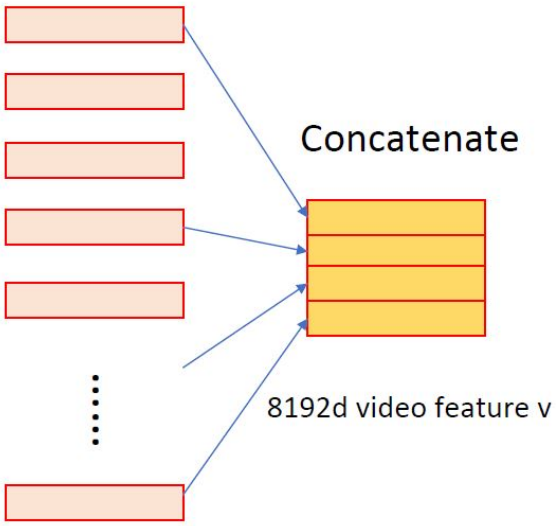
Please use this report template, and upload it in the PDF format. Reports in other format will result in ZERO point. Reports written in either Chinese or English is acceptable. The length of your report should NOT exceed 8 pages.

Name: 林志皓　Dep.:電機大三　Student ID:B04901069

## [Problem1]

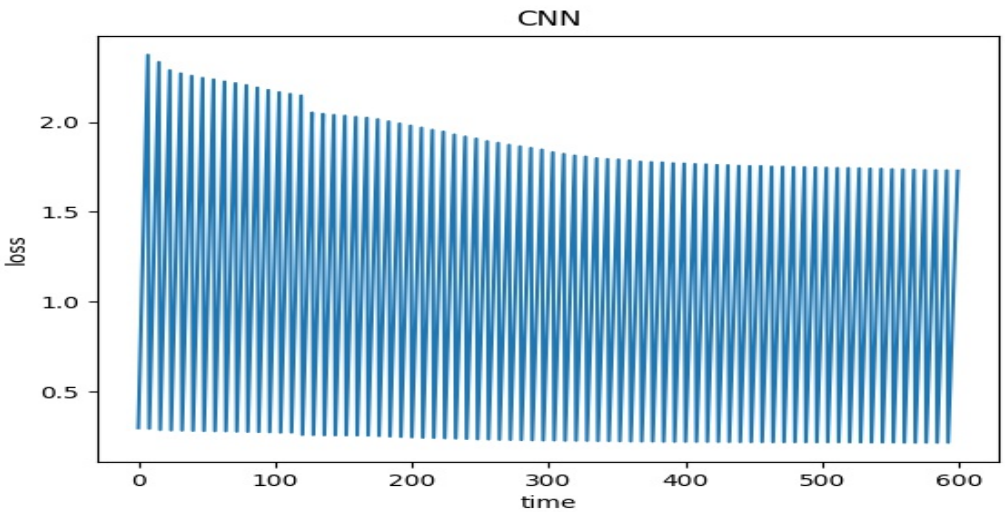
1. (5%) Describe your strategies of extracting CNN-based video features, training the model and other implementation details.

我用的 pretrained model 是 VGG16，用於提取特徵。而我的 strategy 是在每一部短片中，平均在 4 個時間點取出一張 frame，共 4 張，用 VGG16 提取特徵後，形狀各是(512,7,10)，接著把這四張疊加在一起，就變成了每一部影片餵進 classifier 的 input，此方式較為接近作業 pdf 中的第二種作法。




而 classifier model 的部分，我使用 3 層 Full-connected layer，中間有加 Drop out layer。Training 的過程就是很樸實無華的以 batch size = 10 餵給 model，optimizer 的部分使用 SGD(lr = 1e-3)進行訓練。

2. (15%) Report your video recognition performance using CNN-based video features and plot the learning curve of your model.



