壹、題目

Listen & Translate

貳、Team Name

隊伍名稱：

★☆無限期支持宿舍炊膳合法化☆★

Kaggle Name：

NTU_b03502125_★☆無限期支持宿舍炊膳合法化☆★

參、Members

組長：

倪嘉宏 b03502125

組員：

黃文鴻 b03701118

組員：

徐一真 b03901063

組員：

林威利 b03902047

肆、Work Divisions

倪嘉宏：行政、報告撰寫、第一種模型(初期)、實驗

黃文鴻：工作站維護、報告撰寫、開發環境維護、第一種模型(初期)

徐一真：第一種模型、報告撰寫

林威利：第一種模型(初期)、第二種模型、報告撰寫、實驗

伍、Preprocessing & Featuring Engineering

一、第一種 model:

1.Training Data

送入的資料包含已經處理好的音訊MFCC，以及對應的中文翻譯。這些Training Data讀入後，我們會先將對應的中文翻譯的部份每句前都加上'\t'、而每句後面都加上 '\n' 之後將整個句子逐字編碼，編碼採用one-hot encoding。

2.Testing Data

Testing Data 的部分，preprocessing的過程與 training data一樣。所有的答案(中文句子)都會通過與Training Data 相同的編碼。

二、第二種 model:

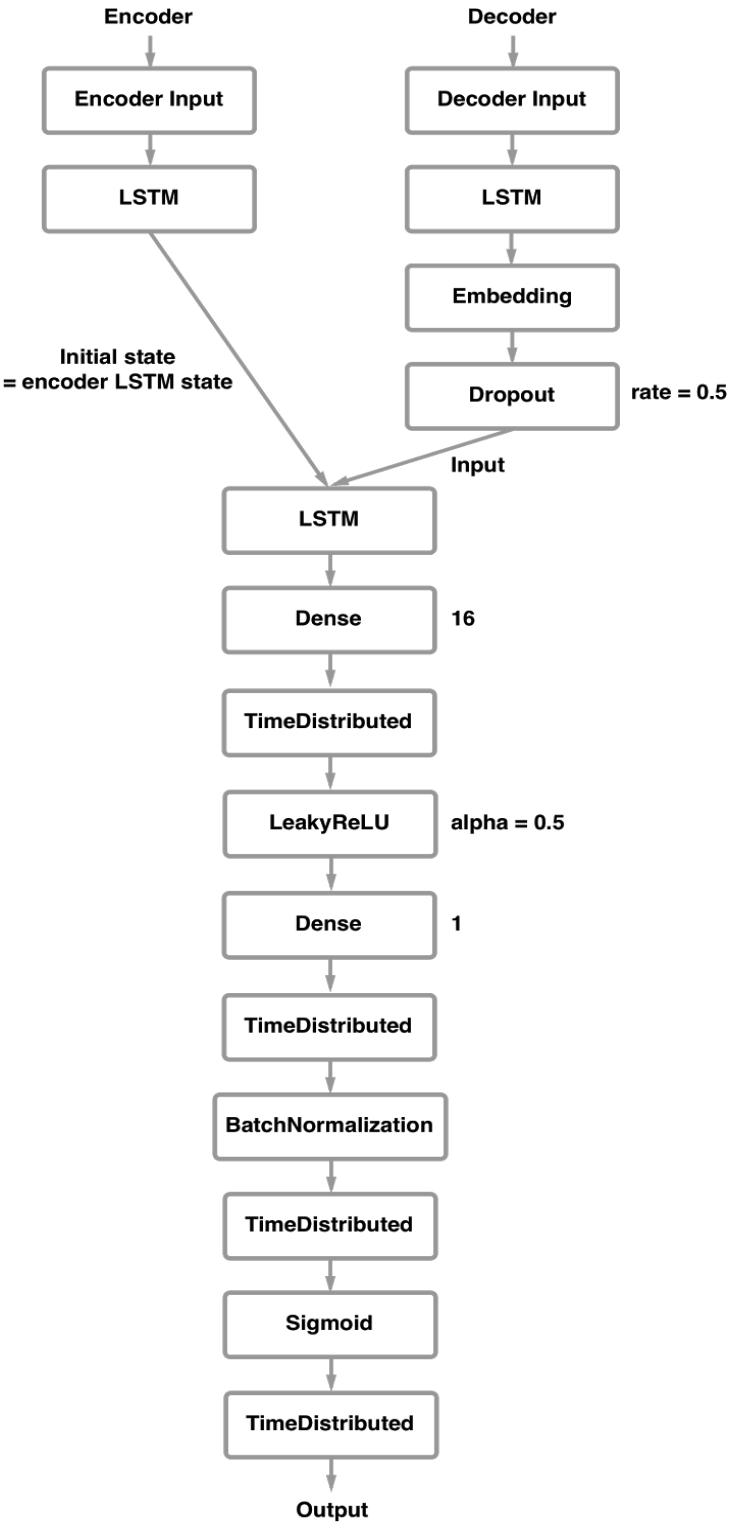
基本上與第一種 model 相同，但 caption 部分的編碼採用單純 zero padding，沒有額外加上 '\t' 與 '\n'。

