# Assignment_10

Folorunsho Atanda

2023-11-08

## Chapter 2 Code

```
suppressWarnings({
  library(tidyverse)
  library(tidytext)
  library(janeaustenr)
  library(textdata)
  library(wordcloud)
  library(reshape2)
})
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Loading required package: RColorBrewer
##
##
## Attaching package: 'reshape2'
##
##
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
get_sentiments("afinn")
```

```
## # A tibble: 2,477 x 2
##    word       value
##    <chr>      <dbl>
##  1 abandon       -2
##  2 abandoned     -2
##  3 abandons      -2
```

```
##  4 abducted      -2
##  5 abduction     -2
##  6 abductions    -2
##  7 abhor         -3
##  8 abhorred      -3
##  9 abhorrent     -3
## 10 abhors        -3
## # i 2,467 more rows
```

```
get_sentiments("bing")
```

```
## # A tibble: 6,786 x 2
##    word        sentiment
##    <chr>       <chr>
##  1 2-faces     negative
##  2 abnormal    negative
##  3 abolish     negative
##  4 abominable  negative
##  5 abominably  negative
##  6 abominate   negative
##  7 abomination negative
##  8 abort       negative
##  9 aborted     negative
## 10 aborts      negative
## # i 6,776 more rows
```

```
get_sentiments("nrc")
```

```
## # A tibble: 13,872 x 2
##    word        sentiment
##    <chr>       <chr>
##  1 abacus      trust
##  2 abandon     fear
##  3 abandon     negative
##  4 abandon     sadness
##  5 abandoned   anger
##  6 abandoned   fear
##  7 abandoned   negative
##  8 abandoned   sadness
##  9 abandonment anger
## 10 abandonment fear
## # i 13,862 more rows
```

```
tidy_books <- austen_books() %>%
  group_by(book) %>%
  mutate(
    linenumber = row_number(),
    chapter = cumsum(str_detect(text,
                                regex("^chapter [\\divxlc]",
                                      ignore_case = TRUE)))) %>%
  ungroup() %>%
  unnest_tokens(word, text)
```

```r
nrc_joy <- get_sentiments("nrc") %>%
  filter(sentiment == "joy")

tidy_books %>%
  filter(book == "Emma") %>%
  inner_join(nrc_joy) %>%
  count(word, sort = TRUE)
```

```
## Joining with `by = join_by(word)`
```

