作業 2 報告

一、各個.py 檔說明：

1.1 main.py(110522613.py)

訓練程式，在本文件中，pkl 檔的資料會先被讀入，然後做 padding 並打包成 tensor，隨後丟入模型中訓練。

1.2 data.py

該文件是 dataloader，裡面編寫了讀取資料的方法

1.3 utils.py

該文件定義了一些公共的變量和函式，以及訓練的基本參數。如 batch 大小、epoch 數量、學習率等等

1.4model.py

該文件定義了模型架構、loss function 以及 validation 的方法

1.5 make_dataset.py

該文件單獨執行，把 dataset 從 txt 的 data 文件用 pickle 打包成 pkl 的文件

1.6 make_vocab.py

該文件單獨執行，把 dataset 中不重複的所有單詞調處來組成詞典，用 pickle 生成 vocab.pkl 文件

1.7 make_vocab_rev.py

該文件單獨執行，通過 vocab.pkl 得出 vocab_rev.pkl，可以通過 encode 後的數字查詢原來是什麼字

1.7 submit.py

跑測試資料，生成帶有識別結果的 txt 文檔

二、資料集

2.1 字典的建立

本項目使用的字典參考了所有訓練資料，將所有訓練資料裡面涉及的詞從 0 開始依次編號。並存成 python 的 dict 格式，用 pickle 進行打包，保存到 vocab.pkl 文件。

此外將所有的分類分別編號並儲存成 dict 格式，以變量形式存在於程式碼中。

2.2 訓練資料的前處理

本項目的資料會被整理成 dict 形式。格式如下所示：

$$data = \begin{Bmatrix} 'data':[...] \\ 'label':[...] \end{Bmatrix}$$

資料的來源不限於題目中提供的內容，同時也引入了 github 上其他專案所使用的資料

集（url：https://github.com/kamalkraj/BERT-NER）。訓練時會將兩份資料做 merge 處理。

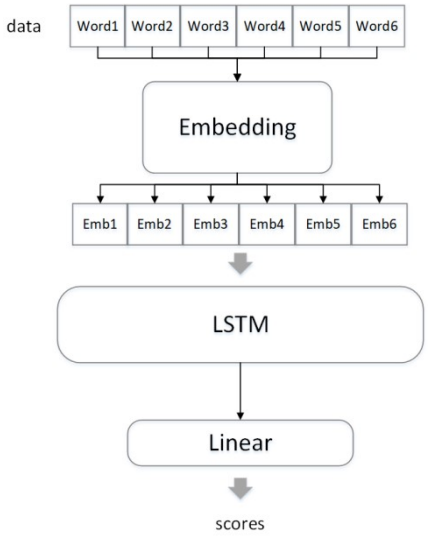
其中，本報告考慮到資料集中有很多並非是成分的內容，比如："-DOCSTART- -X- -X- O"。因此，本項目以空行為分界劃分句子，並捨棄了所有長度不足 5 的句子。

此後，將所有詞和標註分類替換成字典中的編號。

在輸入模型前，會將 batch 內所有的資料進行從大到小排序並做 padding，變成同樣長度的符號序列。

三、模型

該模型分為三部分，embedding、LSTM 和 linear 部分前後連接。示意圖如下所示：



輸入資料會對每一個字進行 one-hot 編碼，而輸出則是每一個字對應每一種預測結果的分數，最後取最大的預測結果。

本模型損失函數使用 cross entropy，優化器使用 Adam，學習率 0.001。

四、訓練

該項目訓練的過程中，每經過一個 epoch，會用測試資料對模型進行評估，算出一個測試資料的 loss 值。如果該 loss 值刷新了最低測試 loss 的記錄，系統就會保存當前的模型，反之則不會。

訓練資料的 loss 值呈現緩慢下降趨勢，而測試資料的 loss 先從 0.7 左右開始下降，直到第 28 個 epoch 降到 0.5736 的最低值後，就開始一路回升，在 70 個 epoch 以後甚至突破了 1.0。本人判斷在這之後出現了 overfitting 的問題，因此最終結果採用測試 loss 為 0.5736 的模型得到 test_submit.txt 文件。