ML Final Report - TV

停院深深深度學習

B03505052 李吉昌
B03902130 楊書文
B03505040 林後維
B03303032 劉祐瑄

I. Introduction & Motivation

此次題目為中文電視劇的對話系統，透過深度學習訓練，對話系統接收到兩到四句對話後，將從 6 個選項找出一個合理最為接下來對話的答案。

訓練過程中，將訓練集做分詞處理後以不同架構，包括詞袋模型(Bag-of-words)、遞迴神經網絡(RNN)、長短期記憶(LSTM)將文字嵌入向量空間，訓練模型對於前後句的學習。

測試答案部分，由於此次題目非 sequence to sequence 輸出回句，因此不需要使用生成式模型，而是使用檢索式模型，在選擇下一句的選項時鄉所有選項與前句作相似度比較，判斷正確答案。

II. Data Preprocessing / Feature Engineering

Data preprocess 可以分成三個部分來討論，分別是中文分詞、停用詞以及 training data 的生成(generator)。

1.中文分詞：

中文與其他語言有一個很大的差別，就是中文並不會將每個詞用空白分開，然而每個詞卻可能享有相同的字(如：「聲音」和「聲望」兩者為意義完全不同的詞，卻都擁有「聲」這個字，而大部分的狀況下，單個字也可以作為一個詞，像是「書」)，因為這個特性，當我們在對中文做語言分析時，通常會對其做分詞，分詞也成為了中文語音處理的重要功能。

這次實作中，在經過網路上的搜尋後，我們實際測試了三個分詞工具，分別為 Jieba(結巴分詞系統)、NAER(國教院分詞系統)、Stanford word segmenter(Chinese)。下表為我們用不同分詞工具分詞的結果：

Jieba	Stanford Word Segmenter	NAER
		

由於我們使用 RNN 訓練，希望詞可以切越細越好；在比較三個不同分詞工具的分詞結果後，可以看到 stanford word segmenter 訓練的結果較差，像是 9937 行的「你們」會被切成「你」和「們」兩個詞，更嚴重的是「們」會和後面的字合併成一個詞，這讓結果中會產生大量罕見詞(像是「們家拉電」、「們那」、「們抓進牢裡」)，而 Jieba 和 NAER 的分詞滿相近且較合乎語法。

將兩者的分詞結果都訓練一次後，Jieba 的準確率只有 0.491，而使用相同的模型訓練 NAER 分詞出來的結果的話，準確度可以達到 0.512，因此最後我們使用 NAER 的分詞結果來做訓練。

2.停用詞：

停用詞是指文章中最常出現的一些字，而這些字通常為語助詞、對判斷該段文字語意的幫助並不大，諸如：「上下」、「噓」、「嗯」，所以在判斷文字語意時會把這些字拿掉，然而我們這次的作業中，我們有嘗試但最後沒有將停用詞拿掉，因為 testing 時用的語句太短，停用詞去掉後只剩 2、3 個詞，甚至 0 個詞，所以最後決定保留停用詞。