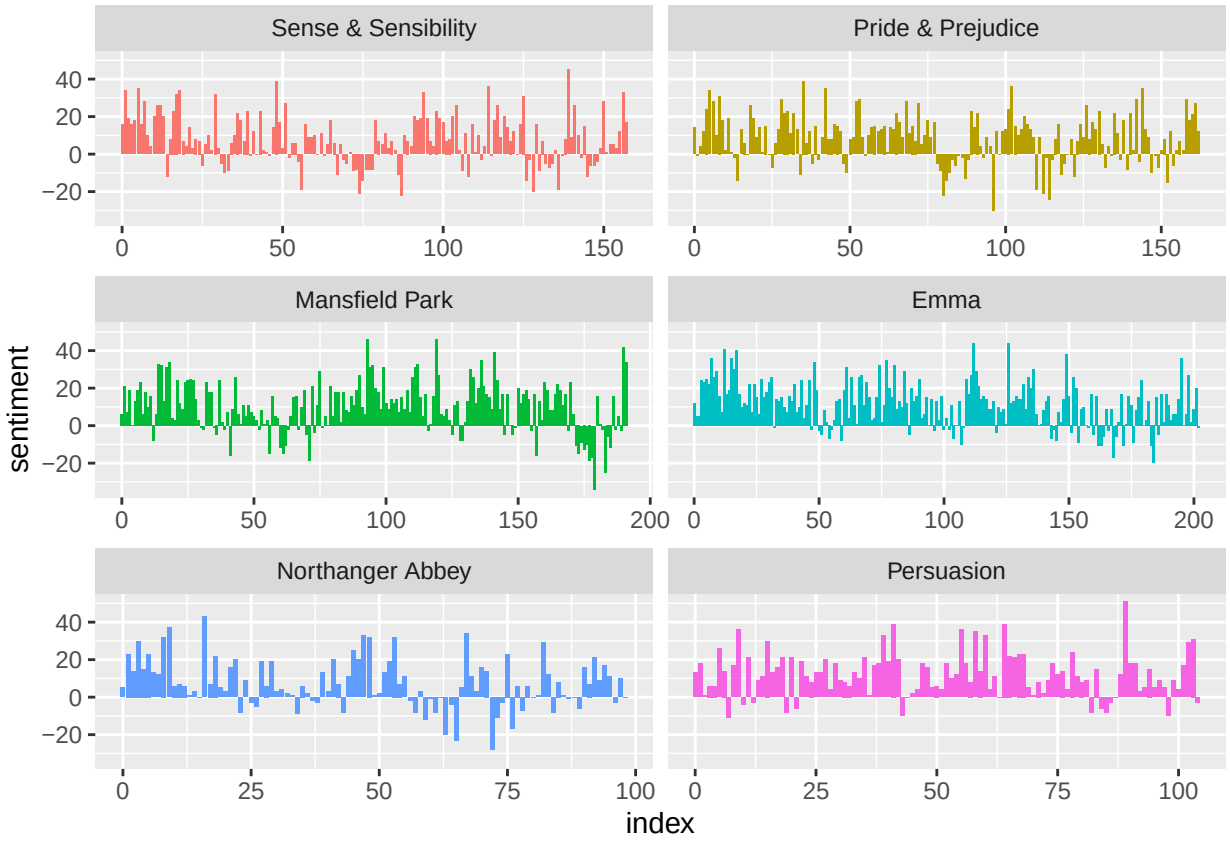
```
## # A tibble: 301 x 2
##    word          n
##    <chr>     <int>
##  1 good        359
##  2 friend      166
##  3 hope        143
##  4 happy       125
##  5 love        117
##  6 deal         92
##  7 found        92
##  8 present      89
##  9 kind         82
## 10 happiness    76
## # i 291 more rows
```

```r
jane_austen_sentiment <- tidy_books %>%
  inner_join(get_sentiments("bing")) %>%
  count(book, index = linenumber%/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment, values_from = n, values_fill = 0) %>%
  mutate(sentiment = positive - negative)
```

```
## Joining with `by = join_by(word)`
```

```
## Warning in inner_join(., get_sentiments("bing")): Detected an unexpected many-to-many relationship be
## i Row 435434 of `x` matches multiple rows in `y`.
## i Row 5051 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

```r
jane_austen_sentiment %>%
  ggplot(aes(index, sentiment, fill = book)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~book, ncol = 2, scales = "free_x")
```




```
pride_prejudice <- tidy_books %>%
  filter(book == "Pride & Prejudice")

pride_prejudice
```

```
## # A tibble: 122,204 x 4
##    book               linenumber chapter word
##    <fct>                   <int>   <int> <chr>
##  1 Pride & Prejudice           1       0 pride
##  2 Pride & Prejudice           1       0 and
##  3 Pride & Prejudice           1       0 prejudice
##  4 Pride & Prejudice           3       0 by
##  5 Pride & Prejudice           3       0 jane
##  6 Pride & Prejudice           3       0 austen
##  7 Pride & Prejudice           7       1 chapter
##  8 Pride & Prejudice           7       1 1
##  9 Pride & Prejudice          10       1 it
## 10 Pride & Prejudice          10       1 is
## # i 122,194 more rows
```