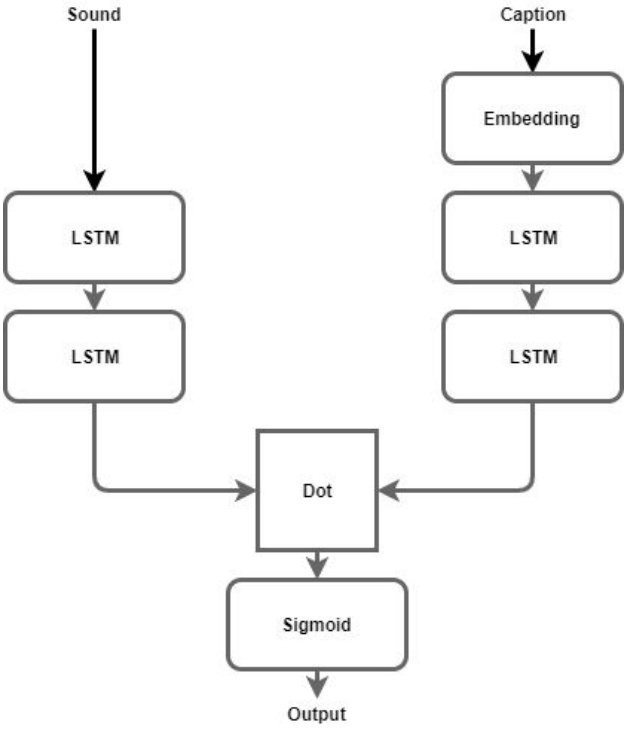
陸、Model Discription

一、第一種模型

如右圖所示：

Encoder input 是 MFCC，Decoder input 則是編碼後的中文翻譯。除了原本的 MFCC 與正確的翻譯以外，我們也透過將 MFCC 與錯誤翻譯配對產生錯誤的範例，並將錯誤範例的 label 設定為 0 (正確範例的 label 則是 1)，讓模型學會判斷正確/錯誤的翻譯。





二、第二種模型

如左圖所示：

第二種模型一樣會 sample 出正確與錯誤的 聲音-翻譯配對，透過兩層 LSTM layer 分別 encode 聲音與翻譯的 embedding 之後做內積，讓正確的部分盡量接近 1，而錯誤的部分則是盡量接近 0。

