請繳交
1.ZhuYin-utf8.map，及其程式。
2.應用SRILM所建立的語言模型 分別將三個文件 (test1_seg.txt, test2_seg.txt, test3_seg.txt) 內的注音翻譯為國字。　繳交每個文件翻譯後的結果。

作業15 語言模型

使用SRILM工具**建立**語言模型，並**應用**此語言模型

SRILM

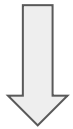
- SRILM (Stanford Research Institute Language Modeling Toolkit)
- SRILM是用來建構和應用**統計語言模型**
- 其他的統計語言模型工具:IRSTLM、MITLM、 BerkeleyLM

作業介紹

- 使用Python建立注音對應國字的 對照檔 (ZhuYin-utf8.map)，以及使用SRILM建立的語言模型，來實現注音文翻譯的功能。

Ex

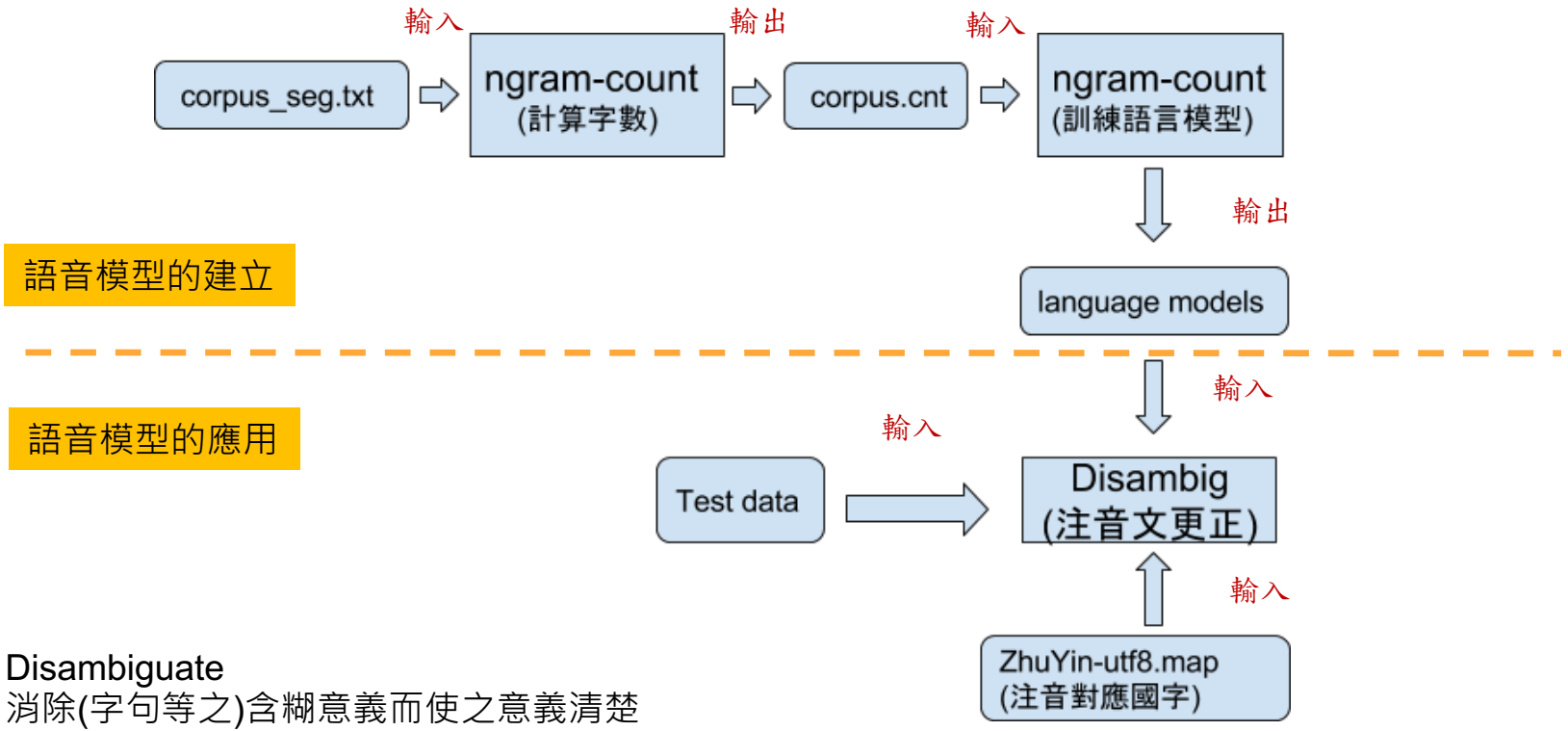
羅力的小ㄋ兒在上上週ㄌ出生，向ㄑ隊請假返ㄇ



羅力的小女兒在上上週六出生，向球隊請假返美

由於訓練的corpus不夠大，所以我們提供待求字的**注音線索**，用以提高正確率。　如果沒有輔助的線索當作限制條件（即，只是□），所有的中文字都是可能的答案。

Flow Chart



Disambiguate
消除(字句等之)含糊意義而使之意義清楚

SRILM 教學

ngram-count 指令介紹

- 指令ngram-count的功能有兩項：1.計算N-grams counts，以及 2.訓練語言模型。

- 輸入ngram-count -help 可查看參數說明

```
user@user-PC /tmp/SRILM
$ ngram-count -help
Usage of command "ngram-count"
 -version:                print version information
 -order:                  max ngram order
                Default value: 3
 -varprune:               pruning threshold for variable order ngr
                Default value: 0
