TidyText - Tokenization & N-grams

#install.packages("Rtools")  
#install.packages("tidytext")  
#install.packages("janeautenr")  
#install.packages("stringr")

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(tidytext)  
library(janeaustenr)  
library(stringr)

text<-c("John likes to watch movies. Mary likes too.","John also likes to watch football game.")  
text

## [1] "John likes to watch movies. Mary likes too."  
## [2] "John also likes to watch football game."

class(text)

## [1] "character"

#### Contruct text in a structred format - dataframe/tibble ####  
  
text\_df<-data\_frame(text=text) ## column name is text

## Warning: `data\_frame()` was deprecated in tibble 1.1.0.  
## ℹ Please use `tibble()` instead.

text\_df

## # A tibble: 2 × 1  
## text   
## <chr>   
## 1 John likes to watch movies. Mary likes too.  
## 2 John also likes to watch football game.

text\_tb<-tibble(text1=text) ## column name is text1  
text\_tb

## # A tibble: 2 × 1  
## text1   
## <chr>   
## 1 John likes to watch movies. Mary likes too.  
## 2 John also likes to watch football game.

str(text\_tb)

## tibble [2 × 1] (S3: tbl\_df/tbl/data.frame)  
## $ text1: chr [1:2] "John likes to watch movies. Mary likes too." "John also likes to watch football game."

text\_df1<-data\_frame(line =1:2, text1=text) # adding the column 'line'  
  
text\_df1$line

## [1] 1 2

text\_df1$text1

## [1] "John likes to watch movies. Mary likes too."  
## [2] "John also likes to watch football game."

text\_tb$text1 ## select a specific column from the data frame

## [1] "John likes to watch movies. Mary likes too."  
## [2] "John also likes to watch football game."

#### Tokenization ####  
  
unnest\_tokens(text\_df1, word, text1) # tokenize the text in 'text1' column into word

## # A tibble: 15 × 2  
## line word   
## <int> <chr>   
## 1 1 john   
## 2 1 likes   
## 3 1 to   
## 4 1 watch   
## 5 1 movies   
## 6 1 mary   
## 7 1 likes   
## 8 1 too   
## 9 2 john   
## 10 2 also   
## 11 2 likes   
## 12 2 to   
## 13 2 watch   
## 14 2 football  
## 15 2 game

text\_df1 %>% unnest\_tokens(word, text1)

## # A tibble: 15 × 2  
## line word   
## <int> <chr>   
## 1 1 john   
## 2 1 likes   
## 3 1 to   
## 4 1 watch   
## 5 1 movies   
## 6 1 mary   
## 7 1 likes   
## 8 1 too   
## 9 2 john   
## 10 2 also   
## 11 2 likes   
## 12 2 to   
## 13 2 watch   
## 14 2 football  
## 15 2 game

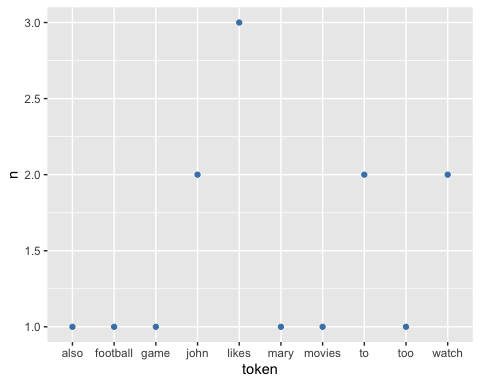
tokens<-unnest\_tokens(text\_df1, token, text1) #create a dataframe with tokens and assign to tokens  
tokens

## # A tibble: 15 × 2  
## line token   
## <int> <chr>   
## 1 1 john   
## 2 1 likes   
## 3 1 to   
## 4 1 watch   
## 5 1 movies   
## 6 1 mary   
## 7 1 likes   
## 8 1 too   
## 9 2 john   
## 10 2 also   
## 11 2 likes   
## 12 2 to   
## 13 2 watch   
## 14 2 football  
## 15 2 game

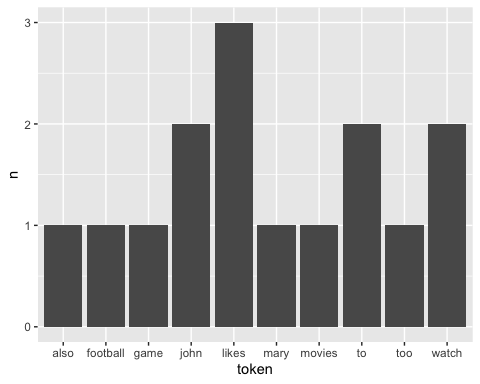
count(tokens, token, sort=TRUE) # count the term frequency