柒、Experiments & Discussions

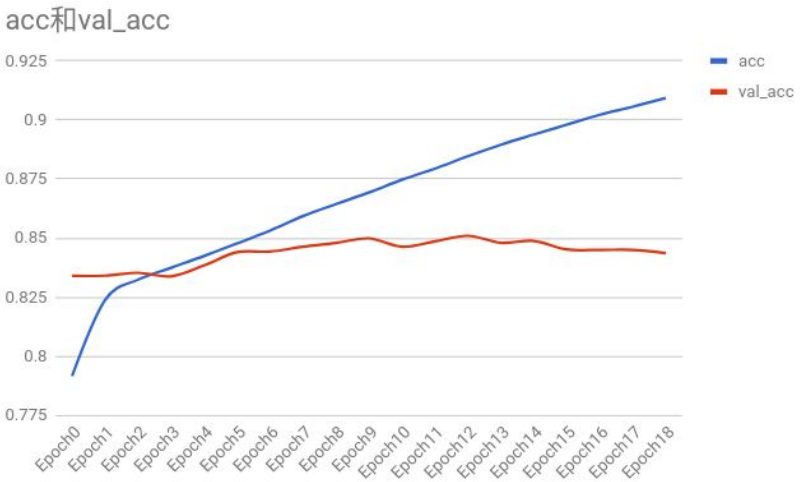
我們對兩種模型分別做了不同的實驗與討論，因此以下將各自描述實驗內容與結果。

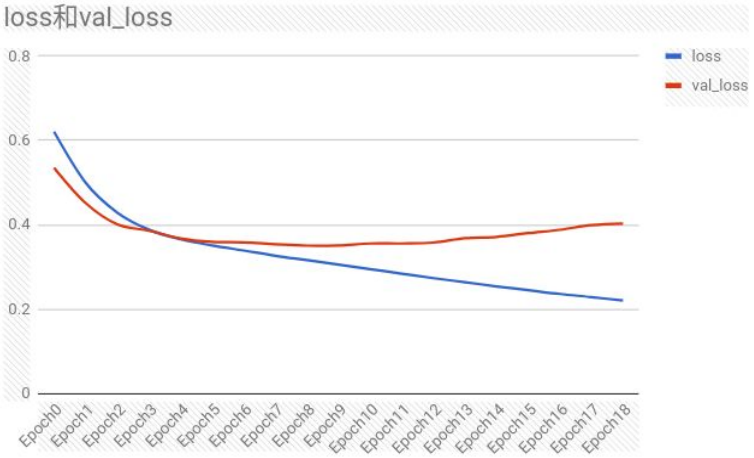
一、第一種模型

1. 模型訓練過程探討

右邊兩張圖分別為訓練的epoch數量對 loss 與對 accuracy 的關係圖。Accuracy 的算法是根據 Model Discription 一節所描述，將正確的 MFCC 與翻譯標上1，而另外將這些 MFCC 搭配不正確的翻譯並標上 0，其中正確的答案與錯誤的答案分別佔一半，使得隨機猜答案時 accuracy 大約為 0.5。

如圖所示，大約在第五個 epoch 時 loss 與 accuracy 就已經達到最佳狀態，因此訓練更久也只會使得結果 overfit。





2.模型參數探討

Layer (type)	Output Shape	Param #	Connected to
input_2 (InputLayer)	(None, None)	0	
embedding_1 (Embedding)	(None, None, 256)	256	input_2[0][0]
input_1 (InputLayer)	(None, None, 39)	0	
dropout_1 (Dropout)	(None, None, 256)	0	embedding_1[0][0]
lstm_1 (LSTM)	[(None, 256), (None,	303104	input_1[0][0]
lstm_2 (LSTM)	[(None, None, 256),	525312	dropout_1[0][0] lstm_1[0][1] lstm_1[0][2]
time_distributed_1 (TimeDistrib	(None, None, 16)	4112	lstm_2[0][0]
time_distributed_2 (TimeDistrib	(None, None, 16)	0	time_distributed_1[0][0]
time_distributed_3 (TimeDistrib	(None, None, 1)	17	time_distributed_2[0][0]
time_distributed_4 (TimeDistrib	(None, None, 1)	4	time_distributed_3[0][0]
time_distributed_5 (TimeDistrib	(None, None, 1)	0	time_distributed_4[0][0]

Total params: 832,805
Trainable params: 832,803
Non-trainable params: 2

除了自變數以外的變數都為控制變數，下面探討的變數當中各項的預設參數分別為 data_size = 5，leaky_alpha = 0.5，dense_width = 64

