Task 2. Draw Wordclouds for AGAC and Genia

Github link: https://github.com/Allen-ZKW/NLP_HZAU/tree/Task2

Author: Kewei Zhao
Date: 2021-4-2

Abstract

In this task, we try to clean the text files from these two corpuses (delete punctuations, numbers, stop-words etc.) and draw wordclouds to reflect frequency of words in text file.

Principle

The size of these words will reflect their frequency in text file.

Measure

1. Clean Data and Calculate Frequency (R x64 4.0.3)

1) Import Essential Packages

```
library (NLP)
library (tm)
library (wordcloud2)
```

2) Setup Working Directory, File Path and Make Corpus

```
setwd("D:\\junior_n\\NLP\\task_2")
cname <- file.path("D:","junior_n","NLP","task_2","AGAC")
docs <- Corpus(DirSource(cname))
```

```
setwd("D:\\junior_n\\NLP\\task_2")
cname <- file.path("D:","junior_n","NLP","task_2","GENIA")
docs <- Corpus(DirSource(cname))
```

3) Remove Punctuation and Other Signs

```
tospace <- content_transformer(function(x,parttern)gsub(parttern," ",x))
docs <- tm_map(docs,removePunctuation)
docs <- tm_map(docs,tospace,'/')
docs <- tm_map(docs,tospace,'@')
docs <- tm_map(docs,tospace,'\\|')
```

4) Remove All Numbers

```
docs <- tm_map(docs,removeNumbers)
```

5) Remove Stop-Words in English

```
docs <- tm_map(docs,removeWords,stopwords("english"))
```

6) Convert the Documents to Plain Text Documents (Removal of Tags)

```
docs <- tm_map(docs,PlainTextDocument)
```

7) Stemming

```
docs <- tm_map(docs, stemDocument)
```

8) Create Frequency Vector

```
dtm <- DocumentTermMatrix(docs)
freq <- colSums(as.matrix(dtm))
```

9) Sort the Vector and Extract 500 words in the Head of Vector

```
ord <- order(freq,decreasing = T)
test = freq[head(ord,500)]
```

10) Change Type of Data and Save as CSV file

```
d <- data.frame(word = names(test),freq = test)
```

```
write.table(d,"genia_freq.csv",row.names=FALSE,col.names=FALSE,sep=",")
```

```
write.table(d,"AGAC_freq.csv",row.names=FALSE,col.names=FALSE,sep=",")
```

2. Get Intersection and Draw Wordclouds (Python 3.8 (64-bit))

1) Import Essential Modules

```
from wordcloud import WordCloud
import imageio
import csv
import jieba
```

2) Define Function to Batch delete

```
def batchdel(tar_list,del_list):
    for i in sorted(del_list,reverse=True):
        del tar_list[i]
    return tar_list
```

3) Define Function to Import Data

```
def importdata(filename,dirpath):
    word_freq = []
    word_list = []
    csv_reader = csv.reader(open(dirpath+filename))
    for row in csv_reader:
        word_freq.append((row[0],int(row[1])))
        word_list.append(row[0])
    return word_freq,word_list
```

4) Define Function to Get Intersection of Two Frequency List

```
def cross(AGAC_freq,AGAC_list,GENIA_freq,GENIA_list):
    cross_freq = []
    AGAC_del = []
    GENIA_del = []
    cross_list = list(set(AGAC_list)&set(GENIA_list))
    for word in cross_list:
        freq = AGAC_freq[AGAC_list.index(word)][1] +
GENIA_freq[GENIA_list.index(word)][1]
        cross_freq.append((word,freq))
        AGAC_del.append(AGAC_list.index(word))
        GENIA_del.append(GENIA_list.index(word))
    AGAC_freq = batchdel(AGAC_freq,AGAC_del)
    GENIA_freq = batchdel(GENIA_freq,GENIA_del)
    return AGAC_freq,GENIA_freq,cross_freq
```

5) Define Function to Draw Wordcloud in Specific Shape

```
def drawcloud(word_freq,pngname,dirpath):
    text=''
    for i in word_freq:
        text = text + (i[0]+' ')*i[1]
    cut = jieba.cut(text)
    string = ' '.join(cut)
    color_mask =imageio.imread(dirpath+pngname)
    wc = WordCloud(
                background_color="white",
                collocations=False,
                mask=color_mask)
    wc.generate(string)
    wc.to_file('wc'+pngname)
```

