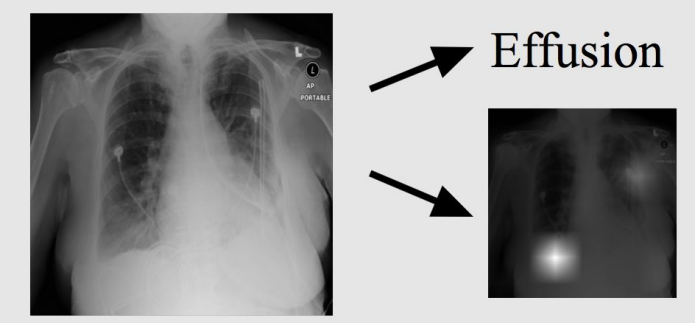
[ADL--final project report]

- 題目:

  Weakly Supervised Learning for Findings Detection in Medical Images(HTC)

- 問題描述：

  在這次的project，我們需要使用weakly supervised learning的方法，對圖像進行mulitlabels的分類，並且進行圖像語義分割，標記出bounding box。我們的訓練資料來自HTC提供的training data，共有約12萬張的Ｘ光片，每張Ｘ片可能對應到多種疾病，即有多個class labels（最多四個），但也可能該Ｘ光片中沒有發現疾病，即no finding的情況，其中可能出現的class labels共有九種（包含no finding為一種），所以我們的任務主要有二，如圖一所示，第一個是讀取Ｘ光片後，判別可能具有哪些疾病，預測出相應的class labels，第二個則是根據前面的預測，分別去標示出每種疾病的病徵位在Ｘ光片中的哪個區域，輸出相應的bounding box。



（圖一）

- 文獻探討：

  相對於strong supervision，weakly supervised learning僅需圖片分類就可以訓練，不需耗費大量的時間和專家人力去標記出bounding box，對於訓練資料的要求較少，因此可以收集到大量的低成本資料來進行訓練。

  關於weakly supervised learning的實作方法，主要可以分為兩個階段，在第一階段中，主要的任務是圖像辨識，目前，基於CNN的分類器常被用於訓練大量圖片的圖像識別任務中，有許多著名的公開pre-trained model架構，如：VGG19、GoogLeNet、ResNet等，透過對於空間的卷積過濾器來取出圖片中的特徵，並用以訓練出分類器，最後使用CNN模型的最後一層輸出來做inference，但這種方法只能做出圖像辨識，偵測物體是否出現，無法進一步去定位物體出現區域。

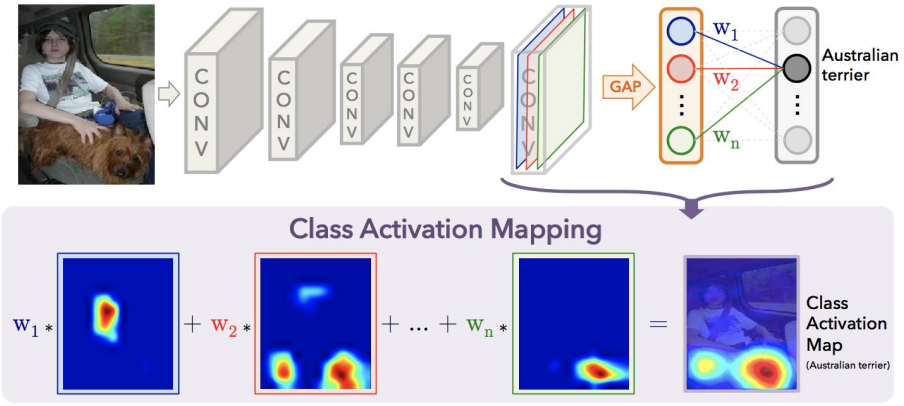
因此在第二階段中，主要的任務就是定位，在<<Weakly Supervised Localization using Deep Feature Maps>>這篇論文中，便希望利用空間和語意資訊之間的關聯以及卷積過濾器的特性，透過訓練好的分類器來標示出bounding box，作者認為CNN模型之所以能正確分類，必然是圖片在卷積過濾的過程中看到了該物體相應的特徵出現，所以利用相同的概念，每次鎖定圖片中的某一區域，重新resize成原圖大小後，再丟入分類器中，看是否能正確分類以及分類出的分數，再藉由beam search的方法來找出分類分數最高的區域，即應是該物體出現的區域。