```
afinn <- pride_prejudice %>%
  inner_join(get_sentiments("afinn")) %>%
  group_by(index = linenumber %/% 80) %>%
  summarise(sentiment = sum(value)) %>%
  mutate(method = "AFINN")
```

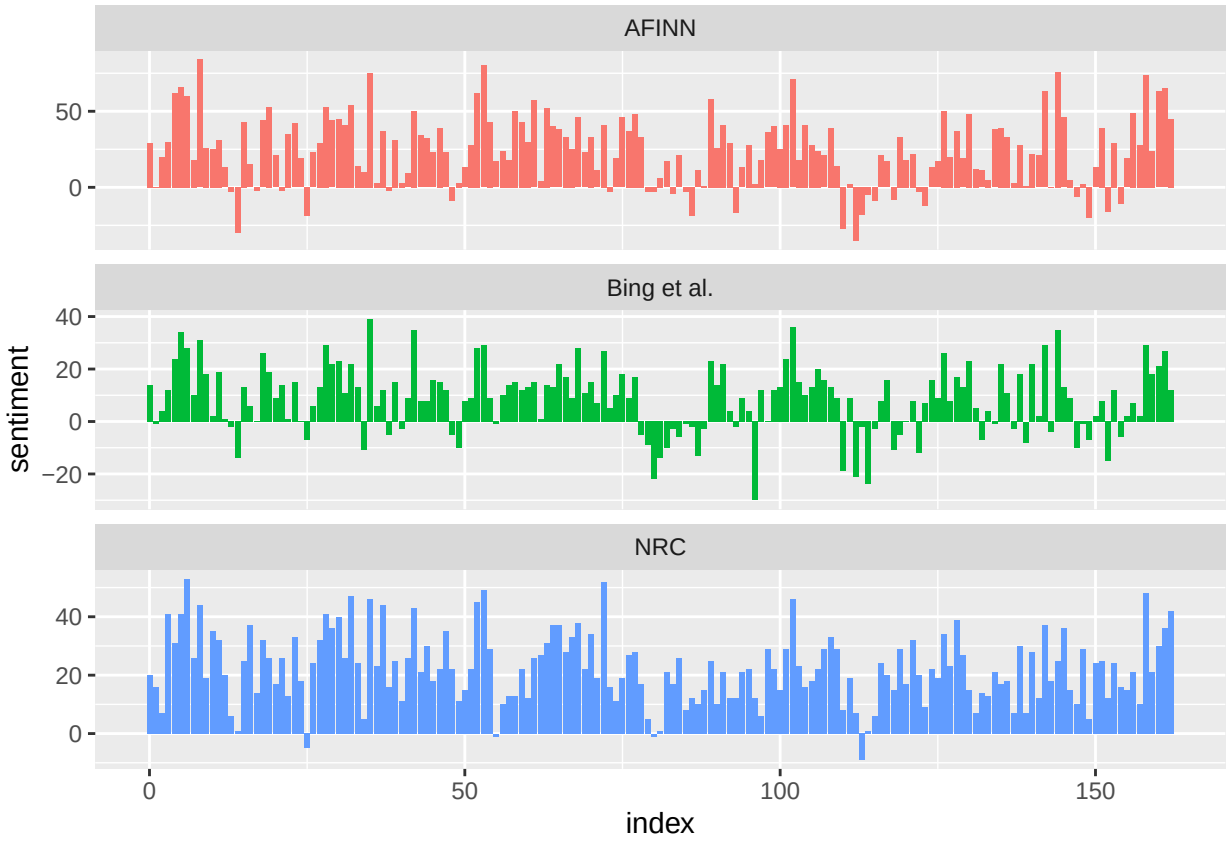
```
## Joining with `by = join_by(word)`
```

```
bing_and_nrc <- bind_rows(
  pride_prejudice %>%
    inner_join(get_sentiments("bing")) %>%
    mutate(method = "Bing et al."),
  pride_prejudice %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive",
                                         "negative"))
    ) %>%
    mutate(method = "NRC")) %>%
  count(method, index = linenumber %/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment,
              values_from = n,
              values_fill = 0) %>%
  mutate(sentiment = positive - negative)
```

```
## Joining with `by = join_by(word)`
## Joining with `by = join_by(word)`
```

