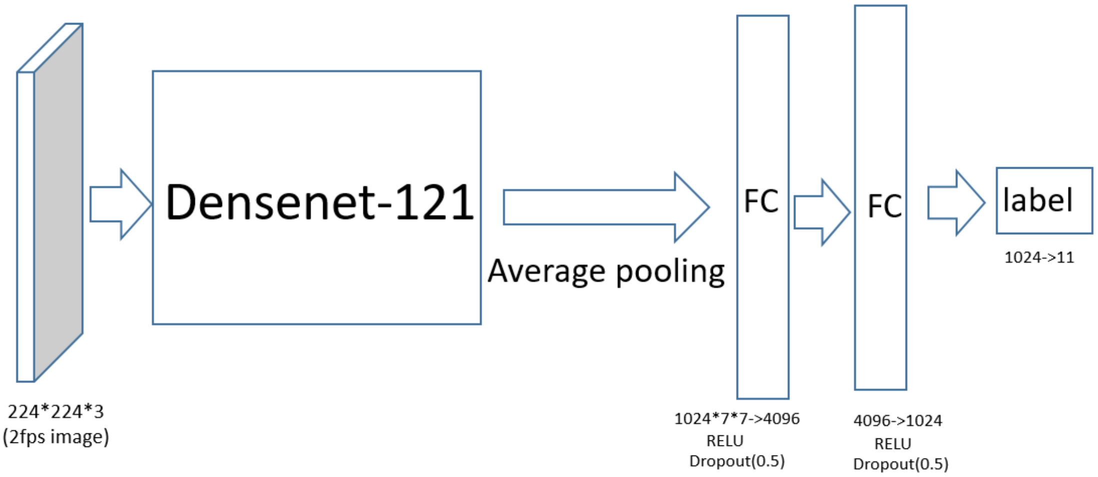
Name:許博閔 Dep:電信碩一 Student ID:R07942091

Problem 1 Data preprocessing

1. Describe your strategies of extracting CNN-based video features, training the model and other implementation details (which pretrained model) and plot your learning curve (The loss curve of training set is needed, others are optional)

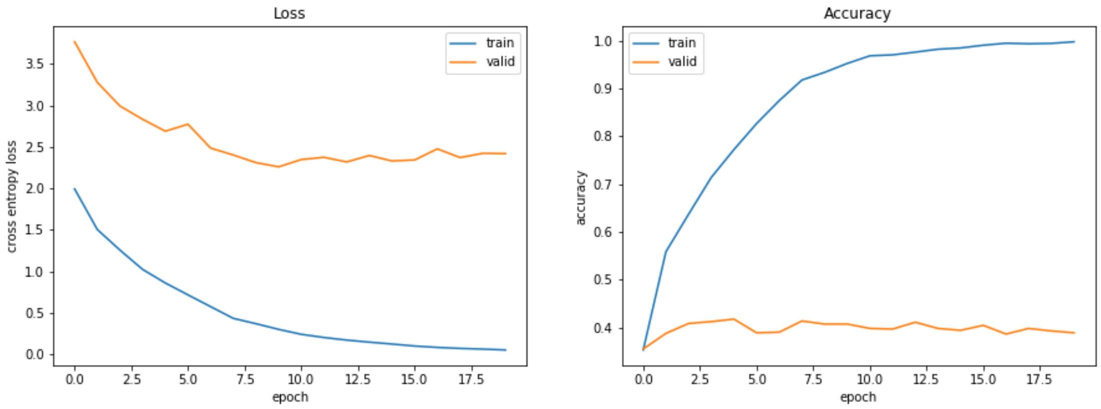
   我用助教提供的 reader.py 將影片 down sample 成 2fps，並用 torchvision transforms 將圖片 resize 成(224,244)，還有 normalize。我用 densenet121 為 pretrained model，將 model 輸出的結果 average pooling，得到維度 1024*7*7 的 feature，再通過 3 層的 fully connected layers 最後得到輸出結果，只有訓練 fully connected layers 的參數。