<<Grad-CAM: Why did you say that?Visual Explanations from Deep Networks via Gradient-based Localization>>這篇論文則是提出一個新的演算法Gradient-weighted Class Activation Mapping (Grad-CAM)，作者希望藉此來增加CNN模型的透明度，增加輸入的資料中對於模型分類「較重要區域」的可視化， 讓人們可以探索機器在決策時的焦點，方法是使用各分類類別在CNN模型最後一層卷積層輸入的梯度資訊來建立關於圖片中重要區域的概略定位圖。Grad-CAM演算法是由Class Activation Mapping (CAM)衍生而來，CAM只適用於一小部分的CNN模型，而Grad-CAM則可以廣泛應用於任何基於CNN模型的架構，處理全連結層和更複雜的結構，這使我們可以將此演算法應用在我們選擇的pre-trained model架構上，透過找出決策焦點的方式來標出所要的bounding box。

而在<<Learning Deep Features for Discriminative Localization>>這篇論文則提到已有研究指出，CNN模型中各層的convolutional units其實都可以視為一個物體偵測器，即使我們並沒有提供任何關於物體位置的標示資料，但儘管在卷積層時的定位能力很強大，在使用全連結層來進行分類後，這樣的能力就會被破壞掉，所以現在已有人提出新的fully-convolutional模型，如：Network in Network (NIN)和GoogLeNet等，藉由避免使用全連結層來最小化參數數量，以及保持著高性能和前述的定位能力。為了達成這一目的，實作的方法是使用global average pooling作為structural regularizer，避免在訓練時overfitting，在作者的實驗中，發現使用global average pooling的好處不只在於作為一個regularizer，實際上只要稍作調整後，它就能保持卷積層原本的定位能力直到最後一層，同時作者也證明這樣的調整方法具有良好的轉移能力，可以應用到不同的資料集上進行一般性的分類、定位。

作者也提出生成class activation maps (CAM)的方法，使用類似於Network in Network和GoogLeNet的網絡體系結構，即網絡主要由卷積層組成，並且在最終輸出層（在分類情況下為softmax）之前，在卷積特徵映射上使用global average pooling，並將其用作產生所需輸出（分類或其他）的完全連接層的特徵，如圖二所示，藉由這樣簡單的連接結構，就可以通過將輸出層的權重投影到卷積特徵映射上來識別圖像區域的重要性，作者便將這樣的技術稱為

class activation mapping。



（圖二）

關於Global average pooling (GAP)和global max pooling (GMP)的比較，作者認為GAP鼓勵網絡去指出物體的範圍，而相對的，GMP則只鼓勵網絡去指出物體的某一個discriminative part，這是因為在做map的平均時，通過查找該物體所有的discriminative parts將可以最大化平均值，因為所有的low activations對於特定map的輸出都減少了，另外，在相同的分類能力下，GAP也能比GMP具有更好的定位能力。

- 實驗方法:

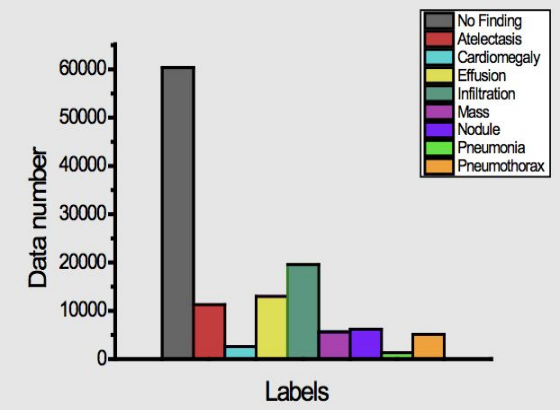
1.圖片預處理：

我們發現每張圖片的維度不同，有(1024,1024,3)，也有(1024,1024,4)，所以我們都先把圖片的第三個維度經由函數計算轉成1維，再重新resize為(224,224)，以符合我們選擇的pre-train model需要的輸入。

