



Digital Speech Processing Homework 3

Nov. 16 2016

ro4922167@ntu.edu.tw 王育軒

To complete the homework, you need to...

- Build a character-based language model with toolkit SRILM.
- Decode the ZhuYin-mixed sequence

Outline

- Introduction
- SRILM
- Step by Step
- Submission and Grading

Introduction

讓他十分ㄟ怕
只ㄟ望ㄟ己明ㄟ度別再這ㄟㄟ命了
演一ㄟ樂產一ㄟ人積ㄟ出型提ㄟ競爭ㄟ

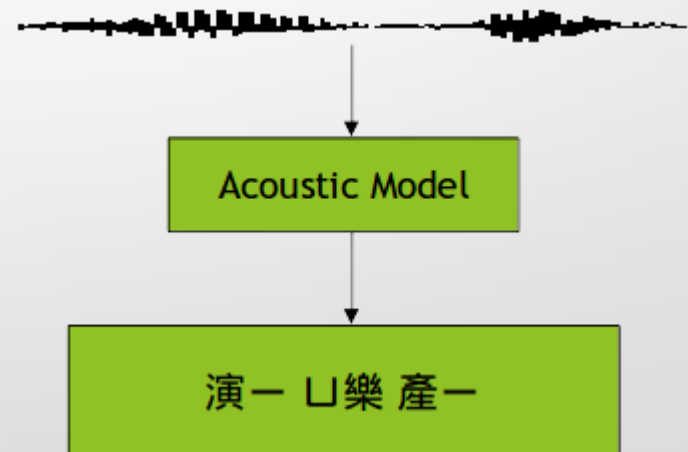
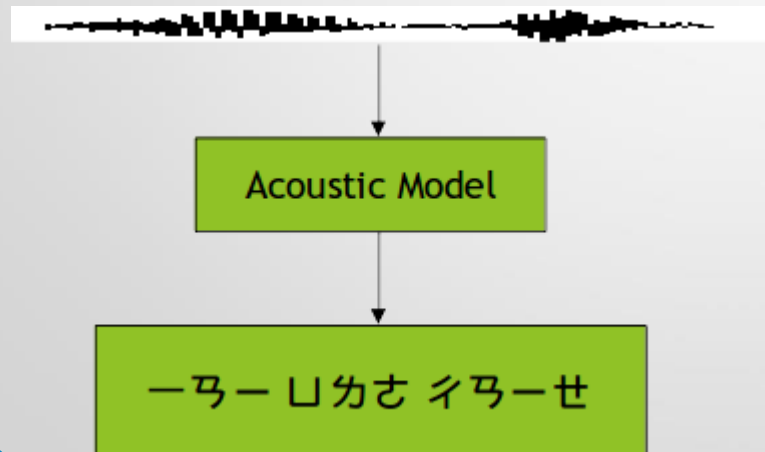


HW3：注音文修正

讓他十分害怕
只希望自己明年度別再這麼苦命了
演藝娛樂產業加入積極轉型提升競爭ㄟ

Introduction

- Imperfect acoustic models with phoneme loss.
- The finals of some characters are lost.



Introduction

- Proposed methods:

- Reconstruct the sentence by **language model**.

