4
BBN	384	10^{-5}	6×20	10

Table 2: Hyperparameter list

3 Experiments

3.1 Attention Map and TSNE for Decoupling

我們可以看出第一階段抽出的Attention map已經有不錯的結果，第二階段過後，我們發現它會連食物的容器也一起抓出來協助判斷分類，像是上面這個食物叫做糖油粑粑，是類似燒

麻糬的東西，他都會用盒子裝，下面的滷蛋則是用湯匙來裝。TSNE的結果，以藍色這組為例，深藍是Rare類別的炸廣味香腸，淺藍是Frequent的臘腸，是容易被混淆的類別，可以看到原先在第一階段被判斷成淺藍色的部分，在第二階段被判斷正確深藍的比例大幅提高。

Decouple (Stage 1 Training)		Decouple (Stage 2 with Reversed Sampler)	
			

Table 3: Attention Map

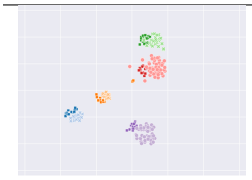
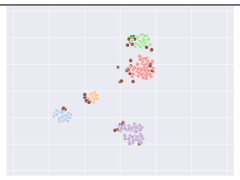
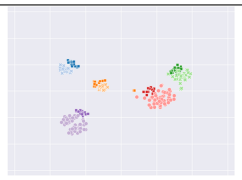
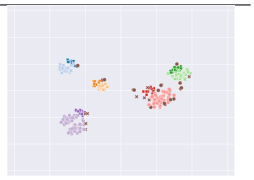
Decouple (Stage 1 Training)		Decouple (Stage 2 with Reversed Sampler)	
Label	Predict	Label	Predict
			

Table 4: TSNE

3.2 Quantitative Measurements

3.2.1 Train Decoupling

雖然採用反轉取樣會讓Frequent下降(與U10比較)，其餘類別有顯著提升，不過在原始Decoupling論文中認為第二階段只需訓練在分類器就好，因為第一階段已經學到不錯的特徵，不須重新訓練，在我們的實驗結果也有觀察到，這樣Rare的成效是最好的，而且訓練的參數量大幅降低。

Method	Sampler	Main	Frequent	Common	Rare
Stage 1 Training	U1	65.40%	86.64%	51.10%	2.22%
	U10	73.76%	90.68%	72.93%	31.20%
Stage 2 Trained on Feature	B	79.20%	90.53%	78.50%	45.04%
	R	80.16%	88.53%	79.98%	53.19%
Stage 2 Trained on MLP	B	79.49%	90.27%	78.78%	48.43%
	R	79.42%	88.92%	78.82%	**58.93%**
Stage 2 Trained on Both	B	77.12%	90.02%	77.00%	38.22%
	R	79.30%	86.39%	79.34%	55.35%

Table 5: Result of decoupling. (U1: Uniform sampler(Epoch 1), U10: Uniform sampler(Epoch 10), B: Balanced sampler and R: Reversed sampler)

3.2.2 Bilateral-Branch Network

表格第二列是這個架構的結果，就數值看來，其實是比第一列原本用規則取樣當第一條分支的BBN原始架構來的好。我們也有做另外的實驗是將BBN的Backbone換成用

均衡取樣做對比，也就是第三、四列，但Main的準確率並沒有第二列來的高。另外若把Backbone從U10換成U1，其Common跟Rare的數值較高，但Frequent就掉蠻多的，推測是U10已經掉到區域最小值，後面的BBN提升的效果有限。

Backbone	Sampler	Main	Frequent	Common	Rare
U10	U/R	78.08%	89.67%	80.70%	26.40%
	B/R	**81.24%**	89.73%	81.78%	50.69%
B	U/R	72.40%	91.20%	62.08%	3.30%
	B/R	78.49%	77.25%	84.30%	53.90%
U1	B/R	78.63%	75.81%	**84.95%**	56.28%

Table 6: Result of Bilateral-Branch Network (U1: Uniform sampler(Epoch 1), U10: Uniform sampler(Epoch 10), B: Balanced sampler and R: Reversed sampler)

3.2.3 Test Time Augmentation and Model Ensemble

雖然BBN的數值在Main與Common的部分比Decouple都來的好，但單看Rare就沒有Decouple來的出色。為了結合兩者，最後測試的時候，我們做了多種嘗試，把表現較好的四個模型做ensemble以及加入test time augmentation，讓所有結果有顯著的提升。

Method	Backbone	Sampler	Main	Frequent	Common	Rare
Decoupling (2 on MLP)	Swin(U10)	R	79.42%	88.92%	78.82%	**58.93%**
Decoupling (2 on Both)	Swin(U10)	R	79.30%	86.39%	79.34%	55.35%
Decoupling (2 on MLP)	Resnest269	R	70.80%	70.09%	72.73%	59.37%
BBN	Swin (U1)	B/R	78.63%	75.81%	**84.95%**	56.28%
Test Time Augmentation + Ensemble			**85.53%**	88.07%	**85.04%**	**61.73%**

Table 7: Result of test time augmentation and model ensemble

References

[1] Liu, Ze, et al. "Swin transformer: Hierarchical vision transformer using shifted windows." *arXiv preprint arXiv:2103.14030* (2021).

[2] Kang, Bingyi, et al. "Decoupling representation and classifier for long-tailed recognition." *arXiv preprint arXiv:1910.09217* (2019).

[3] Zhou, Boyan, et al. "BBN: Bilateral-branch network with cumulative learning for long-tailed visual recognition." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2020.