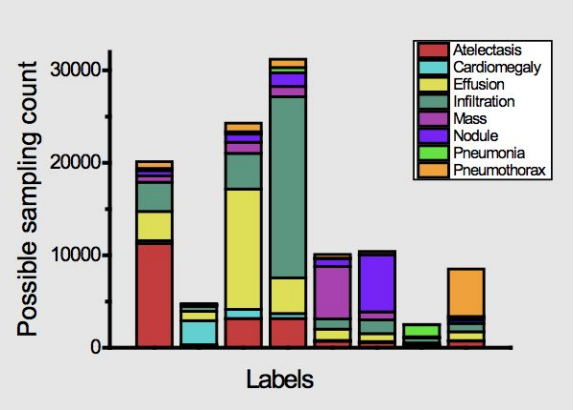
2.挑選訓練樣本：

我們發現訓練資料各類別的比例有很大的差異，其中no finding就佔了60000張，由圖三我們可看出no finding總數比其他各類加起來還多，所以分類器只要將每張圖片都預測為no finding就能有76%的準確率，但這樣模型顯然並沒有學到任何東西，這導致我們一開始訓練相當困難。在發現這個情況後，我們嘗試去掉no finding的資料來訓練，然而這樣訓練資料就少了一大半，而且問題仍然沒有解決，因為除了no finding以外的八類，比例同樣相當不平均，各類別資料數量範圍在2000到13000之間，此外還有multilabel的問題，所以我們若採用隨機抽樣的方式來訓練就容易產生偏差，如圖四所示，各類別被抽樣出來的機率不同，這會造成模型在訓練時發生overfitting的情況，從而在validation時表現很差。

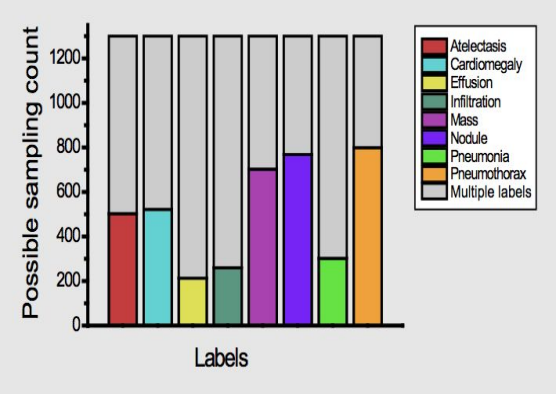
對於這些問題，我們嘗試過只挑選單一類別的資料來訓練，雖然有multilabel的資料量不多，但對訓練資料的數量仍有一定影響，也使得那些標為multilabel的資料我們模型都學習不到。所以我們最後採取的方式是保留multilabel的資料，但不使用隨機抽樣，而是在抽樣時讓各類別加起來的出現機率相同，如圖五所示，這個方法使得我們抽樣出來的訓練資料集中，各類別的可能出現次數會保持相同，我們希望藉此來解決訓練時產生的偏差，以及各類別資料量不均的問題。



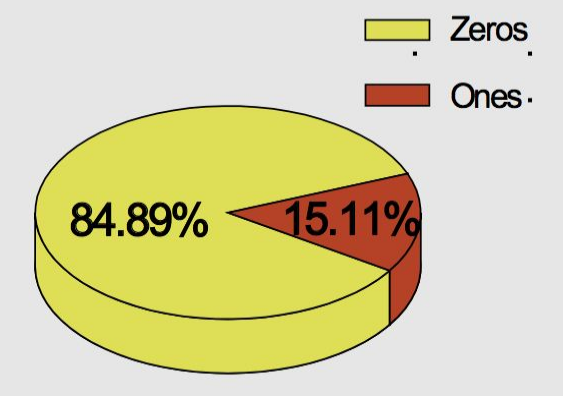
(圖三)