   model 架構 ：



   Epoch : 20, batch size = 64, Adam learning rate = 0.00005, loss : cross entropy

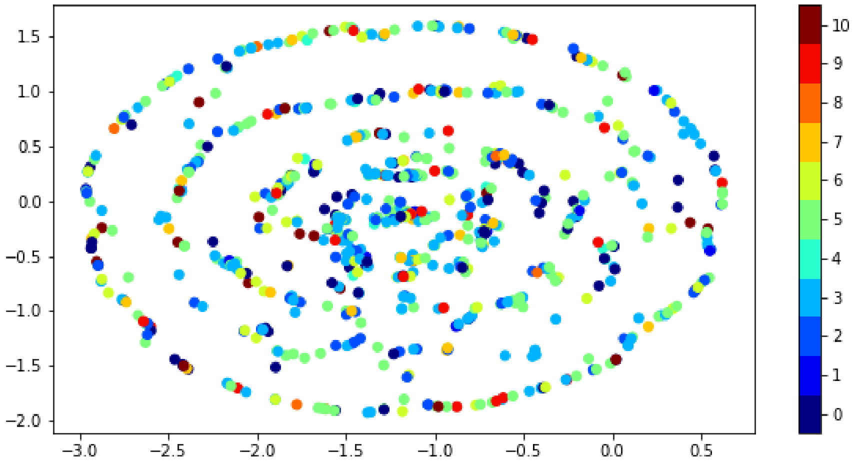
   Learning curve :



2. Report your video recognition performance (valid) using CNN-based video features and make your code reproduce this result

   Validation accuracy : 0.4395 (338/769)

3. Visualize CNN-based video features to 2D space (with tSNE) in your report. You need to color them with respect to different action labels.



Label : follow github

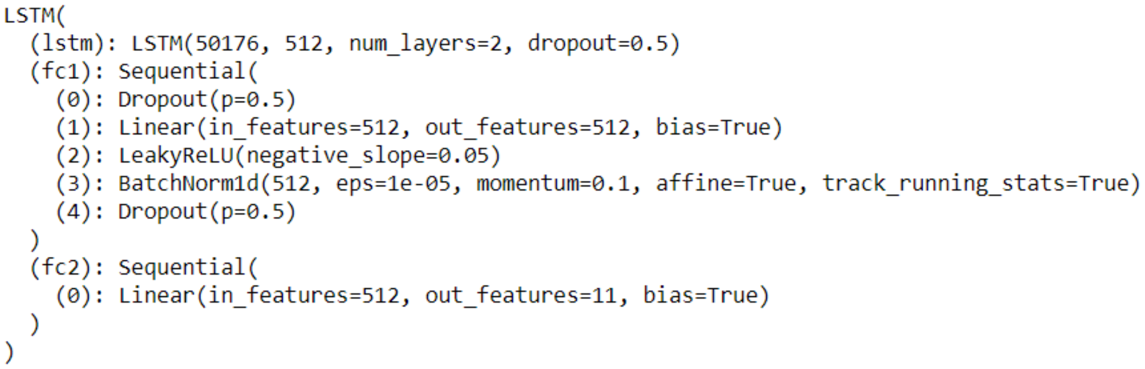
Label	0	1	2	3	4	5	6	7	8	9	10
Action	Other	Inspect/ Read	Open	Take	Cut	Put	Close	Move around	Divide/ Pull apart	Pour	Transfer

Problem2: Trimmed action recognition

1. Describe your RNN models and implementation details for action recognition and plot the learning curve of your model (The loss curve of training set is needed, others are optional).

   前處理的部分和第一題相同，但不做 average pooling，而是將 feature 依序放入 LSTM，有用 pad_sequence 使長度相同，我的 LSTM 為兩層、單向、hidden size = 512，之後將最後的 hidden state 經過 2 層 fully connected layers 最後得到輸出結果，LSTM 的 weight 用 orthogonal initialization。

