應用漸進學習於文本分類之資料標註加速器

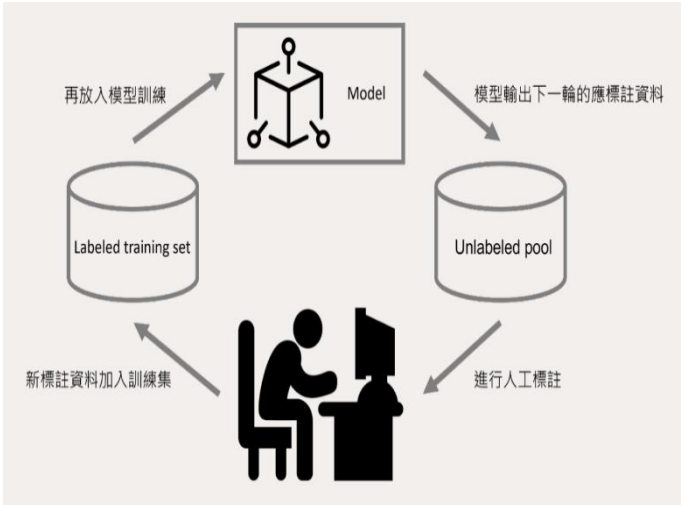
Incremental Learning on Data Annotation Accelerator in Text Classification Tasks

指導教授：高宏宇教授
專題成員：楊智翔、林恩締
開發工具：Python
測試環境：Colab

一、簡介：

動機：我們希望可以有一個模型可以幫助標記，節省人力和時間。
目標：減少需要標記的資料量。

為了驗證成功省下多少資料，我們會先從全部的資料集分出一個獨立的 test set，剩下的會先全部當作 training set 丟入訓練，用 test set 算出一個準確率之後，將此當作目標準確率，然後開始我們的流程，先標記一些資料丟入模型訓練後，等模型選出下一批要標記的資料後，再人工標記那些資料再丟回去給模型，然後用 test set 去驗證準確率後，重複以上步驟，直到達到目標準確率。



用兩種模型：1.Bert 2.Word embedding + LSTM
用三種策略：1. Least Confident 2. Margin sampling 3. Entropy
(每次會改變選取的數字區間)

二、測試結果：

資料一：Kaggle Competition：Natural Language Processing with Disaster Tweets
模型：Bert 預訓練模型：bert-base-uncased
Fine-tuning 階段：BertForSequenceClassification
策略：比較 Least Confident 和 Entropy 在這個資料集的表現

訓練集資料量 利用:Least Confident	測試集的 fl_score	訓練集資料量 利用:Entropy	測試集的 fl_score
300	0.7414	300	0.7414
2077	0.7870	2482	0.7935
2658	0.7832	3350	0.7866
3119	0.7931	4163	**0.7809**
3459	0.7935		
3814	0.7995		
4152	**0.8061**		

比較結果：用 Least Confident 的方法比 Entropy 適合這個資料集。
一開始丟入全部 6613 筆資料當 training set 進行訓練 f1_score 可以達到 0.81，最終用 Least Confident 的策略選出其中 4152 筆訓練資料也達到相同的分數，省下 37.2%的標記量。

資料二：網路爬蟲新聞標題 (使用兩種不同的模型)
1.模型：Word embedding + LSTM (資料前處理用 jieba 分詞)
策略：前三次用 Margin sampling，後兩次用 Entropy

一開始丟入 1600 筆資料當 training set 進行訓練準確率可以達到 83%，最終用 1139 筆當訓練資料就達到相同的準確率，省下 28.8%的標記量。

丟進模型訓練的資料量	從模型輸出的結果去選取	Random選取同樣筆數
100	57.9%	57.9%
150	61.4%	62.8%
210	59.2%	65.6%
338	72.8%	69.4%
925	81.5%	79.3%
1139	**83.0%**	**80.6%**

2.模型：Bert
預訓練模型：bert-base-chinese
Fine-tuning 階段：BertForSequenceClassification
策略：Least Confident

一開始丟入全部 2000 筆資料當 training set 進行訓練準確率可以達到 94.0%，最終用 979 筆當訓練資料就達到相同的準確率，省下 51.0%的標記量。

丟入模型訓練的資料量	Select準確率	Random準確率	選取的機率區間
100	88.9%	88.9%	30%-75%
338	92.9%	91.8%	30%-86%
517	93.3%	92.3%	30%-90%
629	93.4%	92.1%	30%-93%
718	93.5%	93.0%	30%-95%
812	93.7%	93.2%	30%-97%
894	93.4%	93.1%	30%-98%
979	**94%**	**93.3%**	