## # A tibble: 10 × 2  
## token n  
## <chr> <int>  
## 1 likes 3  
## 2 john 2  
## 3 to 2  
## 4 watch 2  
## 5 also 1  
## 6 football 1  
## 7 game 1  
## 8 mary 1  
## 9 movies 1  
## 10 too 1

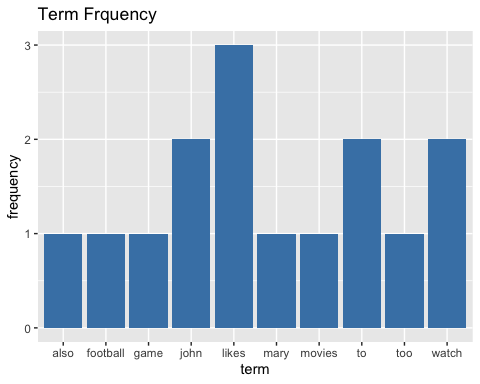
library(ggplot2)  
count(tokens, token, sort=TRUE) %>% ggplot(aes(x=token, y=n)) + geom\_point(col="steelblue")



count(tokens, token, sort=TRUE)%>% ggplot(aes(x=token,y=n)) + geom\_col()



count(tokens, token, sort=TRUE)%>% ggplot(aes(x=token,y=n)) + geom\_col(fill="steelblue") + labs(title="Term Frquency", x="term", y="frequency")



#### N gram #####  
  
?austen\_books  
austen\_books() %>% group\_by(book) # Display text by book

## # A tibble: 73,422 × 2  
## # Groups: book [6]  
## text book   
## <chr> <fct>   
## 1 "SENSE AND SENSIBILITY" Sense & Sensibility  
## 2 "" Sense & Sensibility  
## 3 "by Jane Austen" Sense & Sensibility  
## 4 "" Sense & Sensibility  
## 5 "(1811)" Sense & Sensibility  
## 6 "" Sense & Sensibility  
## 7 "" Sense & Sensibility  
## 8 "" Sense & Sensibility  
## 9 "" Sense & Sensibility  
## 10 "CHAPTER 1" Sense & Sensibility  
## # … with 73,412 more rows

books<-austen\_books() %>%   
group\_by(book) %>%   
mutate(line\_no= row\_number(), chapter=cumsum(str\_detect(text, regex("^chapter [\\d]", ignore\_case=TRUE)))) %>%   
ungroup()  
books

## # A tibble: 73,422 × 4  
## text book line\_no chapter  
## <chr> <fct> <int> <int>  
## 1 "SENSE AND SENSIBILITY" Sense & Sensibility 1 0  
## 2 "" Sense & Sensibility 2 0  
## 3 "by Jane Austen" Sense & Sensibility 3 0  
## 4 "" Sense & Sensibility 4 0  
## 5 "(1811)" Sense & Sensibility 5 0  
## 6 "" Sense & Sensibility 6 0  
## 7 "" Sense & Sensibility 7 0  
## 8 "" Sense & Sensibility 8 0  
## 9 "" Sense & Sensibility 9 0  
## 10 "CHAPTER 1" Sense & Sensibility 10 1  
## # … with 73,412 more rows

# regular expression from the textbook.  
books <-austen\_books() %>%   
group\_by(book) %>%   
mutate(line\_no= row\_number(), chapter=cumsum(str\_detect(text, regex("^chapter [\\divxlc]", ignore\_case=TRUE)))) %>%   
ungroup()

#### Tokenization on books ####  
tidy\_books<-books %>% unnest\_tokens(word, text)  
tidy\_books

## # A tibble: 725,055 × 4  
## book line\_no chapter word   
## <fct> <int> <int> <chr>   
## 1 Sense & Sensibility 1 0 sense   
## 2 Sense & Sensibility 1 0 and   
## 3 Sense & Sensibility 1 0 sensibility  
## 4 Sense & Sensibility 3 0 by   
## 5 Sense & Sensibility 3 0 jane   
## 6 Sense & Sensibility 3 0 austen   
## 7 Sense & Sensibility 5 0 1811   
## 8 Sense & Sensibility 10 1 chapter   
## 9 Sense & Sensibility 10 1 1   
## 10 Sense & Sensibility 13 1 the   
## # … with 725,045 more rows

data("stop\_words") ## load the stop words ##  
head(stop\_words)