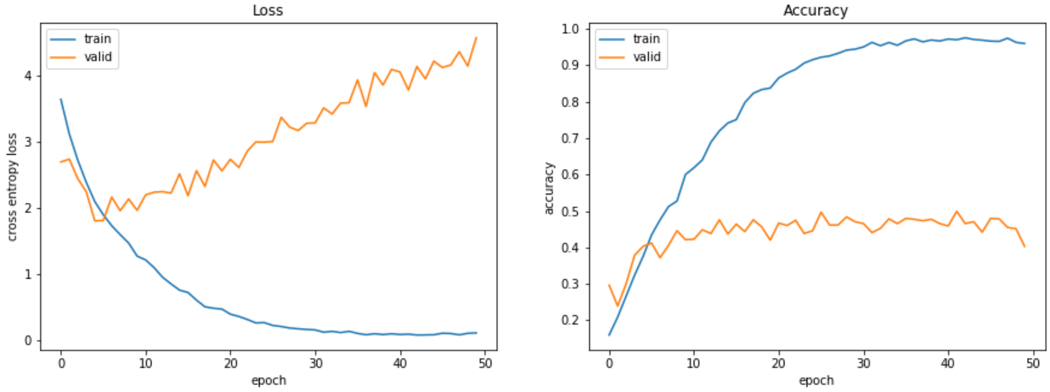
   model 架構 ：



```
LSTM(
  (lstm): LSTM(50176, 512, num_layers=2, dropout=0.5)
  (fc1): Sequential(
    (0): Dropout(p=0.5)
    (1): Linear(in_features=512, out_features=512, bias=True)
    (2): LeakyReLU(negative_slope=0.05)
    (3): BatchNorm1d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (4): Dropout(p=0.5)
  )
  (fc2): Sequential(
    (0): Linear(in_features=512, out_features=11, bias=True)
  )
)
```

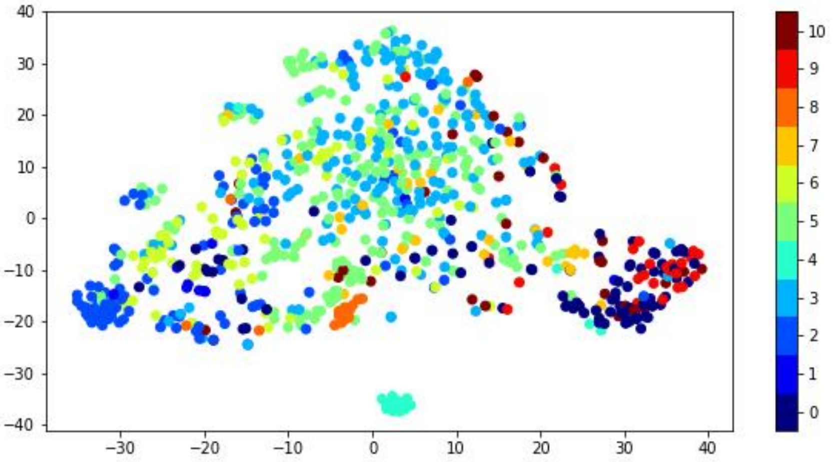
Epoch : 50, batch size = 64, Adam learning rate = 0.0001, loss : cross entropy

Learning curve :



2. Your model should pass the baseline (valid: 0.45 / test: 0.43)
   Validation accuracy : 0.4993 (384/769)

3. Visualize RNN-based video features to 2D space (with tSNE) in your report.
   You need to color them with respect to different action labels.
   Do you see any improvement for action recognition compared to CNN-based video features ? Why? Please explain your observation



Label : follow github

Label	0	1	2	3	4	5	6	7	8	9	10
Action	Other	Inspect/ Read	Open	Take	Cut	Put	Close	Move around	Divide/ Pull apart	Pour	Transfer

和 CNN feature 畫出來的圖相比，明顯是 RNN 的 action label 分的比較好，因為同一種類別的點靠的近，CNN 的結果看起來雜亂無章，但我認為這是因為 densenet 的 feature 維度有 50176，因此在降成 2 維的過程中失去過多的資訊，因此看不出有分群的感覺，但 RNN 的結果就有分群的感覺。

Problem3 Temporal action segmentation

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

   因每部影片長短不同，而且整段影片太長，直接用很可能使 model 訓練不起來，因此我將每部影片切成每 350 個 frame 為一單位的短影片 ，方便 model 的訓練，seq2seq 的 weight 用第二題的 model 來初始化參數，將每一個 hidden state 通過一層的 fully connected 得到每張 frame 的預測結果。