(1) time_distributed_3 當中的 LeakyReLU

time_distributed_3為一 Leaky Rectified Linear 層。調整其 leak 的斜率 (alpha) 對於最後的正確率有顯著的影響。簡單列表如下：

Leaky alpha	Accuracy	Loss
0.3	0.8464	0.3512
0.5	0.844	0.3617
1.0	0.8458	0.3582

(2) time_distributed_1當中的 dense_width

time_distributed_1為一dense層。此參數為dense的node數量

dense_width	Accuracy	Loss
16	0.844	0.3617
32	0.8484	0.346
64	0.851	0.3408

(3) data_size

我們的訓練資料當中包含許多組 MFCC與翻譯的配對，其中有些配對是正確有些配對是錯誤的。產生錯誤配對的方法是由給定的 MFCC 配上其原本對應中文翻譯底下 n 筆翻譯，也就是我們說的 data_size。舉例來說，data_size 是 5 的情況下，產生的總資料數量會是題目給定的資料數量的 6 倍，其中第 1 份資料答案被標注為 1 ，第 2 至 6 份資料答案皆被標注為 0，而內容則分別是NFCC資料對應其原本中文翻譯的後 1 至 5 筆翻譯。

data_size	Accuracy	Loss
1	0.6646	0.6012
2	0.7088	0.5386
3	0.7872	0.4416
4	0.8252	0.3857
5	0.844	0.3617

