Machine Learning Final Project Report

- Whale

我完全沒沒沒想法

R06921038

B02901006

R06921077

B03502059

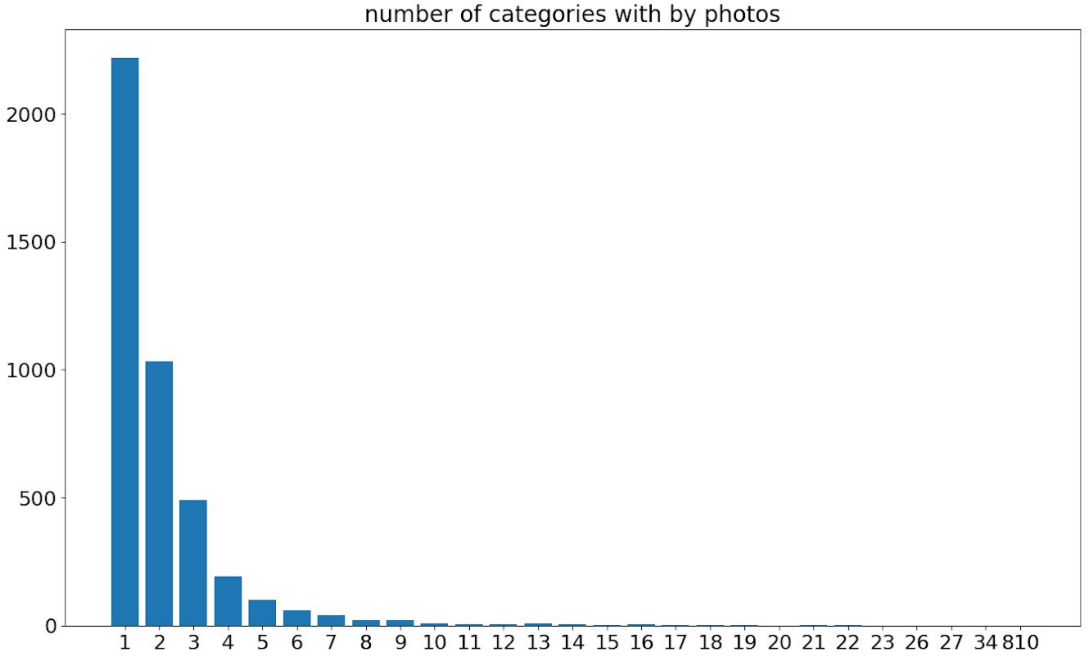
1. Introduction & Motivation : 1%

這次的目標是將鯨魚做分類，科學家決定利用機器學習的方式幫助他們分辨鯨魚尾鰭的照片是屬於何種鯨魚。在這次比賽中，我們必須建立一個演算法去分析Happy Whale的資料集，進而分辨鯨魚的種類。Happy Whale資料集是從各路對鯨魚有研究的專家那邊蒐集了大約有25,000張的相片，大部分的照片都只有包含鯨魚的尾鰭，我們決定用Convolution Neural Networks為主體的模型來達成我們的目標。

kaggle的評分方式為Mean Average Precision @ 5 (MAP@5)，對每張testing image都預測五個categories，再取每張照片的MAP做平均得到分數，簡而言之，把正確答案排在越前面分數會越高。

2. Data Preprocessing/Feature Engineering : 2%

初步觀察整個dataset，training data總共有9850張相片，testing data總共有15610張相片，training dataset總共有4251個categories。我們察覺training dataset只有9850張相片，卻有4251個categories，有點特別，所以決定先看一下各個categories有幾張相片，我們做了一張長條圖來觀察各類別有的資料數的分配：



- 有將近2300個categories只有一張圖片，大部分的categories只有不到十張圖片。
- 有一個category有810張相片，這個是new_whale類別。
- 在這個報告後面會對這兩個情況做更深入的探討與介紹解決的方式。

我們也發現到training dataset和testing dataset中，灰階的圖片大概佔了47%，彩色的圖片大概佔了53%，非常接近，我們就沒有對圖片做特別的處理，就單純的把灰階圖片讀成RGB的格式，彩色圖片就維持原本的RGB格式。