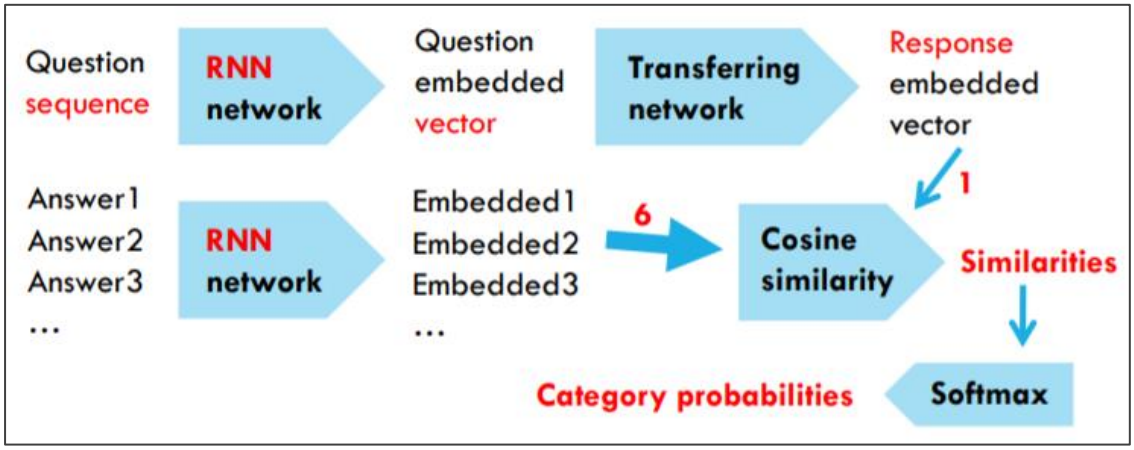
3.生成 training data：

生成 training data 是這次實作的重點，主要原因為 training data 和 testing data 是完全不同的形式，training data 為連續的語句、並且明確的一句一句切開，然而 testing data 是只有兩三句的對話、而且沒有分句混雜在一起的句子，所以我們必須將 training data 盡量貼近 testing data 的格式。

我們的作法是將原始拿到的 data，每兩句併成一句問句(dialogue)，並以下一句作為正確答句(options 的正確答案)，然後從 raw data 中任意取五句出來做為錯誤答句(options 中的錯誤答案)，如此一來我們就可以得到格式類似於 testing data 的 training data，然後以同樣的方法再做一次，只是改成每三句合併成一句問句，如此一來我們就可以產生 raw data 兩倍的量的 training data，而其格式語 testing data 一樣，是每兩三句被壓縮成的問句，以及六句的選項。

III. Model Description (At least two different models)

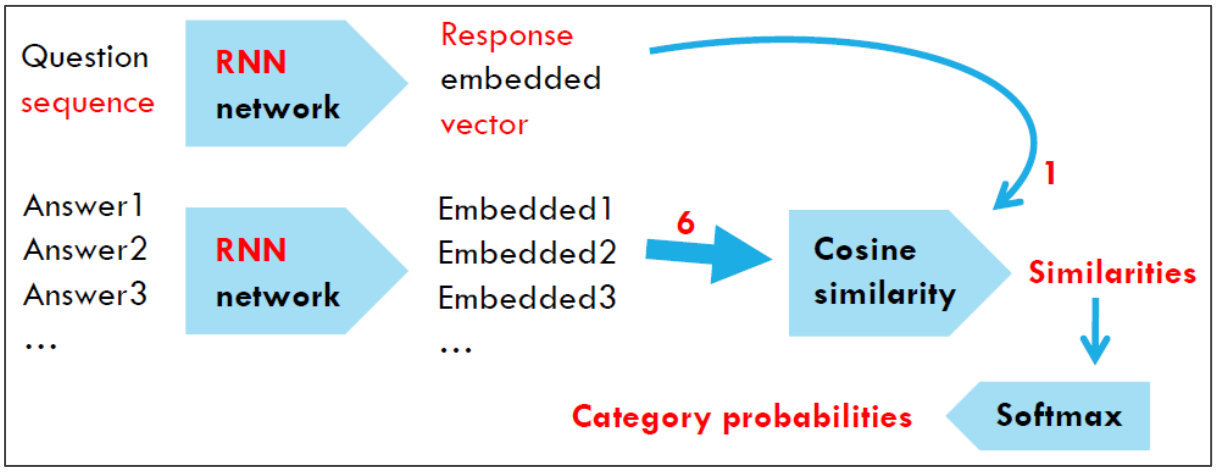
我們主要使用的 model 如下圖：



會先準備 1Q6A，其中 question 為 training data 中連續的 2 或 3 句話 concatenate，6A 則是從 training data 中隨機選出 5 句話(不連續)作為錯誤答案，並將緊接在 question 後的下一句話當作正確答案。每一筆製作出來的 training data 都是 1Q6A 的形式，training label 則是 0~5，代表著六個 category，我們要從 6 類中選出正確的下一句話。如此，此模型將問題歸類為 category classification。

model 的架構首先會將 question(一句較長的話)通過一個 question RNN 得到 question semantic vector。並將 6 個 answer 都通過同一個 answer RNN 得到六個 answer semantic vectors。接著 model 會再將 question semantic vector 通過一個 DNN 得到 response semantic vector。最後將 6 個 answer semantic vector 分別都和 response semantic vector 做 cosine similarity，得到 6 個 similarities，代表了六個選項各自作為 question 下一句話的合適度。最後將六個 similarities concatenate 起來成為一個一維 vector 並經過 softmax，得到六個選項作為下一句話的機率。最後這步非常重要，傳統上直接算出每個選項的 similarity (其他兩組的作法) 再人為比較，會出現多個選項的 similarity 不相上下的情形，造成 model 因為細微差距而選錯答案。透過 softmax 將六個選項放在一起比較，最好的選項拉高機率，最差的壓低機率，能讓 model 學到選項間的「相對好壞」，而不僅是 similarity 要衝到 1 還是 0 的絕對分數。