- For example, let $Z = \text{演一 口樂 產一}$

$$W^* = \arg \max_W P(W | Z)$$

$$= \arg \max_W \frac{P(W)P(Z | W)}{P(Z)}$$

$P(Z)$ is independent of W

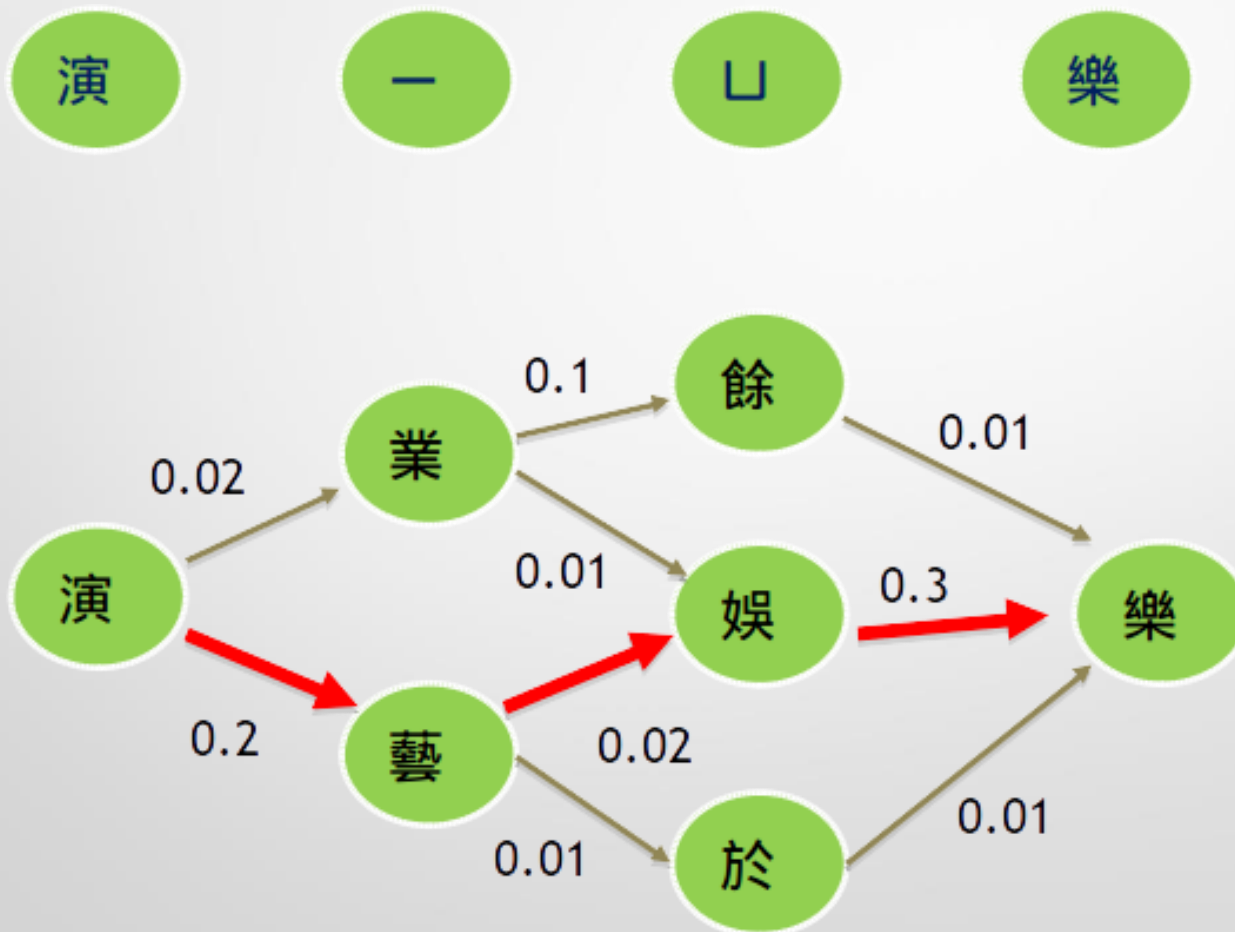
$$= \arg \max_W P(W)P(Z | W)$$

$W = w_1 w_2 w_3 w_4 \dots w_n$, $Z = z_1 z_2 z_3 z_4 \dots z_n$

$$= \arg \max_W \left[P(w_1) \prod_{i=2}^n P(w_i | w_{i-1}) \right] \left[\prod_{i=1}^n P(z_i | w_i) \right]$$

$$= \arg \max_{W, P(Z|W) \neq 0} \left[P(w_1) \prod_{i=2}^n P(w_i | w_{i-1}) \right] \quad \text{Bigram language model}$$

