Final Project

題目: Conversations in TV shows

隊伍名稱: NTU_r06944031_weiting 戰隊

成員:

學號	姓名
r06944031	林蔚廷
r06922117	李岳庭
r06922154	黃俊錚
r06922135	魏禎

分工:

Concatenate data: 黃俊錚、李岳庭

Jieba 斷詞: 林蔚廷、魏禎

Gensim Word to vector: 林蔚廷、黃俊錚

Testing data 預測: 魏禎、李岳庭

Seq to seq: 李岳庭、林蔚廷

參數調整: 林蔚廷

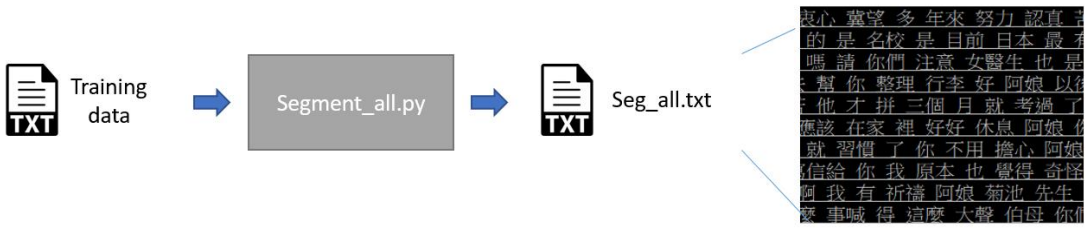
Report: 黃俊錚、李岳庭、林蔚廷

Preprocessing/Feature Engineering:

1. 串接所有的 training data 一起做斷詞
2. 使用 Jieba 斷詞

   dictionary: dict.txt.big
3. 使用停用詞

   stopwords: https://github.com/zake7749/word2vec-tutorial/blob/master/jieba_dict/stopwords.txt
4. 最後所有資料斷詞的結果存在 Seg_all.txt



Model Description:

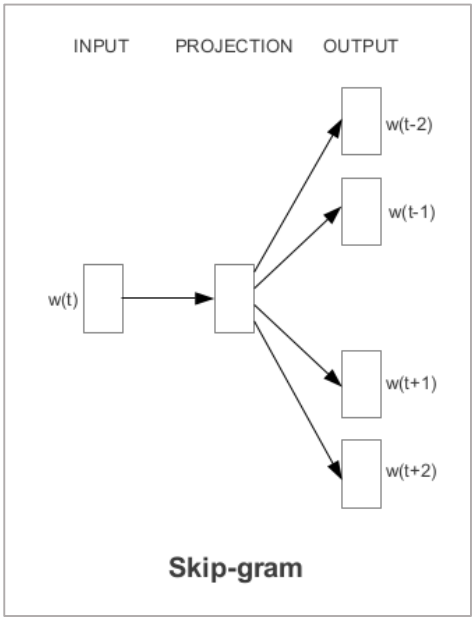
我們使用了 gensim 的 word2vec 套件，train 出幾個 word2vec model，將劇本所有句子先用 jieba 斷詞之後，每個詞用空白分開，輸出成一個很大的 txt 檔，再丟入 word2vec 下去 train，試了好幾組參數得到幾個 model。

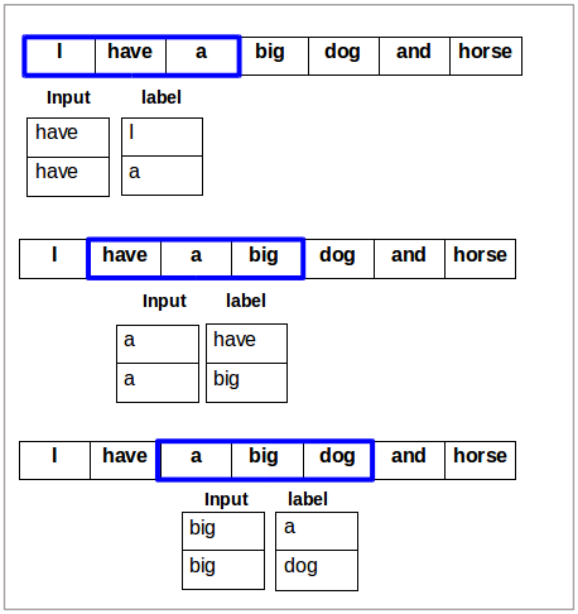
預測 test data 時是將每個問題作斷詞，將每個詞以 model 轉成 vector 後相加，每個選項也是斷詞後將 vector 相加，然後比較問題與每個選項 cosine 值，最大者為最佳解。