第二種 model 架構則是將上圖簡化後的版本，我們將上圖的 Transferring network 拿掉，試圖讓 question RNN 直接將 question embed 成 response semantic vector，並藉此簡化 network 架構。

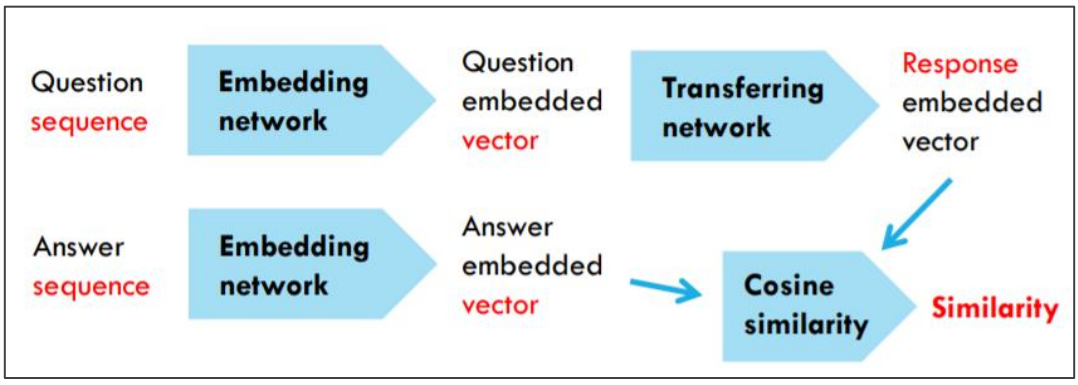


由於拿掉 Transferring network 並不影響 model 的能力，且能提升訓練速度，後來主要以第二種模型訓練。

IV. Experiment and Discussion

Experiment 將分為 Word Embedding 方式、Training Data Generator、Rare Words Matching 三部分討論。

1. Word Embedding



在將 question 和 answer 做 embedding 的時候，我們嘗試使用兩種不同的 embedding 方式，分別為 Bag of Words 和 RNN/LSTM。

- Bag of Words (BOW):

當我們使 BOW 做 embedding 的時候，我們只考慮該句裡面是否出現某個詞，而不考慮先後順序，且當問句較長的時候，我們是使用"OR"把多具串起來。

因為沒有考慮字詞的前後順序，對於語意的判定容易因為每個詞的位置未知而產生誤判，因此使用 BOW 的 model 正確率只有 0.36-0.38。

- RNN:

有鑑於字詞出現的前後順序對於語意分析的影響，為了讓 model 更完整的學習到 training data 內的 question 和 answer 所代表的意思，以及他們出現的 pattern，我們改採用 RNN 的方式，並使用 LSTM 和 GRU 兩種不同的 cell 做實驗。同時因為 RNN 可以學習語句中字詞的先後順序，因此當問句較長的時候改以將多句話"Concatenate"起來當作問句。

