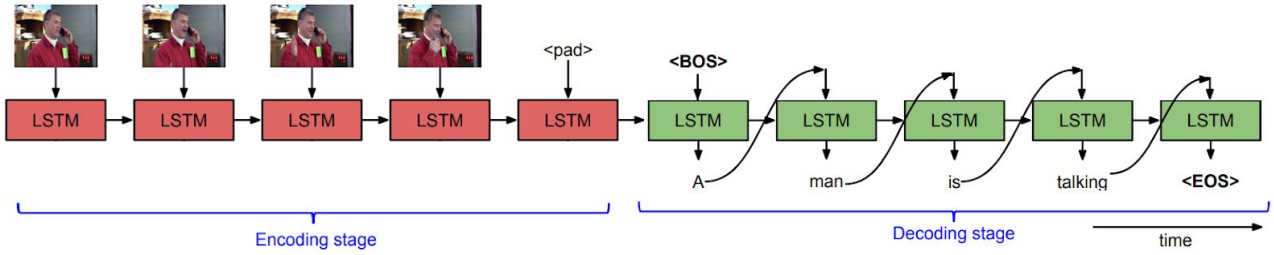
MLDS HW2-1

R06725007 賴冠廷、R06725015 李尚恩、R06725019 江孟軒

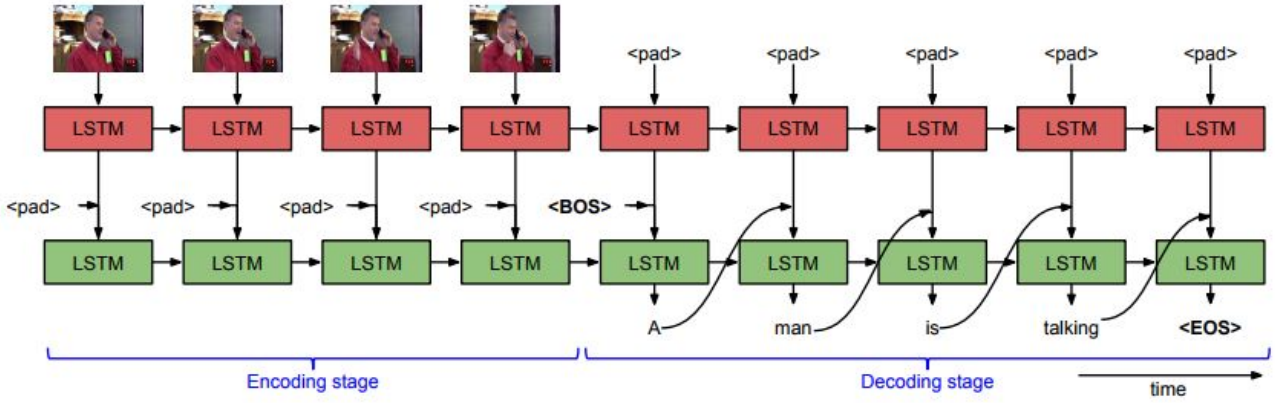
Model Description (3%)

1. Preprocessing: 在資料前處理的部份我們將所有的影片與其 caption 配對，組成若 21900 組 (video, caption) 的 pair 當成 training data。Vocab 的部分則是選用 caption 中出現次數大於 3 的詞。
2. Baseline model: 我們的 baseline 模型為一個單層的 Encoder-Decoder 的 Seq2seq 架構，示意圖如下



模型中 RNN 部分使用了 GRU, hidden dimension = 512, learning rate = 0.0005, optimizer 為 Adam, training epoch = 40 。其中 Decoder 的部分包含了一層 embedding layer 將輸入的字轉成向量表示，embedding dimension 為 512 。

3. S2VT model: 另外我們也實作了 S2VT 的模型，結構示意圖如下:



模型中 RNN 的預設參數和架構與 baseline 模型相同。下層的 GRU 會將上層 GRU 的 hidden state 與過 embedding 後的字詞合併(concat)作輸入。