在 model 的參數中，我們主要調整 size 與 window 兩個參數，size 就是將每個詞轉成幾維的 vector，window 是句子中前後看幾個詞，而試過很多組參數都無法突破 strong baseline，後來發現主要差別是 sg 這個參數，原本預設值是 sg=0，將 sg=1 後準確率提升很多。

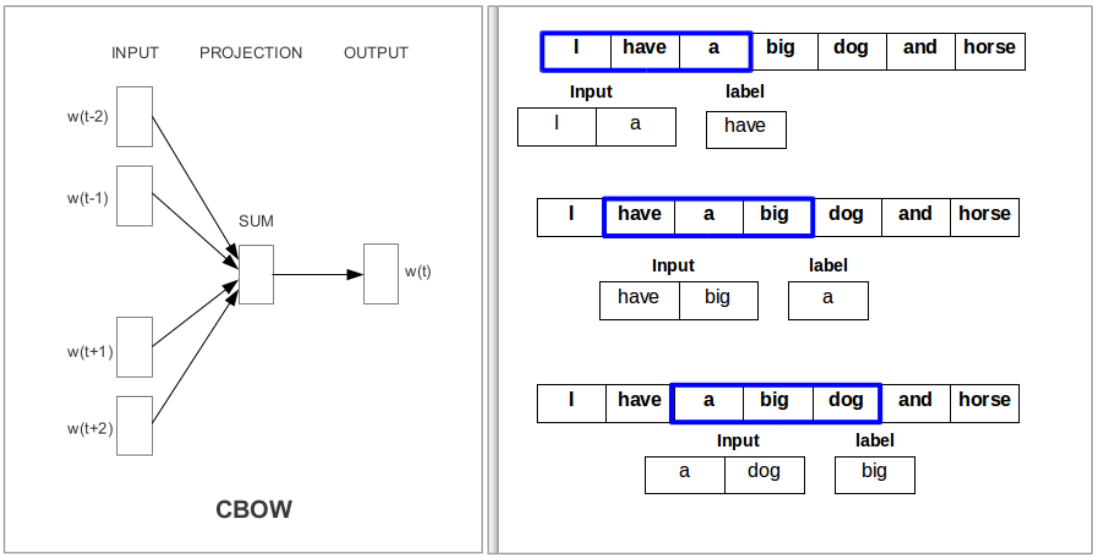
由 gensim 官網上的文件查到，sg 是 skip-gram 的意思，sg=0 是 CBOW 模型，sg=1 就是使用 skip-gram 模型，兩者差別如下：

Skip-gram 的邏輯是，一次只輸入一個字，輸出的 label 為其前後一定距離內的文字，所以同一個 word 會有多個 label 假設一段句子"I have a big dog and horse". 前後距離設定為一個字，我們取"dog"為 input word，label 就是"and" 跟"big"。如果距離是兩字寬，label 就是"and"、"big"、"a"、"horse"。我們以"dog"的前後幾個 label 為依據，給該 word 一個向量值。



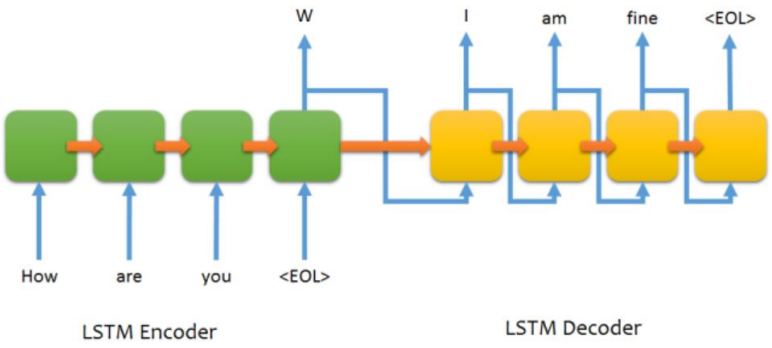


CBOW(Continuous Bag-of Words)是將一段句子的中間字當作 label，其左右文字為 input words，所以是多個字 input 一個輸出 label. 句子的長度可調。



