r06922115
鄭皓謙

HW2 report

Task 2

task2 看起來如何利用sequence input很關鍵，先前嘗試了在task1使用的ALS，取前20個與user dot score最高的食物，結果非常爛0.00409。換個思路，希望RNN能夠有效利用sequence的特性，最後使用attention，配合幾個Attention is all you need論文裡提到的幾個技巧，達到0.30472/0.30506 ，意外的是加入user attribute，private分數居然下降了。這次作業都使用deep learning解決。

Food embedding

首先將吃過的食物當作categorical input，然後根據飲食紀錄，將每一天的飲食記錄下來，訓練資料裡最長的序列是165，所以每一天的飲食記錄變成categorical sequence，這裡使用implicit format，所以與飲食的次數無關，只要當天有吃過該次食物，那個食物的categorical entry就是1，所以每個user輸入變成[165, 5532]。沒有使用食物的attributes。

Attention mechanism

這次作業用了很多attention mechanism的部分，但是並沒有使用到RNN cell，所以我是如何保留sequence的前後資訊？ 我參考google論文attention is all you need，裡面提到的positional encoding，也直接使用文中提到sin/cos的方式為每個sequence加入位置的資訊。避免使用RNN cell則是因為我的機器跑recurrent實在太吃力。Scaled dot attention的部分也是根據論文中的建議。

User embedding

為了加入user attribute，除了age跟gender以外，還有豐富的free text資訊，文字資訊missing value的比例是0.28左右，我沒有選擇加入friend count以及location資訊。處理文字的部分我取 about_me ,reasons, inspirations三個欄位，直接串接，然後vocabulary取top 2k 詞頻的字，然後post padding到2000長度，由於平均長度跟標準差為567 , 737，配合glove pre-trained embedding on Wikipedia 2014 + Gigaword 5 。使用一個trainable weight對整個text sequence做scaled dot attention，然後表示user的text summary latent vector。最後輸出再加入age, gender，age 部分會做normalization，gender則是0/1 categorical attribute。

Sequence self-attention

首先輸入是食物的sequence，整段sequence Q是由[q1, q2, q2… q165]組成，self-attention就是加一層attention，讓每個 qi 對Q做scale dot attention，所以在每個時間點qi都可以對qi以前的輸入算一次attention，用意是讓Q的輸出有長距離的結構關係。要注意的是，這次的題目要根據飲食記錄預測隔天的飲食，所以在做self-attention時要加入mask，避免使用到qi取用到比index i 還大的資訊，這意味著使用了未來的資訊。

Classification

self-attention 的 input /output 都是sequence，然後再與user embedding的串起。然後透過幾層的FCN層，最後輸出每個時間點的飲食分數，於是輸出大小為[165, 5532]，算loss的時候再把padding部分去掉。

我把問題分成5532個one-for-all的分類問題，loss則是使用logistic error，除了最後一層linear，前面都是shared weight。但是這有一個問題是，每個食物最後輸出的分數不能直接比大小，假設只有三個食物 a, b, c，在經過sigmoid之後的輸出可能為[0.9, 0.7 0.3]，這並不代表user會選擇a的機率比b大。我嘗試加入positvie sample weight以及class weight去平衡不同類別及正負樣本的loss，但是結果失敗了，所以我還沒解決這個問

題，我仍然是取前20高分的當作輸出。我想RL也許能解決這個問題，把前面的部分當作feature extractor 然後MAP@20當作reward。

嘗試與心得

以上的嘗試最後都是一層的attention效果最好，我不確定是不是資料的分佈關係，而我使用過GRU cell做過，效果其實不錯已經有0.24，所以我想再加入attention應該就可以過baseline。另外還有衡量model的標準，由於是one-for-all classification，所以正負樣本數量差異很大， 所以accuracy比較不能看出差異，而loss因為每個user的飲食天數不固定，loss跳動也很大。訓練初期accuracy高達99.8，但是recall可能才0.001，這是由於模型全部猜不吃的關係，所以我衡量的標準使用F1 score，綜合考慮precision / recall。另外我希望模型在sequence的後面能夠預測得比較準，所以加入了linear的mask，去加強sequence越後面的loss。

以下是training的訓練圖分別是F1 score, precision, recall，validation則是取出幾個user不參與訓練的結果。

validation 的部分都低於training，但是沒有over fitting的現象。