## # A tibble: 6 × 2  
## word lexicon  
## <chr> <chr>   
## 1 a SMART   
## 2 a's SMART   
## 3 able SMART   
## 4 about SMART   
## 5 above SMART   
## 6 according SMART

tidy\_books<-tidy\_books %>% anti\_join(stop\_words, by="word")  
tidy\_books

## # A tibble: 217,609 × 4  
## book line\_no chapter word   
## <fct> <int> <int> <chr>   
## 1 Sense & Sensibility 1 0 sense   
## 2 Sense & Sensibility 1 0 sensibility  
## 3 Sense & Sensibility 3 0 jane   
## 4 Sense & Sensibility 3 0 austen   
## 5 Sense & Sensibility 5 0 1811   
## 6 Sense & Sensibility 10 1 chapter   
## 7 Sense & Sensibility 10 1 1   
## 8 Sense & Sensibility 13 1 family   
## 9 Sense & Sensibility 13 1 dashwood   
## 10 Sense & Sensibility 13 1 settled   
## # … with 217,599 more rows

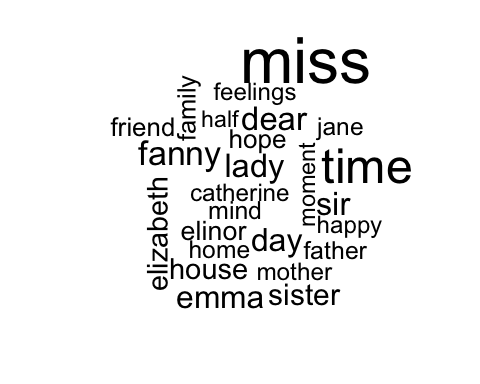
dim(tidy\_books)

## [1] 217609 4

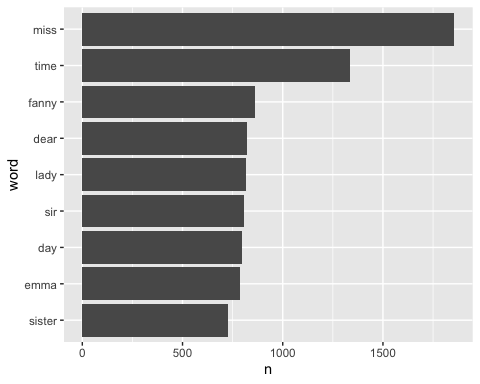
tidy\_books %>% count(word, sort=TRUE) ->word\_fr ## word and number of frequence of tidy\_book   
 ## are assigned to word\_fr  
  
  
  
#install.packages("wordcloud")  
library("wordcloud")

## Loading required package: RColorBrewer

# create a wordcloud  
wordcloud(word\_fr$word, word\_fr$n, min.freq = 500)



library(ggplot2)  
tidy\_books %>% count(word, sort=TRUE) %>%   
 filter(n>700) %>%   
 mutate(word=reorder(word, n)) %>%   
 ggplot(aes(word, n))+  
 geom\_col()+  
 coord\_flip()



library(tidyr)  
bigram <- books %>% unnest\_tokens(bigram, text, token = "ngrams", n=2) %>%   
 filter(!is.na(bigram))  
bigram

## # A tibble: 662,783 × 4  
## book line\_no chapter bigram   
## <fct> <int> <int> <chr>   
## 1 Sense & Sensibility 1 0 sense and   
## 2 Sense & Sensibility 1 0 and sensibility  
## 3 Sense & Sensibility 3 0 by jane   
## 4 Sense & Sensibility 3 0 jane austen   
## 5 Sense & Sensibility 10 1 chapter 1   
## 6 Sense & Sensibility 13 1 the family   
## 7 Sense & Sensibility 13 1 family of   
## 8 Sense & Sensibility 13 1 of dashwood   
## 9 Sense & Sensibility 13 1 dashwood had   
## 10 Sense & Sensibility 13 1 had long   
## # … with 662,773 more rows

bigram2 <-austen\_books() %>% unnest\_tokens(bigram, text, token="ngrams", n=2) %>%   
 drop\_na() ## drop rows with missing values

bigram %>% count(bigram, sort=TRUE)