```
## Warning in inner_join(., get_sentiments("nrc") %>% filter(sentiment %in% : Detected an unexpected ma
## i Row 215 of `x` matches multiple rows in `y`.
## i Row 5178 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

```
bind_rows(afinn,
          bing_and_nrc) %>%
  ggplot(aes(index, sentiment, fill = method)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~method, ncol = 1, scales = "free_y")
```




```
get_sentiments("nrc") %>%
  filter(sentiment %in% c("positive", "negative")) %>%
  count(sentiment)
```

```
## # A tibble: 2 x 2
##   sentiment     n
##   <chr>     <int>
## 1 negative   3316
## 2 positive   2308
```

```
get_sentiments("bing") %>%
  count(sentiment)
```

```
## # A tibble: 2 x 2
##   sentiment     n
##   <chr>     <int>
## 1 negative   4781
## 2 positive   2005
```

```
bing_word_counts <- tidy_books %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  ungroup()
```

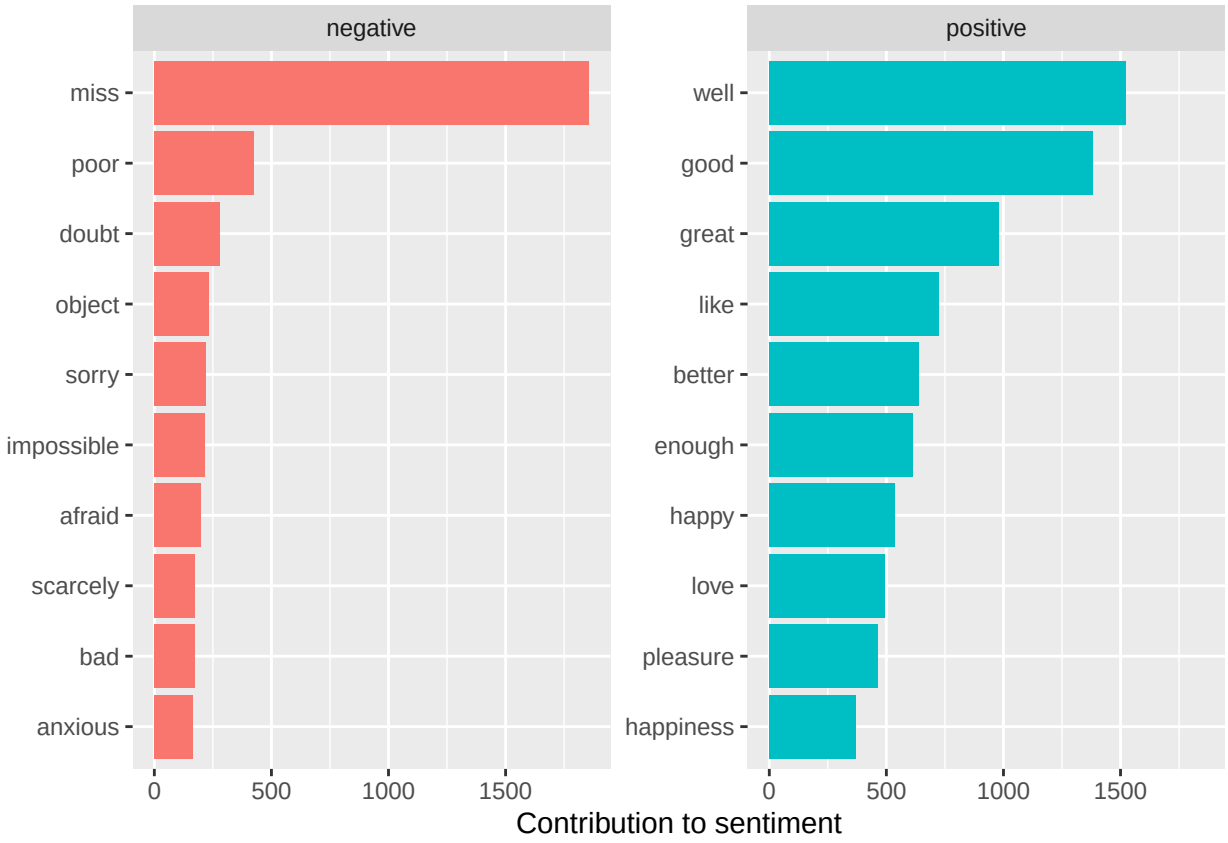
```
## Joining with `by = join_by(word)`
```

```
## Warning in inner_join(., get_sentiments("bing")): Detected an unexpected many-to-many relationship be
## i Row 435434 of `x` matches multiple rows in `y`.
## i Row 5051 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