使用 RNN 做 embedding 的 model 正確率可達 0.42-0.45，明顯較上面 BOW 的方式好。

2. Training Data Generator

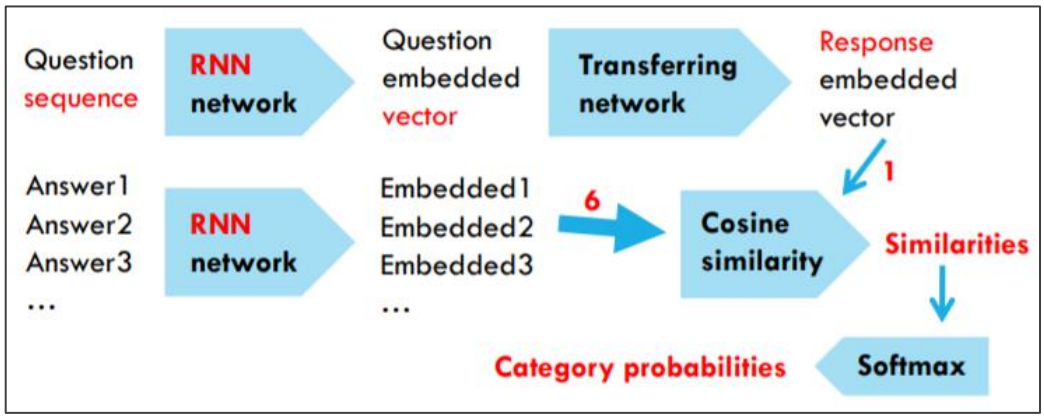
由於這次的 training data 並非與 testing data 一樣是每題有問句有答案選項的資料，而是長篇連續的劇本，因此在產生 training data 的時候我們嘗試了兩種不同的方式。

- 1 Question 1 Answer:

1Q 對 1A 的方式是很直覺依據題目要判斷下一句話是什麼，因此直接將 training data 製造部分的下句是正確的(label 為 true)，部分的下句是錯誤的(label 為 false)作為訓練集。

這個方式主要會遇到兩個問題，第一，訓練過程中的 loss (similarity)並不能反映 testing 選擇題方式的 score；第二，若為單純的 binary classification 的方式去訓練的話，模型學習後只學習到判斷 true 或 false，但 testing 的選擇題需要的是判斷哪個是最有可能的答案，並非單純對錯。由此可知這樣的方式，對於訓練模型並不有效，使得正確率無法超過 0.5，因此我們改嘗試使用 1Q 對 6A 的方式。

-1 Question 6 Answers:



有鑑於測試資料為六的選項的選擇題，因此我們選擇 1Q 對 6A 的方式(如上圖)。這個模型在 answer 產生的部分改以每次產生 6 個答案，其中一為正確的下句，由於不再只是判斷對錯而是選擇哪個是正確的答案，訓練的問題從 binary classification 改為 categorical classification，與 testing 所要求的相同。

實作方式從上圖可看到，將六個答案 embedding 後 cosine similarity，與之前不同的是 similarity 會經過 softmax，找到最有可能的答案。經過 generator 的調整後，模型的正確率可達到 0.50-0.51。

3. Rare Words Matching

依照詞頻的概念愈少出現的字詞愈具特殊性，若某個罕見詞出現在答案中，我們認為有比較大的可能就是正確答案，因次我們在判斷答案的時候設定了不同出現頻率的 threshold，1000 是我們窮舉後最好的量，設定太小可被判斷的字詞數量過小，設定太大罕見詞的意義就會消失。

MODEL	PUBLIC	PRIVATE
w/o rare words matching	0.56047	0.56086
Threshold 500	0.56086	0.56205
Threshold 1000	0.56324	0.56166

V. Conclusion

在處理這次中文電視劇對話系統的題目時，有三個重點，分別為資料前處理、訓練資料生成以及模型架構。經過實驗後我們發現：

1. 分詞方式對於結果有很大的影響。在嘗試不同的分詞系統後，同個模型架構準確率可從 0.49 提升到 0.51，因此最後採用準確率最高的 NAER。

2. 訓練資料生成與測試資料愈相近，訓練結果愈好。當我們從原本生成 1Q1A 的 generator 改為 1Q6A 後，不但訓練時的 loss 可以更準確反映 testing 的結果，同時將問題回歸到 categorical classification，判斷六個選項作為下句的機率，提高模型訓練的成果。

3. 將 Question 與 Answer 的 RNN network 獨立，不僅提升模型準確度，同時能夠在不影響模型 representation 能力的情況下，刪去 transfer network，提升訓練速度。

嘗試不同資料前處理、架構等變化後，最終 private 最佳成績為 0.562，public 為 0.563。

VI. Reference

1. 國教院中文分詞系統：https://github.com/naernlp/Segmentor
2. 結巴分詞系統：https://github.com/fxsjy/jieba
3. Stanford NLP Chinese Word Segmentor：https://nlp.stanford.edu/software/segmenter.html