## # A tibble: 193,209 × 2  
## bigram n  
## <chr> <int>  
## 1 of the 2853  
## 2 to be 2670  
## 3 in the 2221  
## 4 it was 1691  
## 5 i am 1485  
## 6 she had 1405  
## 7 of her 1363  
## 8 to the 1315  
## 9 she was 1309  
## 10 had been 1206  
## # … with 193,199 more rows

library(tidyr)  
  
## split bigram into two columns 'term1', 'term2'  
bigram\_sep <-bigram %>%   
 separate(bigram, c("term1", "term2"), sep=" ")   
bigram\_sep

## # A tibble: 662,783 × 5  
## book line\_no chapter term1 term2   
## <fct> <int> <int> <chr> <chr>   
## 1 Sense & Sensibility 1 0 sense and   
## 2 Sense & Sensibility 1 0 and sensibility  
## 3 Sense & Sensibility 3 0 by jane   
## 4 Sense & Sensibility 3 0 jane austen   
## 5 Sense & Sensibility 10 1 chapter 1   
## 6 Sense & Sensibility 13 1 the family   
## 7 Sense & Sensibility 13 1 family of   
## 8 Sense & Sensibility 13 1 of dashwood   
## 9 Sense & Sensibility 13 1 dashwood had   
## 10 Sense & Sensibility 13 1 had long   
## # … with 662,773 more rows

# Clean the terms with stop words  
bigram\_term\_cleaned <-bigram\_sep %>%   
 filter(!term1 %in% stop\_words$word) %>%   
 filter(!term2 %in% stop\_words$word)  
bigram\_term\_cleaned

## # A tibble: 38,913 × 5  
## book line\_no chapter term1 term2   
## <fct> <int> <int> <chr> <chr>   
## 1 Sense & Sensibility 3 0 jane austen   
## 2 Sense & Sensibility 10 1 chapter 1   
## 3 Sense & Sensibility 14 1 norland park   
## 4 Sense & Sensibility 17 1 surrounding acquaintance  
## 5 Sense & Sensibility 17 1 late owner   
## 6 Sense & Sensibility 18 1 advanced age   
## 7 Sense & Sensibility 19 1 constant companion   
## 8 Sense & Sensibility 20 1 happened ten   
## 9 Sense & Sensibility 22 1 henry dashwood   
## 10 Sense & Sensibility 23 1 norland estate   
## # … with 38,903 more rows

# Count the frequency of terms after cleaning the terms  
bigram\_count <- bigram\_term\_cleaned %>%   
 count(term1, term2, sort=TRUE)  
bigram\_count

## # A tibble: 28,974 × 3  
## term1 term2 n  
## <chr> <chr> <int>  
## 1 sir thomas 266  
## 2 miss crawford 196  
## 3 captain wentworth 143  
## 4 miss woodhouse 143  
## 5 frank churchill 114  
## 6 lady russell 110  
## 7 sir walter 108  
## 8 lady bertram 101  
## 9 miss fairfax 98  
## 10 colonel brandon 96  
## # … with 28,964 more rows

# Merge two columns into one column 'bigram'  
bigram\_united <-bigram\_term\_cleaned %>%   
 unite(bigram, term1, term2, sep=" ")  
bigram\_united

## # A tibble: 38,913 × 4  
## book line\_no chapter bigram   
## <fct> <int> <int> <chr>   
## 1 Sense & Sensibility 3 0 jane austen   
## 2 Sense & Sensibility 10 1 chapter 1   
## 3 Sense & Sensibility 14 1 norland park   
## 4 Sense & Sensibility 17 1 surrounding acquaintance  
## 5 Sense & Sensibility 17 1 late owner   
## 6 Sense & Sensibility 18 1 advanced age   
## 7 Sense & Sensibility 19 1 constant companion   
## 8 Sense & Sensibility 20 1 happened ten   
## 9 Sense & Sensibility 22 1 henry dashwood   
## 10 Sense & Sensibility 23 1 norland estate   
## # … with 38,903 more rows

# Search term 'love' and count the frequency of the related term (consecutive terms)  
bigram\_term\_cleaned %>%   
 filter(term2=="love") %>%   
 count(book, term1, sort=TRUE)