Example



SRILM

- SRI Language Model toolkit
 - <http://www.speech.sri.com/projects/srilm/>
- A toolkit for building and applying various statistical language models
- Useful C++ classes
- Using/Reproducing some of SRILM

SRILM

- Build it from source code (Provided on course website)
 - Allows you to use SRILM library
- Or download the executable from the course website to finish the first part of HW3
 - Different platform:
 - i686 for 32-bit GNU/Linux
 - i686-m64 for 64-bit GNU/Linux (CSIE workstation)
 - Cygwin for 32-bit Windows with cygwin environment

SRILM

- You are **strongly recommended** to read FAQ on the course website
- Possibly useful codes in SRILM
 - \$SRIPATH/misc/src/File.cc (.h)
 - \$SRIPATH/lm/src/Vocab.cc (.h)
 - \$SRIPATH/lm/src/ngram.cc (.h)
 - \$SRIPATH/lm/src/testError.cc (.h)

SRILM

- Big5 Chinese Character separator written in perl:
 - perl separator_big5.pl corpus.txt > **corpus_seg.txt**
 - Why we need to separate it? (Use char or word?)

1 國民黨 立委 帶領 支持者 參加 升旗 心情 百感交集
2 多位 中國國民黨 籍 立法委員今天 一大早 帶領 支持者 到 總統府
3 在國民黨 失去 政權 後 第一次 參加 元旦 總統府 升旗典禮
4 有立委 感慨 國民黨不 團結 才會 失去 政權
5 有立委 則 猛 批 總統 陳水扁
6 人人 均 顯得 百感交集



7	國民	黨籍	1	國	民	黨	立	委	帶	領	支	持	者	參	加	升	旗	心	情	百	感	交	集								
8	到	總	統	府	2	多	位	中	國	國	民	黨	籍	立	法	委	員	今	天	一	大	早	帶	領	支	持	者	到	總	統	府
9	潘	維	剛	第	3	在	國	民	黨	失	去	政	權	後	第	一	次	參	加	元	旦	總	統	府	升	旗	典	禮			
10	新	世	紀	的	第	4	有	立	委	感	慨	國	民	黨	不	團	結	才	會	失	去	政	權								
11	這	一	年	來	5	有	立	委	則	猛	批	總	統	陳	水	扁															
12	她	沒	想	到	6	人	人	均	顯	得	百	感	交	集																	
13	丁	守	中	第	7	國	民	黨	籍	立	委	潘	維	剛	丁	守	中	蔡	家	福	關	沃	暖	洪	讀	李					
14	陳	總	統	應	8	到	總	統	府	前	參	加	升	旗	典	禮															
				9	潘	維	剛	表	示																						
				10	新	世	紀	的	第	一	天	參	加	升	旗	典	禮	讓	她	百	感	交	集								
				11	這	一	年	來	政	局	像	雲	霄	飛	車	般	起	伏	不	知	何	時	能	落	地						
				12	她	沒	想	到	政	權	改	變	影	響	會	這	麼	大													
				13	丁	守	中	表	示																						
				14	陳	總	統	應	該	立	即	拿	出	具	體	政	策	打	開	兩	岸	僵	局								

SRILM

- `./ngram-count -text corpus_seg.txt -write lm.cnt -order 2`
 - `-text`: input text filename
 - `-write`: output count filename
 - `-order`: order of ngram language model
- `./ngram-count -read lm.cnt -lm bigram.lm -unk -order 2`
 - `-read`: input count filename
 - `-lm`: output language model name
 - `-unk`: view OOV as `<unk>`. Without this, all the OOV will be removed

Example

corpus_seg.txt

在國民黨失去政權後第一次參加元旦總統府升旗典禮
有立委感慨國民黨不團結才會失去政權
有立委則猛批總統陳水扁
人人均顯得百感交集



lm.cnt

夏	11210
俸	267
鵠	7
祇	1
微	11421
檣	27
.....	