 -debug:                  debugging level for LM
                Default value: 0
 -recompute:              recompute lower-order counts by summatio
 -sort:                   sort ngrams output
 -write-order:            output ngram counts order
                Default value: 0
 -tag:                    file tag to use in messages
 -text:                   text file to read
 -text-has-weights:       text file contains count weights
 -no-sos:                 don't insert start-of-sentence tokens
 -no-eos:                 don't insert end-of-sentence tokens
 -read:                   counts file to read
```

← 這裏我們Windows OS下，
利用Cygwin在來模擬Linux OS

- Reference: **http://www.speech.sri.com/projects/srilm/manpages/ngram-count.1.html**

1. 計算N-gram counts

➢ ngram-count -text corpus_seg.txt -write corpus.cnt -order 2

- -text: 指定**來源檔** (即corpus_seg.txt)。
- -write: 設定**輸出檔名**。
- -order: 設定N-grams counts的order，若沒設定則預設為3 (trigram)。

註:當order 設定為n時，輸出的N-gram counts 會包含unigram、bigram、…、n-gram。

2. 訓練語言模型

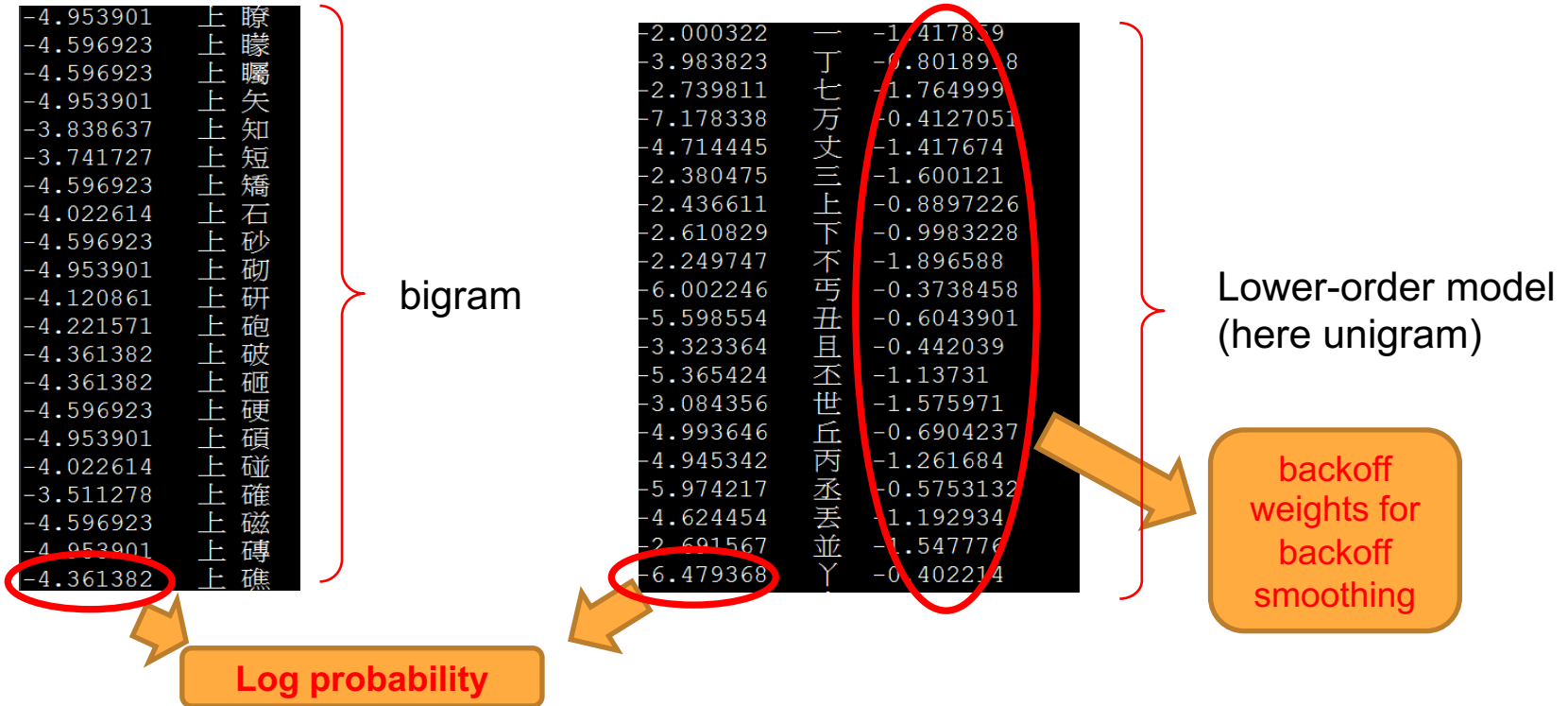
- ngram-count -read corpus.cnt -lm bigram.lm -order 2