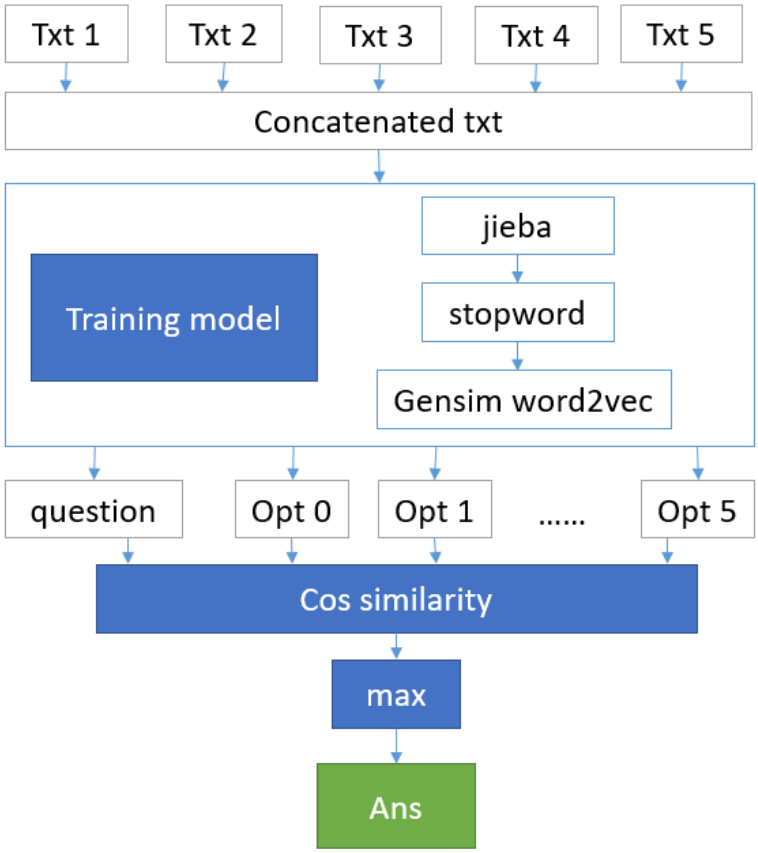
我們就以 window 值來決定要前後看幾個詞，使用 skip-gram 的方式來 train 出每個詞的 vector，建立 word2vec model，至於產生 vector 的原理，跟機率以及 language model 有關，在此不多詳述。

另外我們也有嘗試 sequence to sequence，將 training data 六句話為一組，使用前五句來預測第六句，使用 LSTM 來訓練，但是最終結果並沒有進步。



Experiments and Discussion:

整個 Project 的流程圖:



針對不同的 size, window，在 kaggle 上 public scrore:

[1]一開始使用 CBOW，調整參數仍舊無法過 strong baseline

Size	window	Sg	score
250	10	0	0.41343
400	50	0	0.44071
500	50	0	0.43201
400	60	0	0.43359
400	70	0	0.44150
400	75	0	0.43280
400	80	0	0.43517

在使用 CBOW 下，size=400, window=70 的表現最佳

[2]發現 skip-gram 表現比較好，改用後成績大幅提升

Size	window	sg	score
10	7	1	0.42569
75	7	1	0.48181
100	7	1	0.48458
150	7	1	0.48379
100	10	1	0.49486
100	15	1	0.51225
100	20	1	0.51660
100	25	1	0.51778
100	30	1	0.51343
100	27	1	0.51660
100	26	1	0.51897

最後表現選用的參數:

Size = 100, Window = 26, Sg = 1,

Public score = 0.51897

參考資料:

[1] 如何使用 JIEBA 結巴中文分詞程式

[2] keras/lstm_seq2seq.py

[3] models.word2vec – Deep learning with word2vec

[4] Word2vec Tutorial

[5] pig_latin

[6] Word2Vec model Introduction (skip-gram & CBOW)

[7] negative sampling

[8] gensim - word2vec