(log probability)

bigram.lm

\data\

ngram 1=6868

ngram 2=1696830

\1-grams:

-1.178429 </s>

-99 <s> -2.738217

-1.993207 一 -1.614897

-4.651746 乙 -1.370091

.....

(backoff weight)

SRILM

- `./disambig -text $file -map $map -lm $LM -order $order`
 - -text: input filename
 - -lm: input language model
 - -map: a mapping from (注音/國字) to (國字)
 - You should generate this mapping by yourself from the given Big5-ZhuYin.map.
 - **DO NOT COPY-PASTE TO RUN THIS LINE**

Big5-ZhuYin -> ZhuYin-Big5

Big5-ZhuYin.map

一 一 ˊ / 一 ˋ / 一 _
乙 一 ˋ
丁 ㄉㄣˊ _
柒 ㄘㄣˊ _
乃 ㄋㄠˇ
玖 ㄐㄠˇ
...
...
長 ㄌㄤˊ ㄉㄠˊ / ㄘㄠˊ
行 ㄒㄩㄥˊ ㄉㄠˊ / ㄈㄠˊ
...



ZhuYin-Big5.map

ㄅ ㄆ ㄇ ㄉ ㄋ ㄆ ㄆ ㄆ ...
ㄆ ㄆ
ㄇ ㄇ
ㄉ ㄉ
...
...
ㄆ ㄆ ㄆ ㄆ ㄆ ㄆ ㄆ ㄆ ...
ㄆ ㄆ
ㄆ ㄆ
...
...

- Be aware of polyphones(破音字)
- There could be arbitrary spaces between all characters.
- Key - value pairs
- Can be random permutation

Step by Step

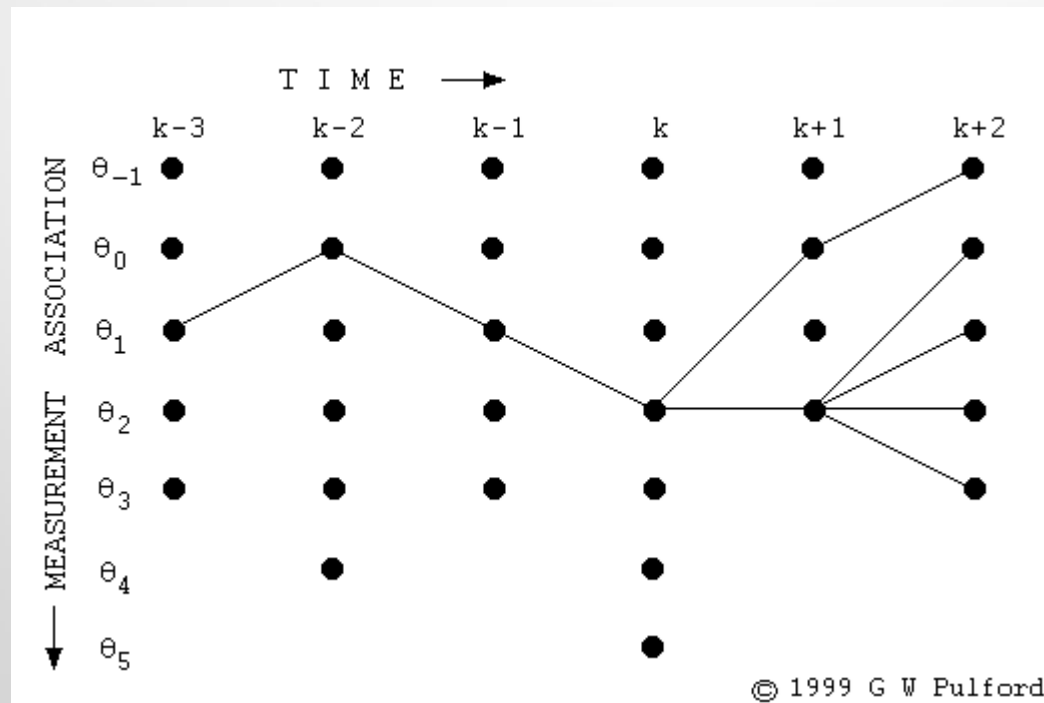
- Segment corpus and all test data into characters
 - `./separator_big5.pl corpus.txt corpus_seg.txt`
 - `./separator_big5.pl <testdata/xx.txt> <testdata/xx.txt>`
 - You should rename the segmented testdata as `testdata/1.txt`, `testdata/2.txt`... and use them in the following task
- Train character-based bigram LM
 - Get counts:
 - `./ngram-count -text corpus_seg.txt -write lm.cnt -order 2`
 - Compute probability:
 - `./ngram-count -read lm.cnt -lm bigram.lm -unk -order 2`
- Generate ZhuYin-Big5.map from Big5-ZhuYin.map
 - See FAQ 4