   預測 validation video 的時候則沒有特別去切割 frame，直接每次放一部完整的 video frame 來預測 label。

   Epoch : 20, batch size = 24, Adam learning rate = 0.0005, loss : cross entropy

2. Report validation accuracy in your report and make your code reproduce this result.

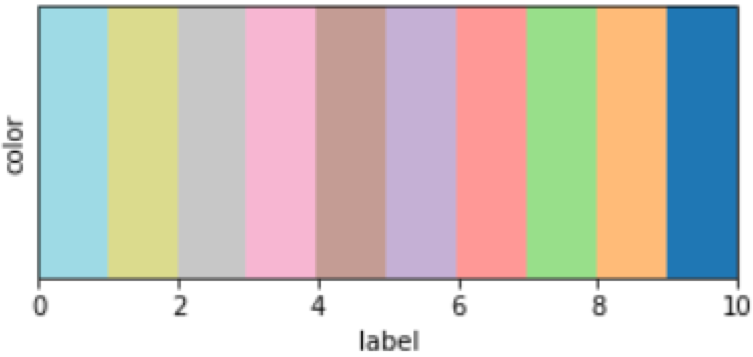
Video name	Accuracy
OP01-R02-TurkeySandwich	0.4476
OP01-R04-ContinentalBreakfast	0.5743
OP01-R07-Pizza	0.57799
OP03-R04-ContinentalBreakfast	0.5129
OP04-R04-ContinentalBreakfast	0.6193
OP05-R04-ContinentalBreakfast	0.4705
OP06-R03-BaconAndEggs	0.5886

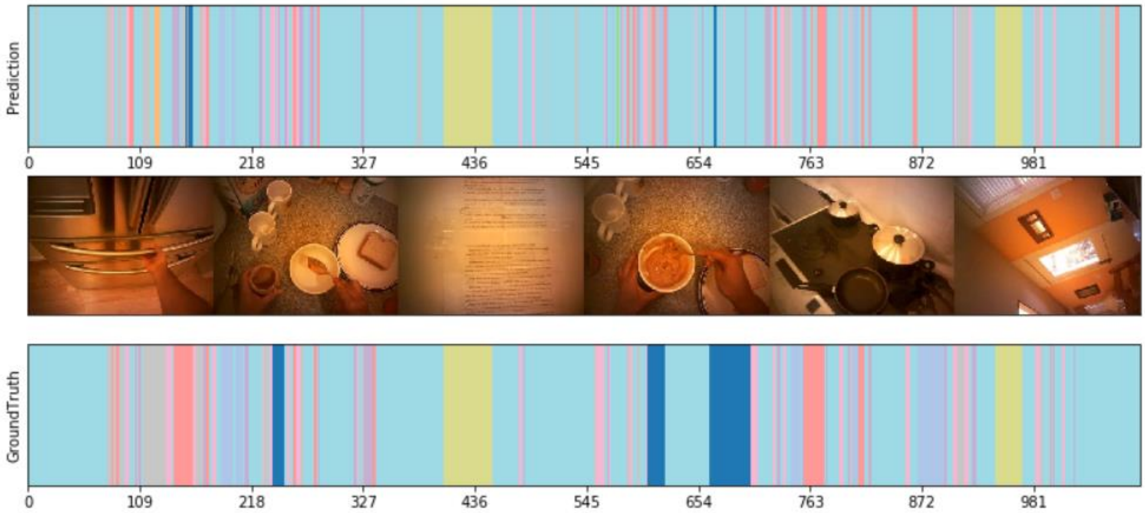
   Average accuracy : 0.5523 (4937/8938)

3. Choose one video from the 7 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 500 continuous frames)

   Video : OP04-R04-ContinentalBreakfast

Label	0	1	2	3	4	5	6	7	8	9	10
Action	Other	Inspect/ Read	Open	Take	Cut	Put	Close	Move around	Divide/ Pull apart	Pour	Transfer





a. 從結果來看，大多數預測正確的 frame 都是淡藍色的 label 0，這應該和 training data 的 ground true 有關，因為 label = 0 的 frame 偏多，因此 model 很容易預測 0 為答案。

b. 在 action label 變化劇烈的影片段落，如圖片中 100~250 的部分，預測的準確度比較差，或許是因為我用固定長度的 frame 來訓練 model，因此在變化快速的影片段落就預測不準。

c. 黃色的 label (Inspect,Read)做得特別好，我想這是因為這個動作和其他 action label 在視覺上有明顯的差異，因此準確度很高。

Reference

https://github.com/thtang/DLCV2018SPRING/tree/master/hw5