```
bing_word_counts
```

```
## # A tibble: 2,585 x 3
##    word      sentiment     n
##    <chr>     <chr>     <int>
##  1 miss      negative   1855
##  2 well      positive   1523
##  3 good      positive   1380
##  4 great     positive    981
##  5 like      positive    725
##  6 better    positive    639
##  7 enough    positive    613
##  8 happy     positive    534
##  9 love      positive    495
## 10 pleasure  positive    462
## # i 2,575 more rows
```

```
bing_word_counts %>%
  group_by(sentiment) %>%
  slice_max(n, n = 10) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word, fill = sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y") +
  labs(x = "Contribution to sentiment",
       y = NULL)
```




```
custom_stop_words <- bind_rows(tibble(word = c("miss"),
                                      lexicon = c("custom")),
                               stop_words)

custom_stop_words
```

```
## # A tibble: 1,150 x 2
##    word        lexicon
##    <chr>       <chr>
##  1 miss        custom
##  2 a           SMART
##  3 a's         SMART
##  4 able        SMART
##  5 about       SMART
##  6 above       SMART
##  7 according   SMART
##  8 accordingly SMART
##  9 across      SMART
## 10 actually    SMART
## # i 1,140 more rows
```

```
tidy_books %>%
  anti_join(stop_words) %>%
  count(word) %>%
  with(wordcloud(word, n, max.words = 100))
```

```
## Joining with `by = join_by(word)`
```


```
tidy_books %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("gray20", "gray80"),
                   max.words = 100)
```

```
## Joining with `by = join_by(word)`
```

```
## Warning in inner_join(., get_sentiments("bing")): Detected an unexpected many-to-many relationship be
## i Row 435434 of `x` matches multiple rows in `y`.
## i Row 5051 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```


```
p_and_p_sentences <- tibble(text = prideprejudice) %>%
  unnest_tokens(sentence, text, token = "sentences")

p_and_p_sentences$sentence[2]
```

```
## [1] "by jane austen"
```

```
austen_chapters <- austen_books() %>%
  group_by(book) %>%
  unnest_tokens(chapter, text, token = "regex",
                pattern = "Chapter|CHAPTER [\\dIVXLC]") %>%
  ungroup()

austen_chapters %>%
  group_by(book) %>%
  summarise(chapters = n())
```

```
## # A tibble: 6 x 2
##   book                chapters
##   <fct>                  <int>
## 1 Sense & Sensibility       51
## 2 Pride & Prejudice         62
## 3 Mansfield Park            49
## 4 Emma                      56
```

```
## 5 Northanger Abbey          32
## 6 Persuasion                25
```

```
bingnegative <- get_sentiments("bing") %>%
  filter(sentiment == "negative")

wordcounts <- tidy_books %>%
  group_by(book, chapter) %>%
  summarize(words = n())
```

```
## `summarise()` has grouped output by 'book'. You can override using the
## `.groups` argument.
```

```
tidy_books %>%
  semi_join(bingnegative) %>%
  group_by(book, chapter) %>%
  summarize(negativewords = n()) %>%
  left_join(wordcounts, by = c("book", "chapter")) %>%
  mutate(ratio = negativewords/words) %>%
  filter(chapter != 0) %>%
  slice_max(ratio, n = 1) %>%
  ungroup()
```

```
## Joining with `by = join_by(word)`
## `summarise()` has grouped output by 'book'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 6 x 5
##   book                chapter negativewords words  ratio
##   <fct>                 <int>         <int> <int>  <dbl>
## 1 Sense & Sensibility      43           161  3405 0.0473
## 2 Pride & Prejudice        34           111  2104 0.0528
## 3 Mansfield Park           46           173  3685 0.0469
## 4 Emma                     15           151  3340 0.0452
## 5 Northanger Abbey         21           149  2982 0.0500
## 6 Persuasion                4            62  1807 0.0343
```

