

隊長：R06922002 姚嘉昇

組員：R06942054 潘仁傑、R06522620 王仁蔚

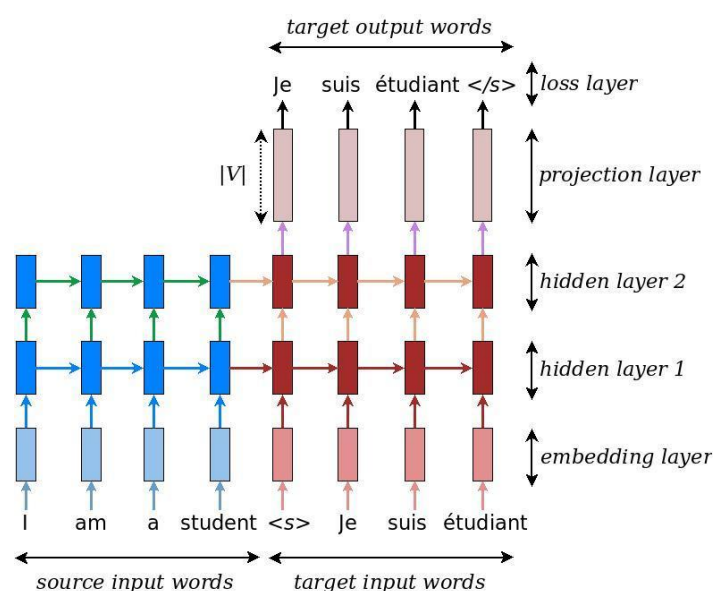
● 分工表

	2-1	2-2	Report 2-1	Report 2-2
姚嘉昇 R06922002		嘗試 pre-train w2v + beam search		
王仁蔚 R06522620		嘗試 attention		
潘仁傑 R06942054		嘗試 scheduling		

● Model description (3%)

參考 Tensorflow 之 Neural Machine Translation (seq2seq) Tutorial 架構圖 (如下圖)，建構這次作業所使用的基礎模型。

- ✓ 一層 Embedding layer (size = 1024)
- ✓ 兩層 Dynamic RNN (size = 1024)
- ✓ 最後一層 Projection layer



● How to improve your performance (3%)

1. Write down the method that makes you outstanding (1%)

- ✓ Pre-train word2vec
- ✓ Beam Search

2. Why do you use it (1%)

- ✓ Pre-train word2vec

因助教提供的字典為以字為單位，但中文大多都是以詞為單位表達意思，因此我們想自己建立一個辭典，並將其轉為詞向量。

✓ Beam Search

因 2-1 已經使用 attention 機制，2-2 則使用 Beam Search，因為傳統的做法是使用 greedy，而 Beam Search 可以讓 model 看過更多輸出，進而找到更好的輸出。

3. Analysis and compare your model without the method. (1%)

在 data preprocessing 的部份，先建立自己的辭典，助教只給字典，因此我們先從 training data(cnr_conversation.txt)中取出長度大於 2 的詞彙，並且將助教給的字典併到裡面成為我們最終的辭典，這樣就能以 word-based 的方式來做 chat-bot，而直接用助教的字典且用 char-based 的方式來做 chat-bot，perplexity 大概是 11 左右，correlation score 大概是 0.45 左右，比較不穩定，而使用 word-based 後，perplexity 雖然差不多，但是 correlation score 大概上升到 0.5 左右，較為穩定

在 data preprocessing 中，先將 training data 濾過一遍 (filter.py)，把沒在辭典裡的 word 變成 unknown token，並在每一句話前面加 begin token 後面加 end token，之後在 data_preprocessing.py 中加 padding 到最大的 time step，將處理後的 data 先丟到 gensim train 出 word2vector model，做為 train seq2seq 的 pre-trained model，這樣的作法在 correlation score 上有明顯的提升(見表格)

Pre-train W2V	Beam Search (size)	Perplexity	Correlation Score
No	No	6.96	0.38256
No	7	11.83	0.49207
Yes	No	9.26	0.45864
Yes	7	11.83	0.53626

W 在 inference 中，使用 beam search 雖然會讓 perplexity 提升(loss 變大)，但是 correlation score 有變好，不過 beam search size 大到一定程度後，correlation score 即不再提升(見表格)

- Experimental results and settings (1%)

Experimental results

Pre-train W2V	Beam Search (size)	Perplexity	Correlation Score
Yes	No	9.26	0.45864
Yes	1	9.26	0.45864
Yes	2	8.05	0.47926
Yes	3	9.34	0.51268
Yes	4	10.13	0.52513
Yes	5	10.87	0.52878
Yes	6	11.6	0.53464
Yes	7	11.83	0.53626
Yes	8	12.01	0.53497
Yes	9	12.25	0.53313
Yes	10	12.44	0.53358
Yes	11	12.55	0.53084
Yes	12	12.59	0.52807
Yes	13	12.70	0.52643
Yes	14	12.80	0.52537
Yes	15	12.81	0.52101
Yes	16	12.82	0.51790
Yes	17	12.87	0.51361
Yes	18	12.89	0.51016
Yes	19	12.90	0.50686
Yes	20	12.91	0.50390
Yes	30	12.99	0.48391
Yes	40	13.16	0.47974
Yes	50	13.40	0.47635

settings

Optimizer	sgd
Learning rate	0.5 (隨著時間下降)
每個 Epoch sample 的資料量	50 萬筆
Batch size	100
Epoch	150 (與助教 750000 iterations 一致)