1. Model description

首先在資料的處理，我選擇把 caption 轉成 word based 的 one-hot encoding。 我計算了所有 caption 的長度平均為 7 ，因此在建立 vocabulary 我只取長度為 14 以下的 caption 去建立字典，其他長度大於 14 的句子並不會影響字典的建立，最後字典的 size 約莫為 5260。training label 只會在小於 14 個字的句子前後分別加上<BOS>, <EOS> 最後 padding 到 15 個字。

最後我的 training feature shape 為 (80, 4096)，training label shape 最後為 (15, 5260)。

```
Layer (type)                 Output Shape              Param #   Connected to
==================================================================================================
input_1 (InputLayer)         (None, None, 4096)        0
__________________________________________________________________________________________________
input_2 (InputLayer)         (None, None, 5916)        0
__________________________________________________________________________________________________
lstm_1 (LSTM)                [(None, 512), (None,      9439232   input_1[0][0]
__________________________________________________________________________________________________
lstm_2 (LSTM)                [(None, None, 512), (     13166592  input_2[0][0]
                                                                 lstm_1[0][1]
                                                                 lstm_1[0][2]
__________________________________________________________________________________________________
dense_1 (Dense)              (None, None, 5916)        3034908   lstm_2[0][0]
==================================================================================================
Total params: 25,640,732
Trainable params: 25,640,732
Non-trainable params: 0
```

我的架構是一個 LSTM 對 frame feature 做 encode，接著接下一層 LSTM 做對上一層輸出做 decode。並且我有採用 teacher forcing。

2. Attention mechanism

3. How to improve your performance

在一開始 training 我並沒有加上 teacher forcing，後來加上 teacher forcing。

```
encoder_input_data, decoder_input_data = parseData.readTraingFeature()
decoder_inputs_dim = decoder_input_data.shape[2]

decoder_target_data = np.zeros(
    (decoder_input_data.shape[0], decoder_input_data.shape[1], decoder_input_data.shape[2]))

for i, caption in enumerate(decoder_input_data):
    for t, word in enumerate(caption):
        if t < len(caption)-1:
            decoder_target_data[i][t][:] = decoder_input_data[i][t+1][:]
```

另外我沒有針對 training feature 去做改變，我覺得改變 training label 的樣貌或許多少能改善訓練效果。

4. Experimental results and settings

我主要先嘗試改 training label 的 shape，來自於 vocabulary 的建立而影響 one-hot encoding 的維度，實驗數據如下：

Training label shape	(15, 5260)	(10, 5916)
BLEU	0.105314275120976237	0.16601627550277254

Reference:

1. https://www.zhihu.com/question/36591394
2. https://github.com/spro/practical-pytorch/blob/master/seq2seq-translation/seq2seq-translation-batched.ipynb