DSP HW3

B05902001 資工三 廖彥綸

一、架設環境

資工系工作站(i686-m64)

二、檔案說明

result1/ 使用 SRILM disambig 產生的結果

hw3.sh 第一部分的 shell code，需要有原始的資料檔 corpus.txt、lm.cnt、separator_big5.pl、Big5-ZhuYin.map、未分割的 testdata/X.txt、執行檔 ngram-count(來自 SRILM)輸出 seg_corpus.txt、testdata/seg_X.txt、bigram.lm、ZhuYin-Big5.map 和 result1 下的結果。(這次不會使用)

hw3_my.sh 第二部分的 shell code，配置和上述相似 (這次不會使用)

Makefile 作業主要執行的檔案

mapping.py 執行 Big5-ZhuYin.map 到 ZhuYin-Big5.map

mydisabig.cpp

三、執行方式

1. bigram.lm, Big5-ZhuYin.map, testdata 放到目錄下
2. make MACHINE_TYPE=$機器型號 SRIPATH=$srilm-1.5.10 的路徑 all
3. make map(目錄下有 Big5-ZhuYin.map 時輸出 ZhuYin-Big5.map)
4. make MACHINE_TYPE=$機器型號 SRIPATH=$srilm-1.5.10 的路徑 run (在 result2/ 下產生結果)

四、其他

1. Makefile 中已提供 result2/下的預測結果，testdata 目錄下必須要有檔案 X.txt (X = 1 to 10)。若要直接執行 mydisabig.cpp 以下列格式

./ mydisabig -text $file -map $map -lm $LM -order $order > $output

2. mydisabig 和 disabig 輸出的內容相似度約 90%，未能做到完全模仿，正確度比較則各有對錯。
3. mapping.py 以 python 撰寫可以直接得到 ZhuYin-Big5.map

python3 mapping.py

五、mapping.py

1. 讀寫使用 encoding = big5hkscs 如果只使用 big5 會有部分字遺失。
2. 創建字典，字或注音為 key，整段可能結果為 value。

六、mydisabig.cpp

1. 使用 Ngram.h 函式庫讀入 bigram.lm
2. 建立 map，讀入 ZhuYin-Big5.map，字或注音為 key，整段可能結果為 value 的 vector，同時去除 tab、空白鍵、換行符號等。
3 讀入 testdata，繪出所有可能性(注音符號用一串 vector 表示)。
4. 應用 Viterbi 尋找路徑。
5. 應用 Viterbi 的 backtrack，找出最大機率的路徑。
6. 回溯的路徑放入 stack 再依序印出。