(圖四)



(圖五)



(圖六)

3.模型架構與訓練 ：

因為在訓練前，我們須先把label轉為one-hot vector的表達方式，但one-hot vector本身的特性使得預測出零就能輕易地降低loss，這將會讓分類器偏好預測出零，而不是1，如圖六所示，預測出零的機率大很多，導致我們很難訓練得起來，因此我們改為使用weighted cross-entropy來解決此一問題。

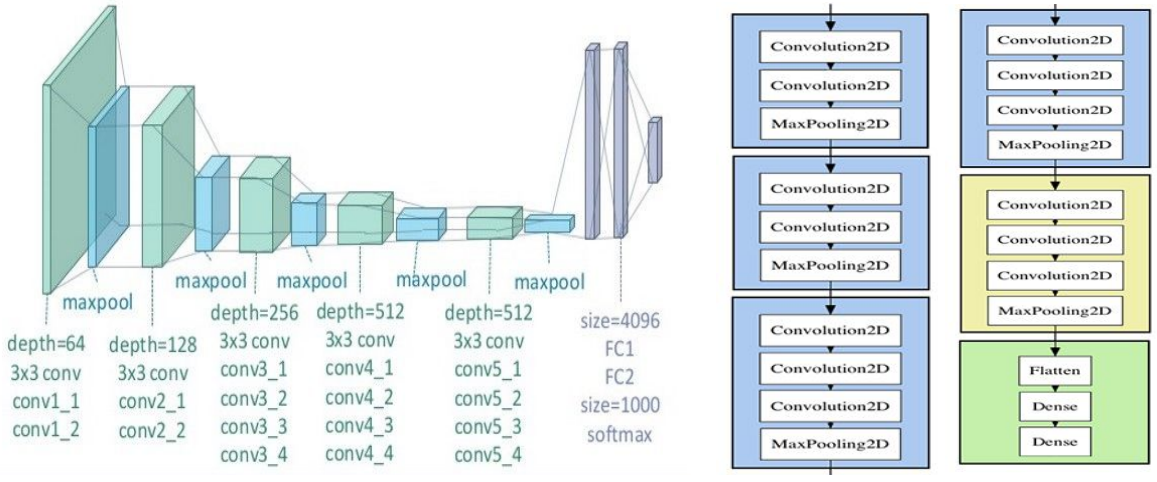
而在眾多的CNN模型架構中，我們選擇了VGG19，由於這些模型的隱藏層數都很多，也使用了大量的資料作訓練，如果重新訓練需要花很多時間，成果也不一定好，所以我們選擇直

接套用研發團隊事先訓練好的pre-trained model，讓我們可以站在巨人的肩膀上，加快訓練效率，而選擇VGG19的主要原因也是它的層數較少，繼續訓練需花費的時間也就較少。

VGG 是英國牛津大學 Visual Geometry Group 的縮寫，他們所提出的VGG模型在2014年的 ILSVRC localization and classification 两个問題上分別取得了第一名和第二名的成績，而我們採用的VGG19實際結構如圖七所示，可以看到前幾層皆為卷積和maxpool層的交替，每個卷積中又包含多個捲積層，這部分共有16層，最後再接三層全連接層，加起來總共是19層，其中激活函数使用的是Relu，並在訓練時做dropout，但不做LRN。

由於我們要辨識的內容不屬於原本團隊用來事先訓練的資料集，所以我們使用pre-trained model繼續訓練時需做些修改，首先，我們需要固定住前16層的conv層，保留原本的參數，即不改變原本抽取特徵的模型參數，只重新訓練最後三層全連接層來學習X光片的分類，此外，我們的預測結果需能產出mulitlabels，所以我們把原本最後一層輸出使用的softmax函數改為sigmoid，並且只要輸出的數值大於0.5我們就視為1，即屬於該類別，讓我們可以預測出multiclasses。