上圖是 CNN Based 方法的 learning curve，表示在 training 過程中 loss 的下降，由於 model 不深，每個 epoch 都能滿快訓練結束，因此我總共訓練了約 70 epochs，loss 可以看出有隨時間減少(劇烈上下起伏是因為我在同一個 epoch 中取了不同 batch 的 loss，因為資料分布的關係，同 epoch 中會有明顯差異)訓練到最後，在 validation set 上的 Accuracy 大概可以到 0.41 左右，滿接近第二題的 base line。

## [Problem2]

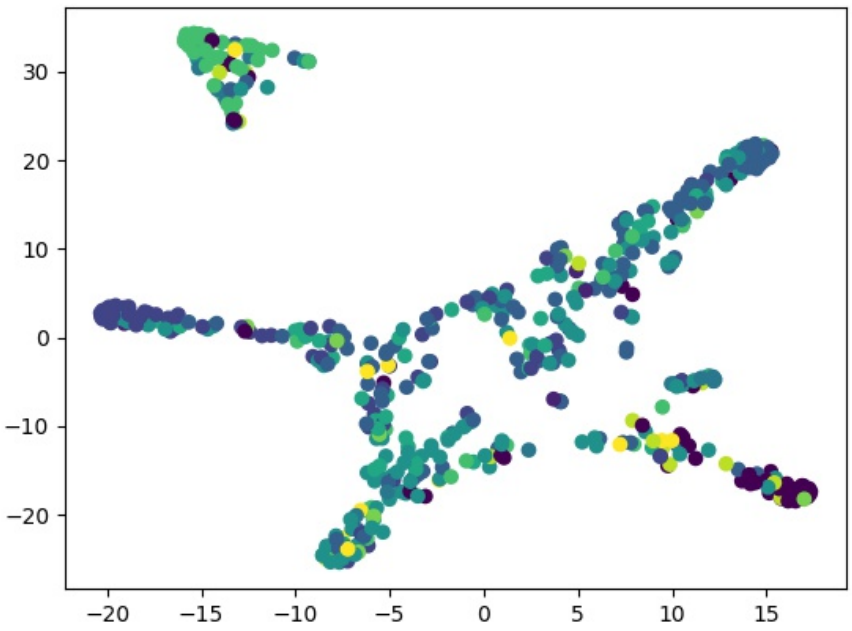
1. (5%) Describe your RNN models and implementation details for action recognition.

在提取特徵的 strategy 上，我每段影片平均選了 10 個時間點的影像，跟第一題相同，使用 pretrained VGG16 提取特徵，而為了縮小儲存容量，我把每張影像縮小 1/2 倍；然後把這些特徵疊加起來，之後用於 RNN 的訓練中，time step = 10。

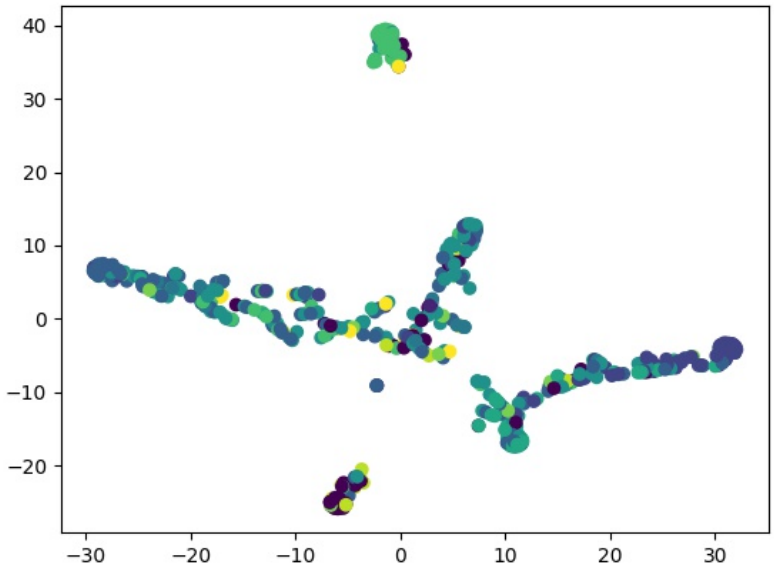
而 model 的部分，我只使用一層 LSTM，hidden size = 1024，其 output 接了兩層 Full-connected layer，最終的輸出為類別的預測。同樣因為不深，訓練速度十分迅速(我沒有去 fine tune 前面的 VGG16)，而 optimizer 我選擇使用 Adam(lr = 1e-4)，在訓練完約 20 epochs 後即有不錯的成果，在 validation set 上有達到 baseline 的要求。

2. (15%) Visualize CNN-based video features and RNN-based video features to 2D space (with tSNE). You need to generate two separate graphs and color them with respect to different action labels. Do you see any improvement for action recognition? Please explain your observation.

