# Project 4 - Error Detection

*Hongru Liu hl3148*

## Step 0: environment setting

### Load required packages

```
if(!require("tm")){
  install.packages("tm")
}
```

```
## Loading required package: tm
```

```
## Loading required package: NLP
```

```
if(!require("qdap")){
  install.packages("qdap")
}
```

```
## Loading required package: qdap
```

```
## Loading required package: qdapDictionaries
```

```
## Loading required package: qdapRegex
```

```
## Loading required package: qdapTools
```

```
## Loading required package: RColorBrewer
```

```
##
## Attaching package: 'qdap'
```

```
## The following objects are masked from 'package:tm':
##
##     as.DocumentTermMatrix, as.TermDocumentMatrix
```

```
## The following object is masked from 'package:NLP':
##
##     ngrams
```

```
## The following object is masked from 'package:base':
##
##     Filter
```

```
library(tm) # VCorpus
library(qdap) # clean
# library(purrr)
# library(ngram)
# library(NLP)
# library(reshape2)
```

### Specify directories

Please change the working directory to the folder containing this Rmd file.

```
setwd("/Users/hongru/Documents/GitHub/Fall2018-Project4-sec1proj4_grp6/Github/doc")
## setwd("../Fall2018-Project4-sec1proj4_grp6/Github/doc")
## use relative path for reproducibility
```

## Step 1: construct a positional binary matrix using ground truth texts

*Note that for the final result, we construct a positional binary list of letter bigrams instead of a matrix. But the former one could be used in the same way as the latter one.*

Call all the required customized functions from the library.

```
source("../lib/get_text.R")
source("../lib/bigramize.R")
```

Read the ground truth text from files.

```
ground_truth_dir <-"../data/ground_truth"
ground_truth_file_name <- list.files(ground_truth_dir)
ground_truth_file_path <- paste0(ground_truth_dir,'/',ground_truth_file_name)
ground_truth_onelines <- lapply(ground_truth_file_path,get_text)
```

Clean the ground truth texts, convert them to words and sort the words by letter length.

*Clean the text by removing all numbers, punctuations, special symbols. Only letters are retained*

```
ground_truth_words_cleaned <- unlist(lapply(ground_truth_onelines,clean_text))
ground_truth_by_length <- sort_by_length(ground_truth_words_cleaned)
```

Construct letter bigrams for the ground truth words

```
### Construct letter bigrams for the words having length >= 3
num_len <- length(ground_truth_by_length)
ground_truth_bigram_from3 <- lapply(ground_truth_by_length[3:num_len],bigramize)
```

```
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
## new length
```

```
## new length
## new length
## new length
## new length
## new length
## new length
```

```
# save(ground_truth_bigram_from3,file="../output/ground_truth_bigram_from3.Rdata")
# load("../output/ground_truth_bigram_from3.Rdata")

### include words having length <= 2
ground_truth_bigram <- c(list(l_1=list(PD_1=ground_truth_by_length[[1]]),
                              l_2=list(PD_1_2=ground_truth_by_length[[2]])),
                         ground_truth_bigram_from3)
# save(ground_truth_bigram,file="../output/ground_truth_bigram.Rdata")
# load("../output/ground_truth_bigram.Rdata")

### romove duplicate bigrams
ground_truth_unibig <- lapply(ground_truth_bigram,unique_bigram) ### this is the final bigram list for c
save(ground_truth_unibig,file="../output/ground_truth_unibig.Rdata")
# load("../output/ground_truth_unibig.Rdata")
```

## Step 2: construct a word list using tesseract texts

Read the tesseract text from files.

```
tesseract_dir <- "../data/tesseract"
tesseract_file_name <- list.files(tesseract_dir)
tesseract_file_path <- paste0(tesseract_dir,'/',tesseract_file_name)
tesseract_onelines <- lapply(tesseract_file_path,get_text)
```

Clean the tesseract texts, convert them to words.

```
tesseract_words_cleaned <- lapply(tesseract_onelines,clean_text)
save(tesseract_words_cleaned,file="../output/tesseract_words_cleaned.Rdata")
# load("../output/tesseract_words_cleaned.Rdata")
```

## Step 3: error detection

Call all the required customized functions from the library.

```
source("../lib/make_label.R")
```

Construct a label list that indicates whether a word in the given string is an error. `label = 1` means the corresponding word is an error and `label = 0` means the word is not an error

```
tesseract_labels <- make_label(tesseract_words_cleaned,ground_truth_unibig)
```

```
## current file number = 1
## current file number = 2
## current file number = 3
## current file number = 4
## current file number = 5
## current file number = 6
## current file number = 7
```

```
## current file number = 8
## current file number = 9
## current file number = 10
## current file number = 11
## current file number = 12
## current file number = 13
## current file number = 14
## current file number = 15
## current file number = 16
## current file number = 17
## current file number = 18
## current file number = 19
## current file number = 20
## current file number = 21
## current file number = 22
## current file number = 23
## current file number = 24
## current file number = 25
## current file number = 26
## current file number = 27
## current file number = 28
## current file number = 29
## current file number = 30
## current file number = 31
## current file number = 32
## current file number = 33
## current file number = 34
## current file number = 35
## current file number = 36
## current file number = 37
## current file number = 38
## current file number = 39
## current file number = 40
## current file number = 41
## current file number = 42
## current file number = 43
## current file number = 44
## current file number = 45
## current file number = 46
## current file number = 47
## current file number = 48
## current file number = 49
## current file number = 50
## current file number = 51
## current file number = 52
## current file number = 53
## current file number = 54
## current file number = 55
## current file number = 56
## current file number = 57
## current file number = 58
## current file number = 59
## current file number = 60
## current file number = 61
```

```
## current file number = 62
## current file number = 63
## current file number = 64
## current file number = 65
## current file number = 66
## current file number = 67
## current file number = 68
## current file number = 69
## current file number = 70
## current file number = 71
## current file number = 72
## current file number = 73
## current file number = 74
## current file number = 75
## current file number = 76
## current file number = 77
## current file number = 78
## current file number = 79
## current file number = 80
## current file number = 81
## current file number = 82
## current file number = 83
## current file number = 84
## current file number = 85
## current file number = 86
## current file number = 87
## current file number = 88
## current file number = 89
## current file number = 90
## current file number = 91
## current file number = 92
## current file number = 93
## current file number = 94
## current file number = 95
## current file number = 96
## current file number = 97
## current file number = 98
## current file number = 99
## current file number = 100
```

```
### optional - change names of label lists
names(tesseract_labels) <- ground_truth_file_name
for (i in 1:length(tesseract_labels)) {
  names(tesseract_labels[[i]]) <- unlist(strsplit(tesseract_words_cleaned[[i]]," "))
}
save(tesseract_labels,file="../output/tesseract_labels.Rdata")
# load("../output/tesseract_labels.Rdata")
```

Preliminary result analysis

```
error_rate <- (sapply(tesseract_labels, sum)/sapply(tesseract_words_cleaned, length))
summary(error_rate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1249  0.1697  0.1810  0.1794  0.1910  0.2422
```