## # A tibble: 47 × 3  
## book term1 n  
## <fct> <chr> <int>  
## 1 Sense & Sensibility edward's 2  
## 2 Mansfield Park dearly 2  
## 3 Sense & Sensibility accepted 1  
## 4 Sense & Sensibility another's 1  
## 5 Sense & Sensibility ardent 1  
## 6 Sense & Sensibility common 1  
## 7 Sense & Sensibility disappointed 1  
## 8 Sense & Sensibility disastrous 1  
## 9 Sense & Sensibility equally 1  
## 10 Sense & Sensibility mother's 1  
## # … with 37 more rows

#### Construct a semantic network  
#install.packages("igraph")  
library(igraph)

##   
## Attaching package: 'igraph'

## The following object is masked from 'package:tidyr':  
##   
## crossing

## The following objects are masked from 'package:dplyr':  
##   
## as\_data\_frame, groups, union

## The following objects are masked from 'package:stats':  
##   
## decompose, spectrum

## The following object is masked from 'package:base':  
##   
## union

bigram\_count

## # A tibble: 28,974 × 3  
## term1 term2 n  
## <chr> <chr> <int>  
## 1 sir thomas 266  
## 2 miss crawford 196  
## 3 captain wentworth 143  
## 4 miss woodhouse 143  
## 5 frank churchill 114  
## 6 lady russell 110  
## 7 sir walter 108  
## 8 lady bertram 101  
## 9 miss fairfax 98  
## 10 colonel brandon 96  
## # … with 28,964 more rows

class(bigram\_count)

## [1] "tbl\_df" "tbl" "data.frame"

g<-filter(bigram\_count, n>30) %>% graph\_from\_data\_frame() # Convert into the igraph object  
g

## IGRAPH 75ac8a2 DN-- 50 42 --   
## + attr: name (v/c), n (e/n)  
## + edges from 75ac8a2 (vertex names):  
## [1] sir ->thomas miss ->crawford captain ->wentworth   
## [4] miss ->woodhouse frank ->churchill lady ->russell   
## [7] sir ->walter lady ->bertram miss ->fairfax   
## [10] colonel ->brandon sir ->john miss ->bates   
## [13] jane ->fairfax lady ->catherine lady ->middleton   
## [16] miss ->tilney miss ->bingley thousand->pounds   
## [19] miss ->dashwood dear ->miss miss ->bennet   
## [22] miss ->morland captain ->benwick miss ->smith   
## + ... omitted several edges

V(g) # Show vertex

## + 50/50 vertices, named, from 75ac8a2:  
## [1] sir miss captain frank lady colonel   
## [7] jane thousand dear henry john dr   
## [13] ten de mansfield thomas crawford wentworth   
## [19] woodhouse churchill russell walter bertram fairfax   
## [25] brandon bates catherine middleton tilney bingley   
## [31] pounds dashwood bennet morland benwick smith   
## [37] knightley crawford's grant elliot price taylor   
## [43] william thomas's minutes bourgh park darcy   
## [49] emma fanny

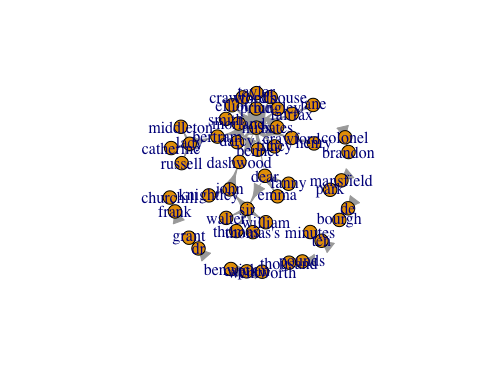
g

## IGRAPH 75ac8a2 DN-- 50 42 --   
## + attr: name (v/c), n (e/n)  
## + edges from 75ac8a2 (vertex names):  
## [1] sir ->thomas miss ->crawford captain ->wentworth   
## [4] miss ->woodhouse frank ->churchill lady ->russell   
## [7] sir ->walter lady ->bertram miss ->fairfax   
## [10] colonel ->brandon sir ->john miss ->bates   
## [13] jane ->fairfax lady ->catherine lady ->middleton   
## [16] miss ->tilney miss ->bingley thousand->pounds   
## [19] miss ->dashwood dear ->miss miss ->bennet   
## [22] miss ->morland captain ->benwick miss ->smith   
## + ... omitted several edges

class(g)

## [1] "igraph"

plot(g) # Display a word network graph



pacman::p\_load(ggraph)  
library(ggraph)  
set.seed(2017)  
ggraph(g, layout="fr") + geom\_edge\_link() + geom\_node\_point() + geom\_node\_text(aes(label=name), vjust=1, hjust=1)

