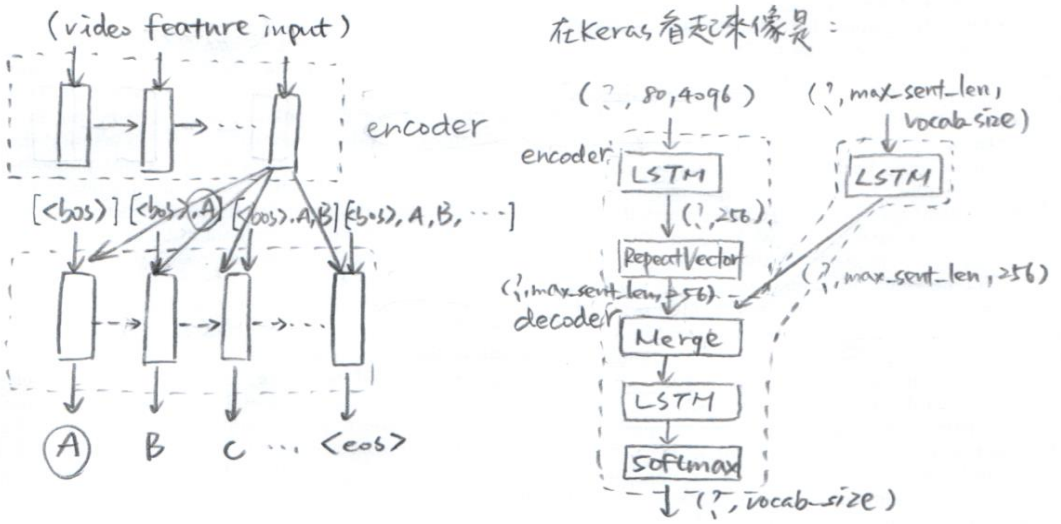
ADLxMLDS HW2: Video Captioning

學號：R06922030 系級： 資工碩一 姓名：傅敏桓

I. 模型描述

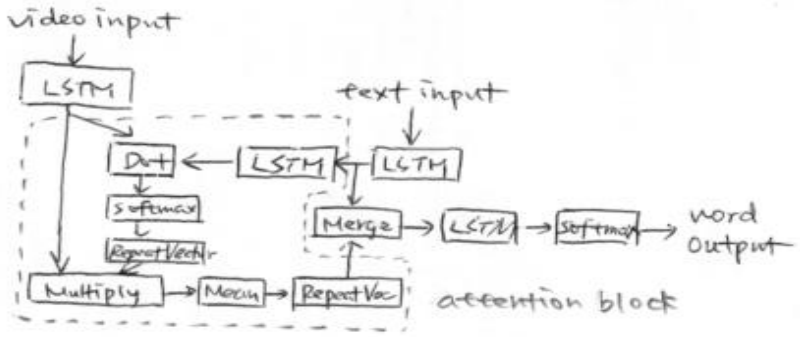
1. Seq2seq 模型

本次作業使用 Keras 2.0 實作類似於 encoder-decoder 架構的 Seq2seq 模型，靈感來自於 caption_generator: An image captioning project [1]。模型主要分成影片特徵模型（encoder）和語言模型（decoder）兩個部分，前者以助教預先抽取的影片特徵按時間序作為輸入，後者以句首標籤〈bos〉為始，在第 t 個時間點以前（t-1）個詞彙作為輸入，輸出第 t 個時間點的詞彙，直到生成句尾標籤〈eos〉或者輸出序列長度到達最大長度。模型架構的示意圖如下。詳細的實驗設定和結果後述。



2. 注意力機制（Attention）

以上述模型架構為基礎，嘗試在 Seq2seq 模型加入注意力機制。本次作業中實作的注意力機制是參考老師投影片提到的方法，先透過比對 encoder 和 decoder 輸出得到各 encoder 輸出的權重 α，再取所有 encoder 輸出的加權總和 c 作為 decoder 的輸入；取權重的比對方法也當作整個模型參數的一部份一起訓練，模型的示意圖如下。與原始模型之比較後述。



$$\alpha = h^T W z$$

$$c^0 = \sum \hat{\alpha}_0^i h^i$$

II. 嘗試改善模型的方法

1. 改變句子挑選的原則

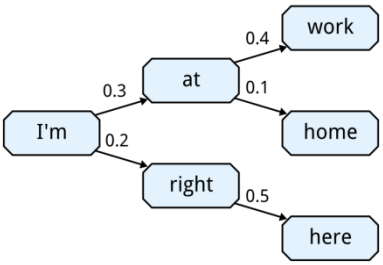
剛開始是在每次訓練時隨機挑選影片對應的句子作為該次的標籤，可能是因為同個影片對應到的句子變化性較大，感覺模型的訓練過程不太穩定。考量到模型的複雜度以及 BLEU 分數的算法，最後決定以每個影片對應的第四長的句子當作標籤下去訓練，雖然可以得到比較穩定的模型，但輸出的變化性就相對下降不少。

