MLDS HW4 Report

Environment & data sets (1%)

- OS: Linux, MAC OS
- CPU: Intel® Core™ i7-6700 CPU @ 3.40GHz
- GPU: K80
- Libraries:
  - `nltk`（corpus處理)
  - `python v3.5.2`
  - `tensorflow v1.0.1`
  - `numpy v1.12.1`
- Dataset:
  - Marsan-Ma chat corpus (https://github.com/Marsan-Ma/chat_corpus)
  - `open_subtitles.txt.gz`
  - `movie_subtitles_en.txt.gz`

Model description & reward functions (2%)

- data 前處理
  - 去掉標點符號，作tokenize
  - 加上結尾字符，以最大長度20字作padding
  - 濾掉回答過短、unkown字過多的句子
  - 切validation後剩下 910,357筆pair作為training
- 採用Encoder-Decoder+Attention的架構，中間使用3~4層的LSTM作為Hidden Cell
  - Hidden layer size 256
  - Embedding size 130
  - Vocabulary size 10000
  - 字串最大長度設為20
- 將最後一層的output project成10000維的vector作為prediction

Reward function我們使用的是BLEU

How do you improve your performance (3%)

Schedule sampling

- 考慮訓練與測試時，最大的不同就是正確答案的有無
  - 訓練時，decoding階段可以直接拿前一個字的正確答案拿來當作LSTM的輸入
  - 測試時，則只能從預測結果中，選機率最大的那一個字，當作是答案
- 為了縮小這個差距，或者說讓機器認知這一件事，我們在訓練時，有一定的機率，不直接拿取正確答案，而是拿前一步預測中機率最大的字，餵給LSTM當輸入
- 前10個epoch，暫時先不使用schedule sampling，第10個epoch之後，以20%的機率從前一步中選機率最大的字，80%從訓練集拿正確答案

Reward functions

- 不同的reward function，可以讓機器講出不一樣風格的對話
- 我們主要嘗試了兩種「reward function」
  1. 用BLEU-Score當作reward
     - 計算機器講的話 X 與真正答案的BLEU-Score r_b，以及baseline c_b
     - $\theta' = \theta + \eta \cdot (r_b - c_b) \cdot \frac{\partial \log(p(X))}{\partial \theta}$
     - 使用該batch的平均BLEL-score當作baseline c_b
     - 如果baseline < 0.05，就取0.05。因為當大家都講不好的時候，沒有必要把「講得比較沒那麼不好」的句子當作優秀的句子
     - 訓練的結果沒有很理想，在訓練集上模型能夠講出與訓練資料幾乎相同的句子，即BLUE-Score接近1，但在測試的時候，卻常常出現重複的詞彙
  2. 用句子長度當作reward
     - 使用「機器講的話 X 的長度除以最大長度 (20)」作為reward，baseline則設為0.5，即長度為10個字的句子
     - 一樣透過policy-gradient更新模型的參數
     - 我們期望透過這樣的reward，讓機器喜歡講長一點的句子
     - 同樣地結果沒有理想，句子確實有拉長，但機器通常只是在原本句子的尾端或中間，重複前幾個字，而不是真正有意義的長句子

Exposure bias

- 不論是Schedule sampling或是BLEU-Score reward，某些程度上都解決了exposure bias的問題
- Schedule sampling藉由在訓練時模擬測試的情境，降低了validation/test set與training set的損失差距
- BLEU reward因為是RL的框架，原本就是讓機器自己生句子，根據句子的好壞決定reward，因此在validation與training時的差距不至於過大

Experiment settings and observation (3%)

用句子長度當作「**reward**」

- 實驗結果

Input	s2s only	s2s + reward
hi	hi	i’m randy
would you like to dance with me	no	i know i did
let’s go to see a movie	okay	why
don’t be sad	i’ve never met anyone who won’t	can’t you come to us are you right here
i am sorry to hear that	well, what do you think	you are right here it s not the way it’s like a weeping weeping it’s weeping

- 觀察與說明
  1. 加上「reward」之後，句子的長度確實有增加，機器會嘗試講一些除了「yes/no」之外的回覆，雖然不見得是適合的答覆
  2. 但是機器為了拉長句子，有時候只是重複一些詞彙，例如最後一個範例，他只是重複了很多「weeping」，句子看起來雖然比較長，但明顯沒有比較好

是否使用「**schedule sampling**」

- 實驗結果

Input	s2s only	s2s + schedule-sampling
how is your dad	when he told my head he was out there	all right he 's on his way
my computer is broken	i 'm gon na have a responsibility	i 'm all available in my life
hurry up	we got ta get out of here	we got ta get back to our
you are so stupid	you think you 're better than me	you know n’t i 've seen your a time
i command you to stop	and i do n’t like nothing we 've already done	i do n’t do to anything else i want to hear
fuck you mother	do you have to stop her doing	just a let from holiday my son a bitch
we are leaving	we do n’t wan na be hiding	we 're not gon na get out

- 觀察與說明
  - 在訓練的時候，有「schedule-sampling」的版本在「validation set」上面的損失明顯較低，但生出來的句子差異度並不會太大
  - 也有可能是因為，我們選擇的句子在訓練集中都有類似的對話出現過，機器本來就步至於回答得太差勁，因此「schedule-sampling」的幫助有限

Team division (1%)

- 簡瑋德 - 實驗、報告
- 黃兆緯 - 模型設計、優化
- 劉岳承 - 模型設計、報告
- 李承軒 - 前處理、優化