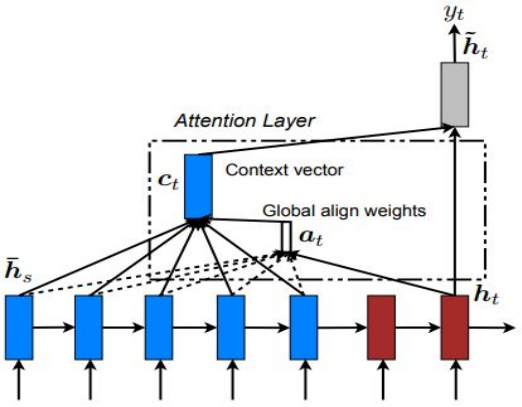
How to improve your performance (3%)

1. Global Attention: 為了讓模型能根據每個字去調整其注重的 video frame，我們實做了兩種 Global Attention ，示意圖如右下。

(a) *Bahdanau et al. model*

第一種方法使用了 Bahdanau et al.[1] 提出的 attention 方法，將 Decoder 上一個 timestep 的 hidden state 與 Encoder 所有的 hidden state 算出每個 state 的權重，計算方式如下：

$$score(h_{t-1}, h_s) = \begin{cases} h_{t-1} \cdot h_s \\ h_{t-1} \cdot W_a h_s \\ W_a[h_{t-1}; h_s] \end{cases}$$



其中 h_{t-1} 為上一個 timestep 的 Decoder hidden state, h_s 為 Encoder 所有的 hidden state, score 為一個計算 attention weight 的 function。計算方式有三種分別為*dot*, *bilinear, concat, bilinear* 與 *concat* 中的 *W* 為一層 Fully-Connected 的 NN。*bilinear* 的計算方式為將 h_s 經過 *W* 後再與 h_{t-1} 內積, *concat* 則是單純將 h_{t-1} 與 h_s 合併後再過 NN 來計算 weight。
計算出 attention weight 後將 h_s 做 weighted sum 獲得 context vector, 並將其與 Decoder input 的 word_embedding 合併和 h_{t-1} 作為這個 timestep 的 input。

(b) *Luong et al. model*

第二種方式參考了 Luong et al.[2] 的作法，計算 attention weight 的方式與上述相似。先將word_embedding 與上一個 timestep 的 h'_{t-1} 經過 GRU 後，用這個 timestep 的 hidden state 計算 attention weight, context vector, 與 h'_t。其 $\tilde{h}_t = tanh(W_c[c_t; h_t])$ 為 context vector 與 hidden state 合併後過一層 NN 再取 tanh 。每一個 timestep 都會將 h'_t， h_t 傳給下一個 timestep 當輸入。

2. Schedule Sampling & Beam Search: 為了避免 Overfitting我們實做了 Schedule Sampling ，其 teacher_forcing_ratio 使用 inverse sigmoid decay function[3] 讓其隨著 epoch 增加而遞減。在訓練過程中發現使用 Schedule Sampling 的模型 validation set 的 loss 相較於沒有使用的模型低。另外也嘗試了 Beam Search 讓模型可以考慮多種可能的預測路徑來產生最後的 sentence。

Experimental results and settings (1%)

我們將上述的方法分別套用到 baseline 模型與 S2VT 模型中，下表為其比較結果：
實驗設定：50 epoch, batch size: 64, 輸出文字最長長度: 21, vocab dictionary min count: 3, word embedding size: 512，並切 770 筆資料作為 validation set，使用的 schedule sampling 方法為 inverse sigmoid 去 decay forcing 的比率，k 值設為 10，輸入的 4096 維 frame 經過一層的 linear 轉換為 512 維再丟進 encoder 中。

model	bleu
Seq2Seq + schedule sampling(Baseline)	0.68443
S2VT + schedule sampling	0.68712
S2VT + attention(bilinear) + schedule sampling	0.71664
S2VT + attention(dot) + schedule sampling	0.72734

Reference

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv:1409.0473*
[2] Minh-Thang Luong, Hieu Pham, Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation
[3] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, Noam Shazeer. 2015. Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks
[4] Natsuda Laokulrat, Sang Phan, Noriki Nishida. 2016. Generating Video Description using Sequence-to-sequence Model with Temporal Attention

分工表

賴冠廷：S2VT + Bahdanau attention, beam search
李尚恩：Baseline model, beam search, Bahdanau and Luong attention
江孟軒：S2VT model, Report, Bahdanau and Luong attention