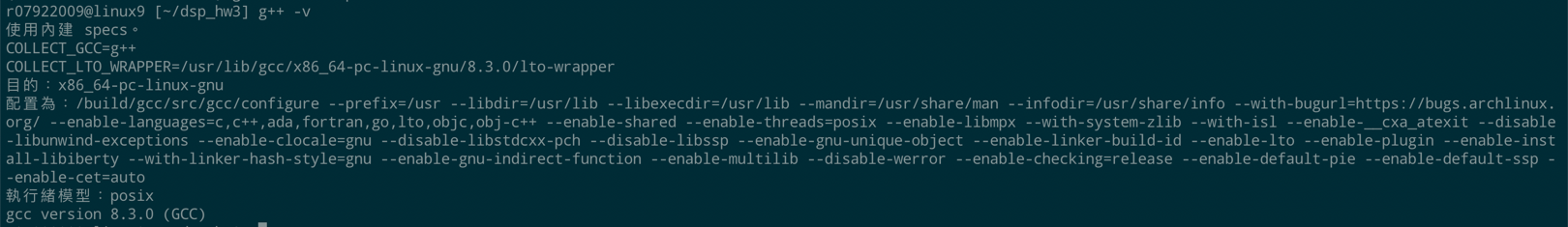
數位語音處理概論作業三

資工所碩一 R07922009 丁國騰

1. 環境：

本次作業是在CSIE workstation完成，使用 linux9.csie.ntu.edu.tw 工作站（gcc版本為8.2.1）



二、How to “compile” your program

將Makefile中的SRIPATH與MACHINE\_TYPE設定好，使用make 指令（即「輸入make於終端介面」）即可，或也可以在設定好環境變數的情況下使用以下指令執行：

g++ -O3 -I$(SRIPATH) -w --std=c++11 -c mydisambig.cpp

g++ -o mydisambig mydisambig.o $(SRIPATH)/lib/$(MACHINE\_TYPE)/liboolm.a $(SRIPATH)/lib/i$(MACHINE\_TYPE)/libdstruct.a /$(SRIPATH)/lib/i$(MACHINE\_TYPE)/libmisc.a

1. How to “execute” your program
2. make map: 執行 mapping.py，產生ZhuYin-Big5.map

python mapping.py Big5-ZhuYin.map ZhuYin-Big5.map

1. make build\_lm: 使用srilm提供的ngram-count對已分割過的檔案製造出bigram.lm

perl separator\_big5.pl corpus.txt > corpus\_seg.txt

$(SRIPATH)/bin/$(MACHINE\_TYPE)/ngram-count -text corpus\_seg.txt -write lm.cnt -order 2

$(SRIPATH)/bin/$(MACHINE\_TYPE)/ngram-count -read lm.cnt -lm bigram.lm -unk -order 2

如果沒有對txt進行分割請先執行mv testdata testdata\_old、perl separator\_big5.pl testdata\_old/${i}.txt > testdata/${i}.txt

1. make test: 使用srilm提供的disambig解析出答案，並存於$(TESTDIR)，預設為result1

[ -d $(TESTDIR) ] || mkdir -p $(TESTDIR);

@for i in $(shell seq 1 10); do \

$(SRIPATH)/bin/$(MACHINE\_TYPE)/disambig -text $(TESTDATA)/$${i}.txt -map $(TO) -lm $(LM) -order 2 > $(TESTDIR)/$${i}.txt; \

done;

1. make run: 執行 mydisambig 並把結果存於$(MYDISAMBIGDIR)，預設為result2

[ -d $(MYDISAMBIGDIR) ] || mkdir -p $(MYDISAMBIGDIR);

for i in $(shell seq 1 10) ; do \

echo "Running $$i.txt"; \

./mydisambig -text testdata/$$i.txt -map $(TO) -lm $(LM) -order 2 > $(MYDISAMBIGDIR)/$$i.txt; \

done;

1. make clean: 清除之前 make 產生的檔案

rm -f mydisambig.o mydisambig

1. What you have done

我按照課程網站的Q&A修正了makefile，並且以SRILM將注音轉為國字

並在mydisambig中實作了viterbi演算法，並對其結果與SRILM的結果比對