除此之外，我們也嘗試在VGG19之後連接不同的層，藉以找出分類的效果最好的模型，並且使用global average pooling作為structural regularizer，除了避免在訓練時overfitting，也希望保留下卷積層的定位能力，幫助我們後面bounding box的生成。



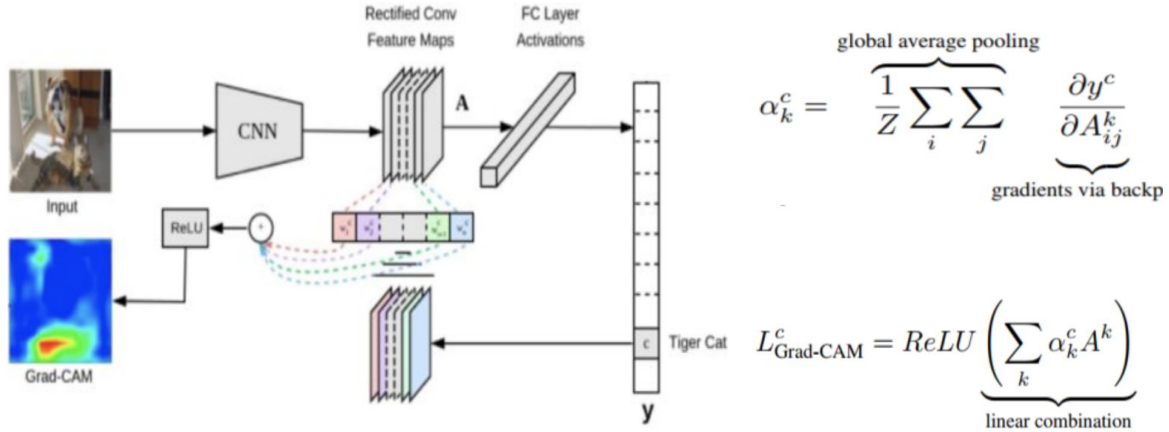
(圖七)

4. Bounding Box Generation ：

我們嘗試了兩種方法，首先是使用window sliding的方式切出圖片中多個候選的區域，並且建成搜尋樹，以加快搜尋效率，接著針對選擇的區域， 重新resize成原圖大小後，再丟入我們訓練出的分類器中，而分類的分數來自CNN模型最後一層輸出的數值，再利用beam search的方法找出分數最高的區域 ，但這種方法的缺點是，標示出的bounding box都是較大的區域

，若是設定的切分區域太小，難以分類，此方法就無法順利找出正確的區域，然而我們需要標示出的病徵必須精準，bounding box不能太大，所以我們最後轉為使用grad-cam演算法。

grad-cam演算法在實作上，主要包含了三個步驟，如圖八的示意圖，第一步驟是去計算類別y相應於最後一層卷積層的梯度，第二步驟則是做global-average-pooled(GAP)來獲得neuron importance weight，如圖八右上角的公式，而在第三步驟，使用圖八右下角的公式去計算在forward activation map上的weighted sum，接著加上ReLU後，再去做upsamlping，顯示在圖片上。