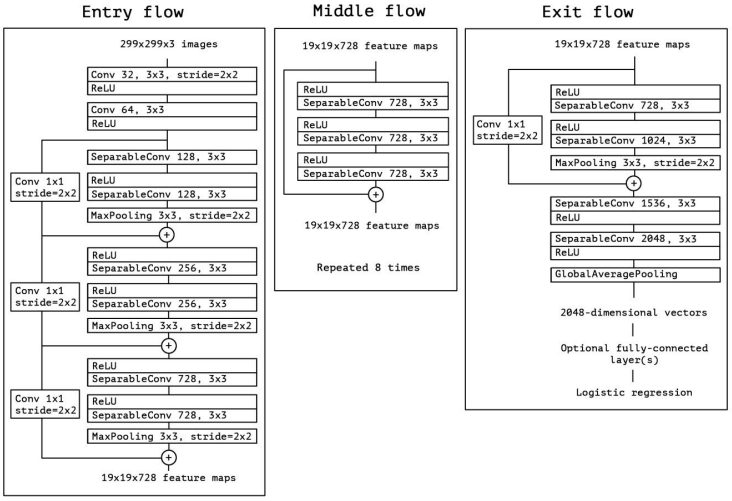
將所有的圖片resize成相同的大小（224*224)，然後將所有的圖片轉成RGB的格式。在訓練時把所有的圖片轉成(224,224,3)的陣列，對所有像素做除以255.的標準化，使其介於0~1之間。最後將每張圖片對應到的鯨魚做0,1,2,3,...的label，進行one-hot encoding轉換。

此外我們也做了大量的數據擴增(Data Augmentation)，觀察原始圖片，我們選擇對圖片做隨機的水平翻轉，隨機旋轉(rotaion range : 30度)，以及隨機的縮放(scale : 0.1)。

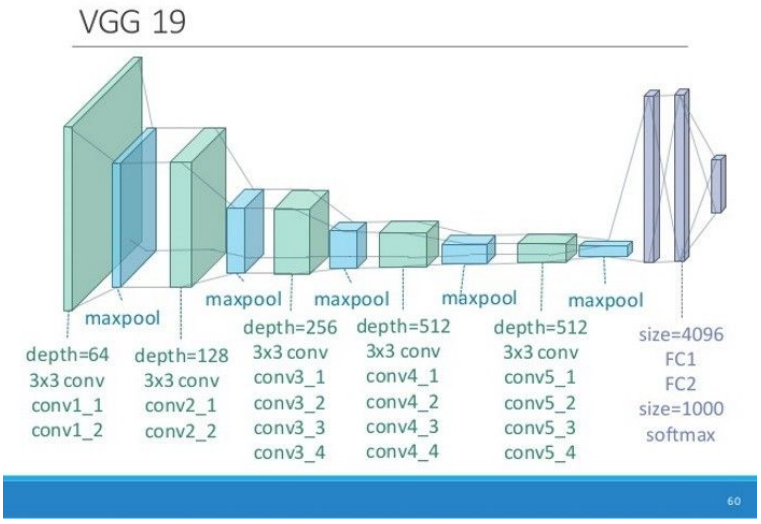
3. Model Description (At least two different models) : 4%

這次的project，我們總共使用了以下四種pretrained model做fine-tune:

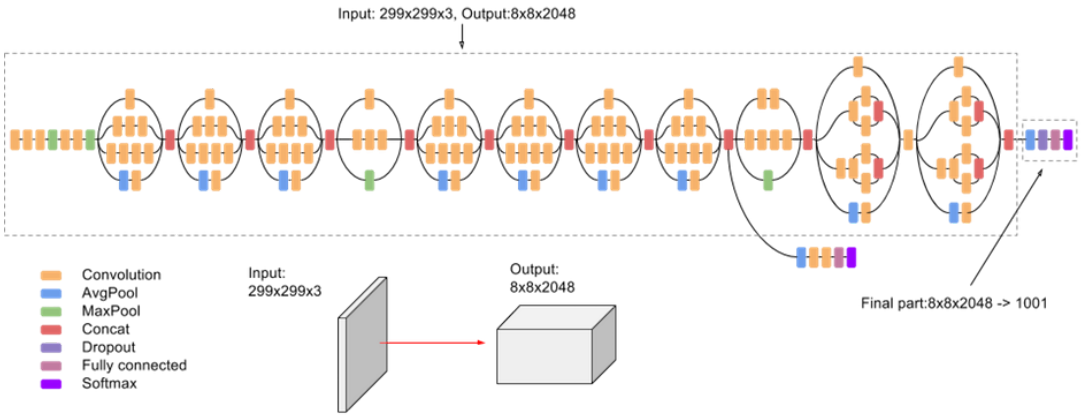
1.Xception



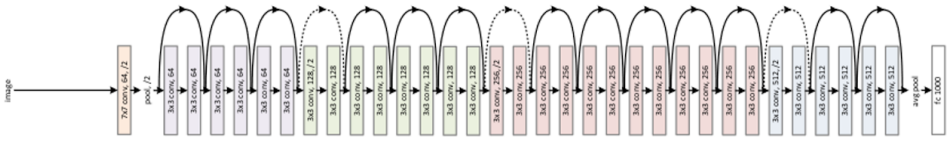
2.VGG19



3.InceptionV3



4.ResNet50



模型結構如上圖所示，對於這四種model，我們的weight都是使用imagenet。首先，我們保留了所有 Conv 和 pooling 層，拿掉最後的fully-connected layer和1000個 class 的softmax layer，另外加上新的fully-connected layer和4250(whale的類別數)個class的softmax classification output layer layer。

接著我們先凍結前面的所有層數，使用rmsprop optimizer只訓練我們另外加的fully-connected layer。經過幾個epochs之後，根據每一個model的結構，逐漸解凍最後面的幾層convolutional layers，使用SGD optimizer以非常小的learning rate去訓練調整。使用這個方法，慢慢地解凍更多的層數，隨著訓練的層數增加，model的accuracy也越來越高，一直訓練直到結果變差為止。

這次的實驗，我們總共使用了以下六個fine-tune後的model做ensemble:

model	basemodel	description
1	Xception	做三次fine tune，output 4251 class(最初訓練的model，包含new_whale，但predict時將new_whale類別拿掉，剩下4250個class。使用後面第四部份討論的設定threshold方法預測new_whale)
2	Xception	做四次fine tune，output 4251 class(最初訓練的model，包含new_whale，但predict時將new_whale類別拿掉，剩下4250個class。使用後面第四部份討論的設定threshold方法預測new_whale)
3	Xception	做三次fine tune，output 4250 class(不包含new_whale)
4	InceptionV3	做兩次fine tune，output 4250 class(不包含new_whale)
5	VGG19	做兩次fine tune，output 4250 class(不包含new_whale)
6	ResNet50	只訓練FC layer(transfer learning)，output 4250 class(不包含new_whale)

4. Experiment and Discussion : 6%

針對這次的project，我們把它當成分類問題來著手。一開始做的時候，我們依照原始資料的標籤，將所有的鯨魚分類，而標籤為new_whale的全部歸為一類，總計4251類，使用pretrained model去做fine-tune訓練。

然而顯然的，最後效果並不是太好。於是我們做了一些分析，參考原始競賽的描述："*Whales that are not predicted to be one of the labels in the training data should be labeled as new_whale.*"，我們推測可能的原因是因為new_whale裡面的800多張鯨魚中，包含了許多不同的鯨魚，只是恰好都不在標籤過的類別中，因此並不能將他們視為一類去訓練。另外，參考競賽提供的sample_submission.csv，全部優先預測new_whale，已經有高達0.32786的分數，表示testing數據集中應該具有相當數量的new_whale。因此我們思考之後，擬定了新的策略：

1.訓練時拿掉所有的new_whale，只針對已知的鯨魚去做分類訓練

2.設定一個閾值，當最後預測的結果中，所有類別的機率都低於這個閾值，也就是這隻鯨魚很可能不屬於任何已知的類別，我們就將他判定為new_whale。

在改變了方法之後，比較結果如下(以Xception模型為例)：

	Kaggle 分數
拿掉new_whale前	0.40094
拿掉new_whale後	0.44447

(strong baseline:0.435)