二、第二種模型

在沒有另外說明的情況下，除了各項所變動的參數以外，各項參數的 default value 分別為

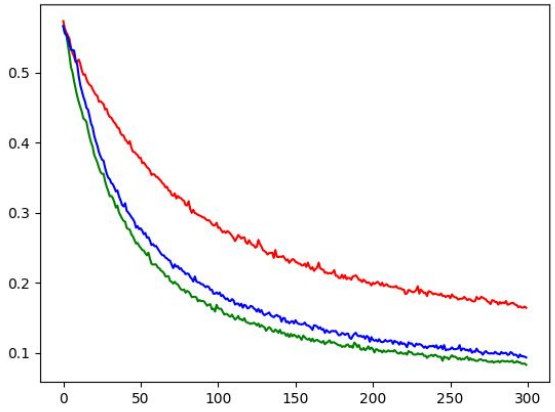
LSTM 層數 : 2

LSTM hidden dimension : 100

Bidirectional LSTM : 無

Negative Sampling : 4

1. LSTM層數



X軸: epoch

Y軸: training loss

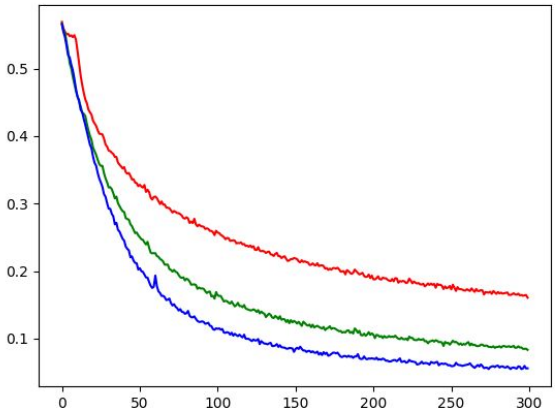
紅色: 1 層 LSTM

綠色: 2 層 LSTM

藍色: 3 層 LSTM

在這個題目裡，2 層 LSTM 似乎就足以學習 MFCC 與 翻譯的配對，加到 3 層反而對 training 過程沒有甚麼幫助。

2. LSTM hidden dimension



X軸: epoch

Y軸: training loss

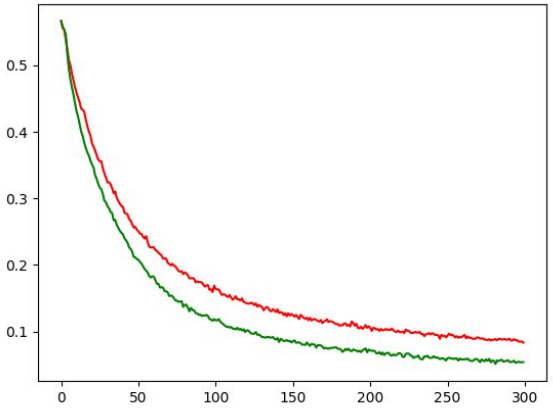
紅色: 50

綠色: 100

藍色: 200

LSTM hidden dimension 在 [50, 100, 200] 的範圍裡似乎是越多越好，但 hidden dimension 越大 training 和 testing 也會越慢，因此我們在其他實驗裡以 100 為 dufault value。

3. Biderectional LSTM



X軸: epoch

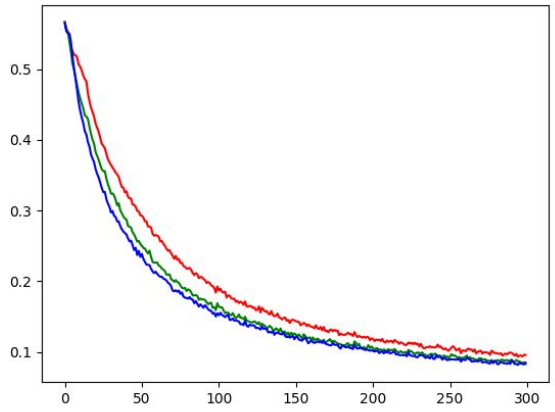
Y軸: training loss

紅色: LSTM

綠色: Bidirectional LSTM

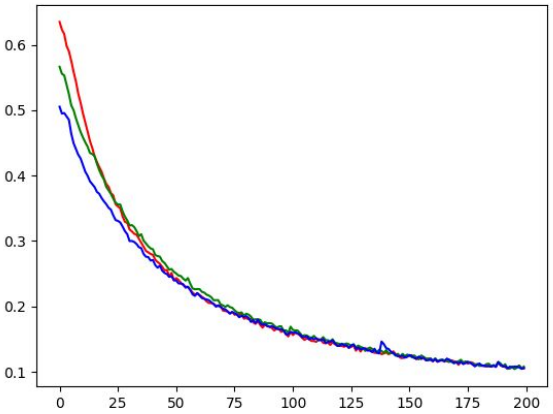
加上 Bidirectional 的 LSTM 的確表現比較好，但進步幅度不大。與 hidden dimension 相同，他的缺點是 model 的大小，training/testing 要花的時間等等也會跟著 double，是準確率與效率之間的取捨。

4. Word embedding dimension



X軸: epoch
Y軸: training loss
紅色: 50
綠色: 100
藍色: 200
Word embedding 的 dimension 在這次的題目裡影響並不明顯。如圖所示，除了收斂稍微快一點以外，word embedding dimension 從 50 增加到 200 對準確率並沒有甚麼差異。

5. Negativa sampling



X軸: epoch
Y軸: training loss
紅色: 3
綠色: 4
藍色: 5
與第一種模型的 data_size 相同，我們在第二種模型也有做類似的實驗。雖然初始的 loss 因為 sample 比率不同而有落差，但很快他們就幾乎重合在一起，說明 negative sampling 的比率對訓練不是特別重要。

捌、Reference

[1]Krupakar, H., Rajvel, K., Bharathi, B., Deborah, S. A., & Krishnamurthy, V. (2016, February). A survey of voice translation methodologies—Acoustic dialect decoder. In *Information Communication and Embedded Systems (ICICES), 2016 International Conference on* (pp. 1-9). IEEE.

[2]Wöllmer, M., Zhang, Z., Weninger, F., Schuller, B., & Rigoll, G. (2013, May). Feature enhancement by bidirectional LSTM networks for conversational speech recognition in highly non-stationary noise. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on* (pp. 6822-6826). IEEE.

[3]Bérard, A., Pietquin, O., Servan, C., & Besacier, L. (2016). Listen and translate: A proof of concept for end-to-end speech-to-text translation. *arXiv preprint arXiv:1612.01744*.