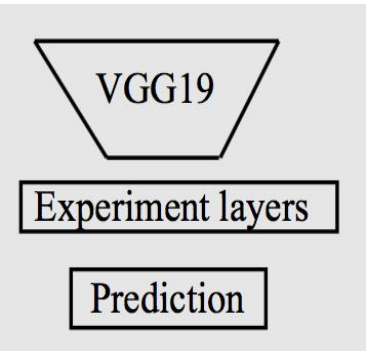


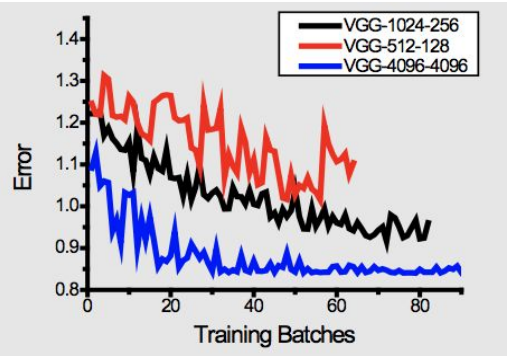
（圖八）

- 實驗結果：

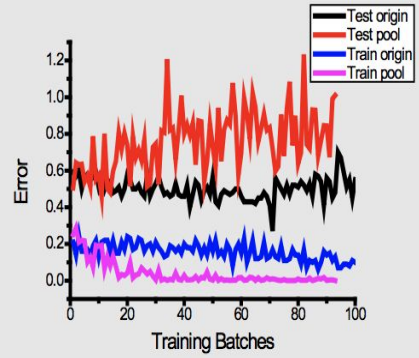
我們嘗試連接不同的層，如圖九所示的架構，試圖找出分類的效果最好的模型，而根據我們的實驗結果，沒有pooling data的話，training error幾乎不會下降，但如果有pooling data，training error又可以透過一直預測零而輕易地下降，如同前面圖六提過的問題，因此使用weighted cross-entropy變得很重要，如此可以幫助我們減少overfitting的問題。

而在test error部分，從圖十二可以看出下降的很緩慢，所以我們嘗試了許多方法來改進，試著在VGG抽取出特徵的最後一層pooling後接不同結構的層，圖十是我們的實驗結果，我們發現VGG-1024-256 是比較好的模型結構，可以看出training error呈現緩慢且穩定下降的趨勢，也因此我們發現在 VGG19模型和全連結層中間加入卷積層並不會顯著的增加預測的結果。

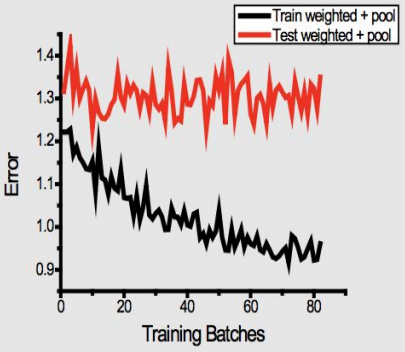




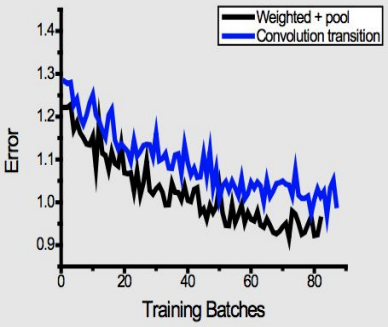
（左：圖九、右：圖十）



（圖十一）



（圖十二）



（圖十三）

- 結論與future work

在這次的final project，我們使用VGG19這一pre-trained model來從X光片中抽取特徵並進行分類，我們也嘗試了不同結構後，發現VGG-1024-256是表現比較好的模型，接著我們使用grad-cam演算法來標示bounding box，定位各類病徵出現在X光片中的哪個區域。在整個訓練過程中，訓練資料中各類別出現比例的巨大差異是我們面對的棘手問題之一，但我們藉由在抽樣時讓各類別加起來出現機率相同的方法，嘗試解決此情況帶來的訓練偏差問題，另外一個棘手問題則是如何預測出multiclass，以及偏好預測出零的情況，我們採取的方法是使用weighted cross-entropy鼓勵模型預測出壹。

而在未來，我們希望可以嘗試更多的pre-trained model，如前面提到過的GoogLeNet、ResNet等，比較它們之間的差異，增加我們的分類準確率，另外，我們也嘗試實作了Data Augmentation，在圖片資料上做幾何的轉換，改變圖片中像素的位置但保證特徵不變，如此就能在現有資料量的基礎上生出更多資料，我們希望可以增加到至少20萬筆的資料，若能使用更多的資料訓練，我們也就可以進一步減少overfitting的問題。

- 組員心得：

[郭勅君]