# Excercise

Performing a sentiment analysis on Pride and Prejudice

```
p_and_p <- tidy_books %>%
  filter(book == "Pride & Prejudice")

p_and_p
```

```
## # A tibble: 122,204 x 4
##    book              linenumber chapter word
##    <fct>                  <int>   <int> <chr>
##  1 Pride & Prejudice          1       0 pride
```

```
##  2 Pride & Prejudice          1       0 and
##  3 Pride & Prejudice          1       0 prejudice
##  4 Pride & Prejudice          3       0 by
##  5 Pride & Prejudice          3       0 jane
##  6 Pride & Prejudice          3       0 austen
##  7 Pride & Prejudice          7       1 chapter
##  8 Pride & Prejudice          7       1 1
##  9 Pride & Prejudice         10       1 it
## 10 Pride & Prejudice         10       1 is
## # i 122,194 more rows
```

```
# using a nrc sentiment
nrc_sentiment <- get_sentiments("bing")
```

```
p_and_p_sentiment <- p_and_p %>%
  anti_join(custom_stop_words) %>%
  inner_join(nrc_sentiment) %>%
  count(word, sentiment, sort = TRUE)
```

```
## Joining with `by = join_by(word)`
## Joining with `by = join_by(word)`
```

```
p_and_p_sentiment
```

```
## # A tibble: 1,403 x 3
##    word      sentiment     n
##    <chr>     <chr>     <int>
##  1 love      positive     92
##  2 pleasure  positive     92
##  3 happy     positive     83
##  4 happiness positive     72
##  5 affection positive     58
##  6 regard    positive     49
##  7 object    negative     48
##  8 pride     positive     48
##  9 perfectly positive     47
## 10 agreeable positive     45
## # i 1,393 more rows
```

```
p_and_p_sentiment %>%
  with(wordcloud(word, n, max.words = 100))
```

```
## Warning in wordcloud(word, n, max.words = 100): astonishment could not be fit
## on page. It will not be plotted.
```

```
## Warning in wordcloud(word, n, max.words = 100): perfectly could not be fit on
## page. It will not be plotted.
```

```
## Warning in wordcloud(word, n, max.words = 100): compliment could not be fit on
## page. It will not be plotted.
```

```
## Warning in wordcloud(word, n, max.words = 100): object could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): assure could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): doubt could not be fit on page.
## It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): regard could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): extraordinary could not be fit
## on page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): exceedingly could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): resentment could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): handsome could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): delight could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): disappointment could not be fit
## on page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): anxious could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): delightful could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): civility could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): beauty could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): fine could not be fit on page.
## It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): charming could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): silent could not be fit on
## page. It will not be plotted.
```

```
## Warning in wordcloud(word, n, max.words = 100): advantage could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): temper could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): happiness could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): pleasant could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): fortune could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): ignorant could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): politeness could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): comfort could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): respect could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): happy could not be fit on page.
## It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): vanity could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): amiable could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): perfect could not be fit on
## page. It will not be plotted.

## Warning in wordcloud(word, n, max.words = 100): admire could not be fit on
## page. It will not be plotted.
```


proper
pleasure
poor
capable
smile
fear
affection
love
success
affectionate
fancy
dislike
readily
intelligence
kindness
favour
superior
trouble
ready
led
pleasing
absence
lost
promised
gratitude
humour
cold
spite
pretty
delighted
regret
vain
distress
strong
danger
glad
anxiety
admiration
concerned
angry
pain
pleased
ease
scarcely
pride
loss
fair
fond
praise
afraid
beg
confess
easy
felicity
concern
indifference
excellent
mistaken
agreeable
satisfied
impossible
instantly
master
pardon
promise
proud