CNN



RNN



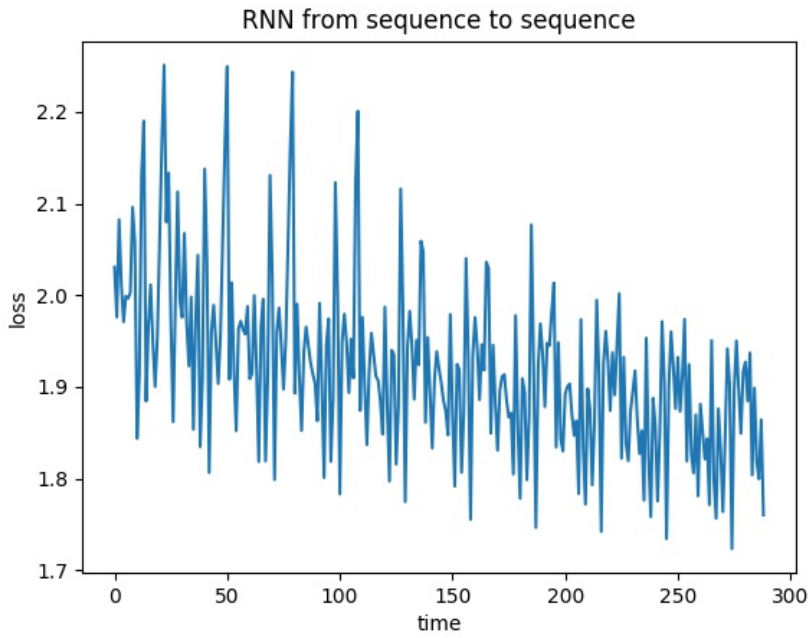

上下圖分別是 CNN、RNN 在 validation set 上的 TSNE 作圖，由結果可以看出 CNN 有些資料並未分得很好，還是集中於中間區域；而 RNN 相較之下表現較好，很多資料都分得開，區塊狀的情形更加明顯，也相對應於在 Accuracy 上，CNN 的是 0.41；而 RNN 的則是 0.456。可以看出 RNN 在這種有時續性的分類，效果是比 CNN 好的。

## [Problem3]

1. (5%) Describe any extension of your RNN models, training tricks, and post-processing techniques you used for temporal action segmentation.

基本上 model 是延續上一題的架構，提取特徵的方式也與上一題一樣。訓練的 strategy 我認為是這題最難的部分。由於每部影片都不一樣長，若要合成一樣的長度可以方便塞進同一個 batch，需要花費心思去設計如何延長或縮短某些影片。而又要符合 from sequence to sequence，所以也沒辦法像上一題依樣訓練一定 time step 後在 output prediction。而我選擇的 strategy 是，每處理一張影像的 feature 後，即把當下的預測結果與 ground truth 做比較，並立即計算 loss，也用 optmizer 更新 weights，然後把這次的 hidden vector (h_n, h_c)回傳給 model 進行下一次的訓練。也因此我這種方式的訓練速度相當緩慢，但是 loss 確實是有在隨時間減少的。

2. (10%) Report validation accuracy and plot the learning curve.




Among Validation set :

Accuracy of OP01-R03-BaconAndEggs = 0.564018691588785
Accuracy of OP02-R04-ContinentalBreakfast = 0.5085287846481876
Accuracy of OP03-R02-TurkeySandwich = 0.4364060676779463
Accuracy of OP05-R07-Pizza = 0.41656365883807167
Accuracy of OP06-R05-Cheesebuger = 0.5029411764705882

3. (10%) Choose one video from the 5 validation videos to visualize the best prediction result in comparison with the ground-truth scores in your report. Please make your figure clear and explain your visualization results. You need to plot at least 300 continuous frames (2.5 mins).

我選擇的是 OP01-R03-BaconAndEggs 的整部影片

Ground Truth

Prediction

上圖中的兩條色條，每種顏色相對應於一個動作，在這部影片的 prediction accuracy 大約是 0.56，算是還不錯。而進一步觀察，可以看出預測的結果以黃色、大紅、藍為最多也最為準確，我猜想原因是因為 training data 中各個類別的數量並不平均所致，以第一二題的 training set 為例，最多樣本數的類別，數量大概是最少類別的 10 倍，第三題的樣本也有類似不均的狀況。因此在訓練上可能很難讓 model 在這類別上能順利做出正確預測。上圖 ground truth 中粉紅色量很少，在下面的 prediction 中便幾乎沒看到；而最多的黃色，model 有做出還不錯的預測。

**[BONUS]**