- -read:指定N-gram counts的來源檔 (即步驟1所產生的corpus.cnt) 。
- -lm: 指定輸出的語言模型檔名。
- -order:設定language models的order。

註:當order 設定為n時， 輸出的language models會包含unigram、bigram、…、n-gram 。

Estimated Language Models

訓練後的language models (bigram.lm) 如下圖，其中機率是使用log base10來表示。



disambig指令介紹

- disambig指令功能：使用N-gram language model來消除歧異字(注音文修正) -- 消除(字句等之)含糊意義；使意義清楚。

- 輸入disambig -help 可查看參數說明

```
user@user-PC /tmp
$ disambig -help
Usage of command "disambig"
 -version:              print version information
 -lm:                   hidden token sequence model
 -use-server:           port@host to use as LM server
 -cache-served-ngrams: enable client side caching
 -count-lm:             use a count-based LM
 -factored:             use a factored LM
 -bayes:                context length for Bayes mixture LM
                Default value: 0
 -bayes-scale:          log likelihood scale for -bayes
                Default value: 1
 -mix-lm:               LM to mix in
 -lambda:               mixture weight for -lm
                Default value: 0.5
 -mix-lm2:              second LM to mix in
 -mix-lambda2:          mixture weight for -mix-lm2
                Default value: 0
 -mix-lm3:              third LM to mix in
 -mix-lambda3:          mixture weight for -mix-lm3
                Default value: 0
 -mix-lm4:              fourth LM to mix in
 -mix-lambda4:          mixture weight for -mix-lm4
                Default value: 0
 -mix-lm5:              fifth LM to mix in
 -mix-lambda5:          mixture weight for -mix-lm5
                Default value: 0
```

- Reference: http://www.speech.sri.com/projects/srilm/manpages/disambig.1.html

disambig

- disambig -text $file -map $map -lm $LM -order $order > result.txt

- -text : 輸入**需修正的檔案** (e.g., $file → test1.txt)
- -map: a mapping from Y (e.g, 注音/國字) to X (e.g. 國字)
  (e.g., $map → ZhuYin-utf8.map)
- -lm: 輸入的語言模型 (e.g., $LM → bigram.lm)
- -order:設定language models的order (e.g., $order → 2)
- > :指向 儲存輸出結果 的檔案 (e.g., result.txt)

作業步驟

Step by Step

- utf8-ZhuYin.map是 以國字為索引對應到 其注音 的對應檔。
- 現在需要將utf8-ZhuYin.map 轉換為：以每個character (含國字及注音)為索引 來對應 (將之取名為：ZhuYin-utf8.map)。

A. 利用corpus_set.txt訓練數據集 來訓練 bigram 語言模型

1. N-gram counts :

```
ngram-count  -text corpus_seg.txt  -write corpus.cnt  -order 2
```

2. Estimate language models:

```
ngram-count  -read corpus.cnt   -lm bigram.lm   -order 2
```

B. 利用我們給的 國字對應注音的 對照檔utf8-ZhuYin.map 產生 注音對應國字的 對照檔 ZhuYin-utf8.map

(字 對應 注音 → 注音 對應 字 (如下頁說明)，**可使用Python實現**)

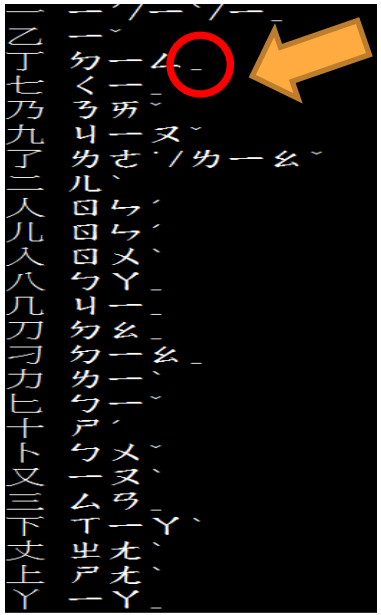
C. 使用disambig對testdata進行歧義字修正(注音文修正)

```
disambig -text $file -map ZhuYin-utf8.map -lm $LM -order $order > $output
```

utf8-ZhuYin.map

破音字：行 ㄒㄧㄥˊ/ㄏㄤˊ會對應到兩個ZhuYin

utf8-ZhuYin.map:



_ 代表第１聲

ZhuYin-utf8.map (注意每一個字之間都有空格)

索引 對應的字

ㄅ 八 匕 卜 不 卞 巴 比 丙 包 …
八 八
匕 匕
卜 卜
…

→ 注音對應到國字

SRILM也要求每個character (國字)
都要有對應的索引

ㄅ ㄅ 不需要

ㄆ 仆 匹 片 丕 回 平 扒 扑 疋 …
仆 仆
匹 匹
片 片
…

ㄦ 二 而 耳 兒 洱 貳 爾 餌 邇 …
二 二
而 而
耳 耳
…

Reference

HomeWork of Digital Speech Processing, Lin-shan Lee