2. 對文本進行前處理

由於文本中的句子包含單複數、動詞型態等語法變化，導致詞幹相同的詞被視為不同的詞彙，因此也嘗試對文本進行詞幹提取(stemming) 後再訓練模型，但訓練出來的語言模型就失去文法概念，之後還要再修正輸出句的文法。考慮到文本中出現的句子時態大多相同，且作業的 BLEU 分數應該沒有考慮詞幹提取的部分，最後並沒有採用這個方法。其他關於標點符號的處理，除所有格或縮寫 (') 外全部刪除。

3. 集束搜索 (Beam search)

嘗試在上述之 Seq2seq 模型實作集束搜索，預測時每次都保留機率最高的前 k 名作為下次的輸入，最後取機率最高的句子作為輸出，目的是希望找出預測過程中潛在的可能更好的輸出，避免一步錯步步錯的情況。集束搜索之示意圖如右。



4. 調整模型的各參數設定，例如調整 LSTM 的維度等。

III. 實驗設定與結果比較

1. 環境設定

Python 版本：Python 3.5.3 / GPU：GTX 1080 Ti

引用的套件包：numpy (1.13.3), pandas (0.20.3), Keras (2.0.7), tensorflow (1.3)

2. 前處理

影片特徵的部分不做額外的處理，直接以助教預先抽取的特徵作為輸入。影片對應之標籤以 II. 1. 描述的方法處理，取對應的標籤中第四長的句子作為實際標籤，然後給其中出現的每一個詞對應的索引值，再根據其索引值做 one-hot 編碼。訓練時以影片特徵和前 (t-1) 個詞彙作為輸入，第 t 個時間點的詞彙作為對應之輸出。另外，由於這次作業提供的資料集資料數量較小，且有提供測試集之正確標籤，故沒有再另外切部分訓練集作為驗證集。

3. 實驗設定

經過前處理後文本中最長的句子長度為 m = 24，得到訓練資料共 1450m 筆，文本中的詞彙含特殊標籤<pad>, <bos>, <eos>, <unk>共 1984 個，以批大小 32 進行訓練 10 個迭代，訓練時間共 100 分鐘。模型中使用的 LSTM 維度皆為 256，以 RMSProp 進行模型參數更新 (學習率 = 0.001)。

4. 結果與比較

以上述之實驗設定最後得到的 BLEU 分數為 0.2641，僅略高於本次作業的底線。雖然嘗試了各種方法改善模型，但都沒有辦法提升 BLEU 分數，產生出來的結果也沒有顯著的改善。以下表列各種模型最後得到的分數，其他模型皆使用相同的實驗設定。

模型	BLEU 分數
(1) Seq2seq	0.2641
(2) Seq2seq + Attention	0.2306
(3) Seq2seq + Beam search	0.2369
(4) Seq2seq + Attention + Beam search	0.2268

考慮到 BLEU 分數的計算方式與限制，分數高的輸出不盡然是好的輸出，底下也列出兩組不同模型產生出的句子互相比較，並觀察和正確標籤之間的差異。

例 1. "WTf5EgVY5uU_124_128.avi"

模型	輸出
(1)	A woman is pouring some chicken into a glass bowl
(2)	A person is slicing some onion
(3)	A woman is pouring eggs into a bowl of water
(4)	A person is slicing some onion

在以上的例子中正確的標籤包含"A woman is adding sliced onion to pan.", "A person puts garlic in a pan.", "A person pours ingredients into a pan."等句子，幾種不同模型的輸出都與正確標籤有相似的部分。

例 2. "ScdUht-pM6s_53_63.avi"

模型	輸出
(1)	A man is putting a piece of paper into a bowl
(2)	A man is playing an guitar
(3)	A man is cutting a piece of paper
(4)	The man is playing the guitar

在以上的例子中正確的標籤包含"A man is putting salt on a chicken.", "The man is putting flour on the chicken.", "A man making a chicken"等句子，雖然模型(1) 的輸出看起來和正確的句子有相似之處，但是四個輸出都不符合該情境之描述。

參考資料

[1] https://github.com/anuragmishracse/caption_generator
[2] https://github.com/chenxinpeng/S2VT/blob/master/model_RGB.py
[3] https://research.googleblog.com/2016/05/chat-smarter-with-allo.html