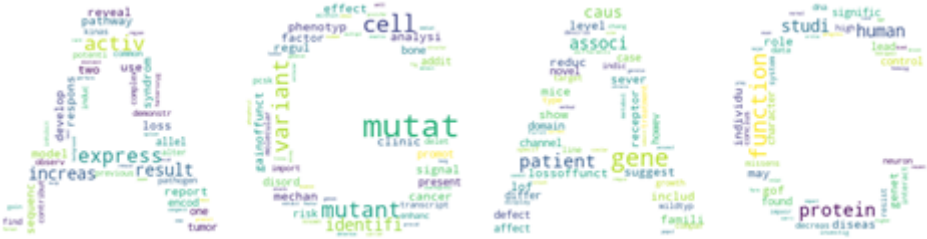
6) Setup Path and Call Functions

```
def main():
    dirpath = "D:/junior_n/NLP/task_2/"
    AGAC_freq,AGAC_list = importdata("AGAC_freq.csv",dirpath)
    GENIA_freq,GENIA_list = importdata("genia_freq.csv",dirpath)
    drawcloud(GENIA_freq,"GENIA.png",dirpath)
    drawcloud(AGAC_freq,"AGAC.png",dirpath)
    AGAC_freq,GENIA_freq,cross_freq =
cross(AGAC_freq,AGAC_list,GENIA_freq,GENIA_list)
    drawcloud(GENIA_freq,"GENI.png",dirpath)
    drawcloud(AGAC_freq,"GAC.png",dirpath)
    drawcloud(cross_freq,"A.png",dirpath)
main()
```

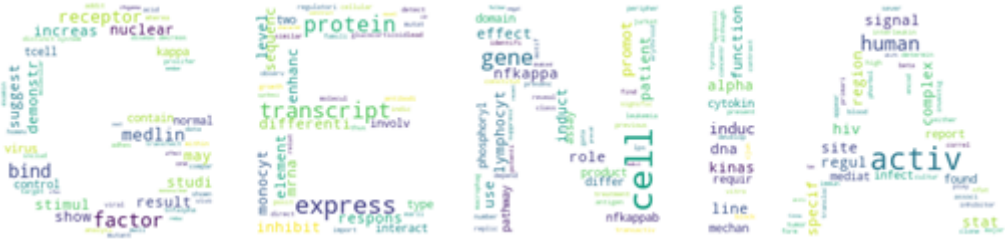
3. Make up Wordclouds to One (Adobe Photoshop CC 2019)

Result

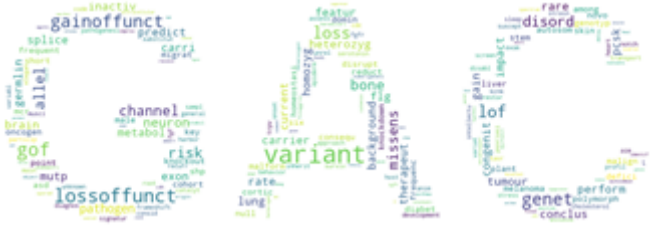
Wordcloud of Words in AGAC corpus



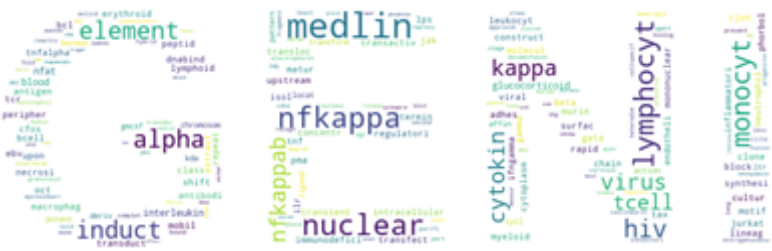
Wordcloud of Words in GENIA corpus



Wordcloud of Unique Words in AGAC Corpus (Not in GENIA Corpus)



Wordcloud of Unique Words in GENIA Corpus (Not in AGAC Corpus)



Wordcloud of Intersection of Two Corpus

Union Wordcloud

Discussion

We try to make some wordclouds in special shapes, so we import 'wordcloud 2' not 'wordcloud' in R. However, the function named ‘lettercloud’ in 'wordcloud 2' cannot function properly even if we use demo-freq supplied by Author. But picture of strings as the intermediate products of this function can output normally. For example:

Unfortunately, function to draw wordcloud by the shape of png file also cannot complete its mission.

So, having no choice, we have to use python to complete the surplus steps. We change frequency list to string and draw wordcloud by using module named wordcloud.

In order to reflect the similarity and difference between two corpuses, we also calculate the intersection of 500 high-frequency words in each corpus. After that, we draw 277 words in both corpuses in the shape of “A”, 223 words only in AGAC corpus in the shape of “GAC”, 223 words only in GENIA corpus in the shape of “GENI”. In the end, we use photoshop to combine these wordclouds, and we believe this union wordcloud can achieve our goals reflecting similarity and difference between two corpuses to some degree.