很高興這次能來參與HTC醫學影像辨識的競賽。以前總覺得機器學習就是找好模型，然後丟進去訓練就可以讓機器訓練的好了，頂多就是調調hyperparameter來讓模型更好。但經過這次競賽之後，我們遇到了很多以前沒遇過的事情，包括資料裡面類別的數量很不平衡，還有就是能訓練的資料量太少而且很容易讓模型overfitting。在處理這些實際會遇到的問題時，我們也用了很多以前沒想過的方法，但也嘗到了試了很多方法都無效的心情，在當下其實還挺崩潰的，還好有組員們可以一起討論，想想解決的方法，也因為其他組員也幫忙做了一些嘗試知道某些路行不通，也幫我在選擇方法的時候少走了很多冤枉路，所以還滿享受在這種一起討論做出東西的感覺。

[吳俊德]

相較於之前的其他作業，這次的project就真的比較接近真實狀況了。這些data和之前的作業不同的是這些data都是沒處理過的raw data，因此在實作的過程中不斷碰到不同的狀況，像是圖片上面會有醫師的註記、每個類別資料量不平均的問題。往往在如何解決這些問題上就花費了很多的心力。雖然最後的結果並不如預期的好，但是其實整體來說玩的還滿開心的。而且在這短短的幾個月之內從完全不懂深度學習的方法甚至對python一知半解，到現在可以藉由手上的資源嘗試實作並解決這些現實中會遇到的問題，這種進步的感覺很有成就感。

[吳中群]
很高興MLDS給我們這個機會讓我們可以參加HTC的比賽，因為平常作業可能都沒有像這次的影像辨識競賽有這麼實用性的題目，難度也是比以前做的題目難非常多，遇到了很多困難，像是怎麼訓練也訓練不起來，或是怎麼上傳都傳不上去，表現不如預期之類，從中思考是那邊出現問題，並努力去解決，謝謝各位好伙伴一起討論，學習怎麼解決問題的方法！

[韓宏光]
在這次final project，我實際接觸到了許多過去沒聽過的名詞，像是：VGG, Resnet, ...等，而這也是我第一次認真的去實作並比較論文提到的方法。而這次final project我認為跟往常的作業差最多的地方就在於這次的final project沒有提示，也沒有任何確保一定可以達到的performance，所有的一切都要依靠自己，不管train壞幾次都要想辦法看有哪些parameter去tune，還有哪些方法可以去改進。這讓我感覺到做研究需要有一顆堅毅的心：就算結果再不樂觀，都要相信是有一個可行的方案，只是自己現在沒找到而已。最後很感謝在這次final所有組員的幫助，這是我第一次接觸並實作深度學習，雖然最後沒有一個很令人驚艷的結果，但在這次嘗試中我學到了很多，不論是實作技巧或是論文閱讀以及正確的心態。

[黃于真]
藉由這次的final project，我們接觸到了平常難以接觸到的醫學領域，要嘗試對十幾萬張的X光片做分類和標記出bounding box，看著每個label的名詞都是那麼陌生，我們不是醫生，但透過這次的project卻彷彿在做著醫生的工作，只是換種方式，透過圖像辨識的方式來讓機器學習判讀X光片中有哪些疾病。而這次project很特別的一部份是需要採用weakly supervised learning的方法，和以往的作業有很大的不同，一方面覺得有趣，但另一方面也可以預想到它的困難，在訓練過程中，我們遇到了許多次瓶頸，怎麼訓練也訓練不起來，努力去思考可能是哪裡有問題，發現問題了，也要想辦法找出可行的方法解決，雖然有很崩潰的時候，但也學到很多，學習其他人怎麼發現問題，也學習如何嘗試找出解決方法。

- 參考資料 :

https://arxiv.org/pdf/1705.02315.pdf
https://www.microsoft.com/en-us/research/wp-content/uploads/2017/10/1793.pdf
https://arxiv.org/abs/1603.00489
http://cnnlocalization.csail.mit.edu/Zhou_Learning_Deep_Features_CVPR_2016_paper.pdf
https://arxiv.org/abs/1610.02391