SRILM disambig

- Using disambig to decode testdata/xx.txt
 - `./disambig - text $file - map $map - lm $LM - order $order > $output`

My Disambig

- Implement your version of disambig
- Use dynamic programming(Viterbi)
- The vertical axes are candidate characters



Tips

- C++ is **strongly** recommended
 - Speed
 - SRILM compatibility and utility
 - You must provide **Makefile** for execution
(See. Evaluation Procedure for details)
- Dual OS or VirtualBox with Ubuntu **strongly** recommended
- **Your output format should be consistent with SRILM**
 - `<s>` 這是一個範例格式 `</s>`
 - There are an `<s>` at the beginning of a sentence, a `</s>` at the end, and whitespaces in between all characters.

How to deal with Chinese char?

- Chinese character: You should use **Big5** encoding
- All testing files are encoded in **Big5**
- A Chinese character in Big5 is always 2 bytes, namely, **char[2]** in C++

Submission Example: student ID: ro4922167

- dsp_hw3_ro4922167.zip
- When unzipped, your uploaded file should contain a directory as following:
 - dsp_hw3_ro4922167/
 - result1/1.txt~10.txt (generated from SRILM disambig with your LM by yourself)
 - your codes
 - Makefile
 - report.pdf
- **Don't** hw3_Ro4922167, HW3_ro4922167, hw3_ro4922167/Result1, hw3_ro4922167/best_result1, hw3_ro4922167/result1/segmented_1.txt...

Submission

- Your report should include:
 - Your environment (CSIE workstation, Cygwin, ...)
 - How to “compile” your program
 - How to “execute” your program
 - Not familiar with makefile is fine, tell me how to execute your program
 - **However, you should also strictly follow the spec** (regulations about filenames, input files and output files)
 - ex: `./program -a xxx -b yyy`
 - What you have done
 - **NO** more than two A4 pages.

Grading

- (10%) Your code can be successfully compiled
- (10%) Correctly generate ZhuYin-Big5.map
- (30%) Correctly use SRILM disambig to decode ZhuYin-mixed sequence
- (10%) mydisambig program can run with no errors and crashes
- (25%) Your results decoded by your own program are the same as expected
- (10%) Your report contains required information
- (5%) You strictly follow format regulation
- **(10% bonus!)** Your program can support trigram language models **with speed pruning.**

If there are errors during TA's testing

- Like compilation error, crash...
 - TA will ask you to demo your program **only** with the files you uploaded.
 - If you can prove that you followed the runles correctly, you will get your credits.

Evaluation Procedure

- There are some files provided by TA **but you shouldn't upload them**
 - Big5-ZhuYin.map, bigram.lm...
 - **Strictly follow regulations about format**
 - However, you can utilize the files in makefile
- **test_env shows locations of files during evaluation**
- **In the following slides, this color specify makefile commands of evaluation scripts**