可以發現有很大的進步，因此我們認為這是這次project中，我們所作的最關鍵的一個步驟。

在這之後，我們嘗試使用了各種不同的模型去做實驗，希望能找到效果最好的模型。我們嘗試了四種不同的model去做fine-tune:

Xception, VGG19, InceptionV3, ResNet50。

fine-tune的過程中，可以發現在一定的層數範圍中，訓練的層數增加時，accuracy大致上也會跟著上升。但當訓練到太前面的層數或是訓練太多epochs，可能會破壞原先預訓練模型的結構特徵，而導致表現下降。因此，針對不同的model，我們也必須不斷的嘗試調整，在表現變差前停止fine-tune。

不同的模型表現，比較結果如下：

model	Kaggle 分數
Xception	0.44447
VGG19	0.41685
InceptionV3	0.42151
ResNet50	0.45398

觀察發現雖然結果有好有壞，但大致上差異不大。

在無法進一步突破的情況下，於是我們就將之前訓練過的模型，拿去做ensemble。

一般而言，分類問題應該使用voting比較恰當，但由於最終的結果是要預測Top5，也就5個最可能的whale。因此我們決定將所有model的預測結果，對每一個class做平均，取平均後機率最高的5類別做預測，當然，如前面所述，我們有設定一個閾值，假如為超過這個值，就優先預測new_whale。我們嘗試過了加權平均以及其他的ensemble方法，最終還是選擇最簡單的算術平均(效果最好)。此外，由於每個model預測的準確率不一，我們也必須不斷調整的閾值，看大概低於多少才要預測為new_whale，以達到最好的預測結果。

另一個觀察到的有趣現象是，當ensemble的model數越多，調整後得到最佳結果的new_whale閾值越低。譬如只有一個model時(以最開始的Xception model為例)，threshold設定為0.72時的結果最好；三個model做ensemble時，threshold 設為0.65效果最佳；當使用6個models做ensemble時，threshold則是0.55。我們推測，一種合理的解釋是，當ensemble數量越多時，結果可能越準確，也因此可能超過一定的機率，就有機會猜到正確的鯨魚類別。

number of ensemble models	Kaggle 分數
3	0.46484
5	0.47497
6	0.48953

Ensemble的結果如上表所示，比起單一個model，又進步了不少。經過反覆的嘗試後，我們選擇了其中的6個model做ensemble，Kaggle分數達到0.48953。

最後，由於我們的ensemble是經由平均的方式實現，為了避免一些極端的預測值拉低平均，或者多數model預測同一隻鯨魚，卻因為未超過threshold而被捨棄掉的情況，我們選了其中四個表現較好的model先做一次voting，當有三個model預測為同一隻鯨魚時，我們就優先預測那隻鯨魚。結果如下:

	Kaggle 分數
未做voting	0.48953
先做voting	0.49923

訓練參數:

-optimizer & learning rate :

fully connected layer : rmsprop(lr=0.001)

fine_tune convolutional layer: sgd(lr=0.0001, momentum=0.9)

-batch size :

vgg19 : 64

Xception, InceptionV3, ResNet50 :128

5. Conclusion : 1%

這次project的dataset非常特別，是我們在六次作業中完全沒有遇過的，訓練資料非常缺乏，我們只能從僅有的資料做data augmentation來得到更多的training data。在標籤處理上也很獨特，new_whale的出現讓我們必須額外考慮更多可能。

準確度提昇的過程中，最大的兩個突破點就是拿掉new_whale跟拿多個模型做ensemble，這兩個步驟是這次project中最不可或缺的兩個要素。在這次期末專題中也讓我們學習到了更多應對資料的方法：如何去處理資料、對不同類型的資料做不同的規劃、遇到瓶頸該如何解決等等，收穫良多。

6. Reference : 1%

1.https://keras.io/applications/

2.https://www.kaggle.com/c/whale-categorization-playground/discussion

3.https://www.kaggle.com/keras/resnet50

4.https://cloud.google.com/tpu/docs/inception-v3-advanced

5.https://ithelp.ithome.com.tw/articles/10192162

6.https://blog.csdn.net/KangRoger/article/details/69929915