

ADL HW1 report

學號:B04705043 資管四 張凱庭

- Q1 (1)使用NLTK套件中的word_tokenize() function 對dataset進行tokenize
(2)每筆training data 隨機取其中9個negative sample, 這樣加上positive sample 剛好10個options
(3)將每個speaker 的 utterance 用空白連接, utterances 將長度大於300以後的token去除, option則是將長度大於50以後的token去除
(4)使用 FastText crawl-300d-2M.vec 的 pre-trained embedding model

- Q2 (1)utterances 和 candidates 分別feed 進3 layers 的gru, 然後對結果進行max pooling得到兩個向量u 和 c, 再把u 和 c 輸入Bilinear layer得出utterances對於每個option(candidate)的score
(2)recall@10:

validation	kaggle public	kaggle private
0.585	0.625	0.675

(3)loss function: Binary cross entropy loss with logits

(4)optimization: adam

learning rate: 0.0001

batch size: 100

大約在epoch 10 收斂

- Q3 (1)utterances 和 candidates 分別輸入3 layers 的gru, 然後分別對 candidates結果的每個hidden state 和utterances的結果計算attention scores, 把這個attention scores和 candidates output concat, 然後輸入到下一個入3 layers 的gru。再將得到的結果, 做max pooling然後輸入到dnn得出utterances對於每個option(candidate)的score
(2)recall@10:

validation	kaggle public	kaggle private
0.59		

(未上傳kaggle)

(3)loss function: Binary cross entropy loss with logits

(4)optimization: adam

learning rate: 0.0001

batch size: 100

大約在epoch 15 收斂

- Q4 (1)utterances 和 candidates 分別輸入3 layers 的gru, 然後分別對 candidates結果的每個hidden state 和utterances的結果計算attention scores, 把這個attention scores和 candidates output concat, 然後輸入到下一個入3 layers 的gru。再將得到的結果還有utterances 做max pooling得到兩個向量u 和 c, 再把u 和 c 輸入Bilinear layer得

utterances對於每個option(candidate)的score

(2)recall@10:

validation	kaggle public	kaggle private
0.6298	0.71	0.73

(3)loss function: Binary cross entropy loss with logits

(4)optimization: adam

learning rate: 0.0001

batch size: 100

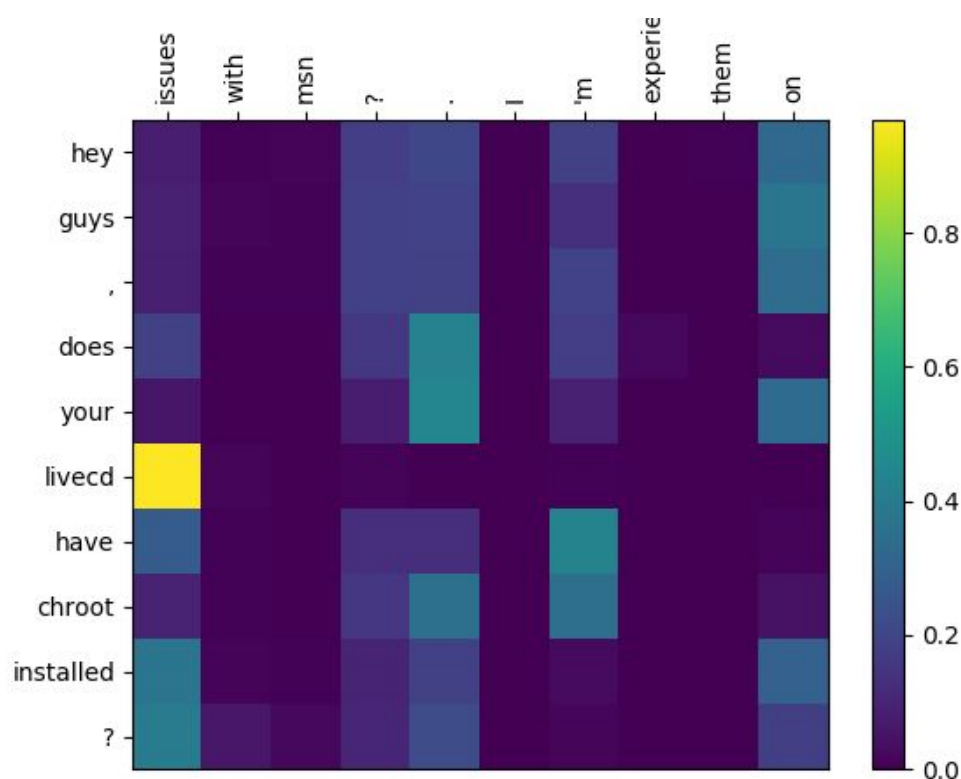
大約在epoch 10 收斂

(5)這個model 比單純將attention的結果直接dnn效果來得要好，也可以更快收斂

Q5 Compare GRU and LSTM models for the following properties (1%)

- the recall@10 score (on validation set or public leaderboard).
- required GPU memory.
- training, testing speed.

Q6



attention model 認定重要的單字會特別高分，其他不重要的字則分數十分平均

Q7 Compare training with different settings:

(1)不同數量negative sample 1:4和1:9

1:9訓練時每個epoch大約多花10min，每個epoch大約30min，而最終收斂的 valid recall@10大約高出0.05，因此最後都是採用1:9，雖然會花上比較多時間訓練，但效果較好。

(2)不同長度的utterances:

分別嘗試了只取 last utterances和暴力取全部的 utterance，取全部的valid recall@10高出了0.1左右，明顯效果比取last的好