Evaluation Procedure

- Initialization
 - `make clean`
 - copy ta's bigram.lm, Big5-ZhuYin.map, testdata to your directory
- (10%) Your code can be successfully compiled.
 - `make MACHINE_TYPE=i686-m64 SRIPATH=/home/ta/srilm-1.5.10 all`
 - i686-m64 is TA's platform
 - Your code should be **machine-independent**(`system("pause")` is invalid in **my system**) and the user can easily specify the platform and SRILM path
- (10%) Correctly generate ZhuYin-Big5.map
 - `make map` (it should generate hw3_ro4922167/ZhuYin-Big5.map)
 - check if **hw3_ro4922167/ZhuYin-Big5.map** is correct
 - (You have to write your own makefile to achieve it. Generation must be based on **hw3_ro4922167/Big5-ZhuYin.map**)
 - (Your output in this step should be **hw3_ro4922167/ZhuYin-Big5.map**)
 - (python/perl/C/C++/MATLAB/bash/awk permitted)

Evaluation Procedure

- (30%) Correctly use SRILM disambig to decode ZhuYin-mixed sequence
 - Check if result1/1.txt~10.txt is the same as expected
- (10%) mydisambig program can run with no errors and crashes
 - `make MACHINE_TYPE=i686-m64 SRIPATH=/home/ta/srilm-1.5.10 LM=bigram.lm run;`
 - (it should run based on **bigram.lm** and generate result2/1.txt~10.txt)
- (25%) Your results decoded by your own program are the same as expected
 - check result2/1.txt~10.txt
 - **TA's testdata will be segmented testdata, not the given raw testdata**

Late Penalty

- 10% each 24 hours, according to the announced deadline instead of the deadline on Ceiba
- 100 -> 90 -> 80, not 100 -> 90 -> 81

Notes

- Follow the spec!!!!
- All of your program should finish the tasks assigned below 10 minutes
- Totally checking the correctness with good documents is YOUR JOB
- Only the latest files you uploaded to ceiba will be evaluate (All of your previous uploaded version will be ignored)

Makefile example

```
# The following two variable will be commandline determined by TA
# For testing, you could uncomment them.
SRIPATH ?= /data/DSP_HW3/103_2/srilm-1.5.10
MACHINE_TYPE ?= i686-m64
LM ?= bigram.lm

CXX = g++
CXXFLAGS = -O3 -I$(SRIPATH)/include -w
vpath lib%.a $(SRIPATH)/lib/$(MACHINE_TYPE)

TARGET = mydisambig
SRC = mydisambig.cpp
OBJ = $(SRC:.cpp=.o)
TO = ZhuYin-Big5.map
FROM = Big5-ZhuYin.map
.PHONY: all clean map run

all: $(TARGET)

$(TARGET): $(OBJ) -loolm -ldstruct -lmisc
    $(CXX) $(LDFLAGS) -o $@ $^

%.o: %.cpp
    $(CXX) $(CXXFLAGS) -c $<

run:
    @#TODO How to run your code toward different txt?
    @for i in $(shell seq 1 10) ; do \
        echo "Running $$i.txt"; \
        ./mydisambig -text testdata/$$i.txt -map $(TO) -lm $(LM) -order 2 > result2/$$i.txt; \
    done;

map:
    @#TODO How to map?
    @echo "Mapping!"
    @#./mapping $(FROM) $(TO)
    @#matlab < mapping.m ;
    @#python mapping.py $(FROM) $(TO)
    @#sh mapping.sh $(FROM) $(TO)
    @#perl mapping.pl Big5-ZhuYin.map ZhuYin-Big5.map

clean:
    $(RM) $(OBJ) $(TARGET)
```

Reminders and Suggestions

- Read the spec carefully
 - Finish the first part (SRILM disambig) as early as possible
 - If everything goes well, you should finish the first part in an hour
 - Fix the issue of dependencies early
 - Big5 encoding issue
 - Be sure that you prepare the correct Makefile
 - Evaluation procedure is in part automatically done by scripts. You can see the details in the previous slides
 - See the **FAQ** in the website
 - Contact TA if needed
 - **Check [email-FAQ!](#)**
 - ro4922167@ntu.edu.tw 王育軒
- TA **will not** help you debug your program