textmining

Shuting

12/5/2021

Import Book

```r
Metamorphosis <- gutenberg_download(gutenberg_id = 5200)
```

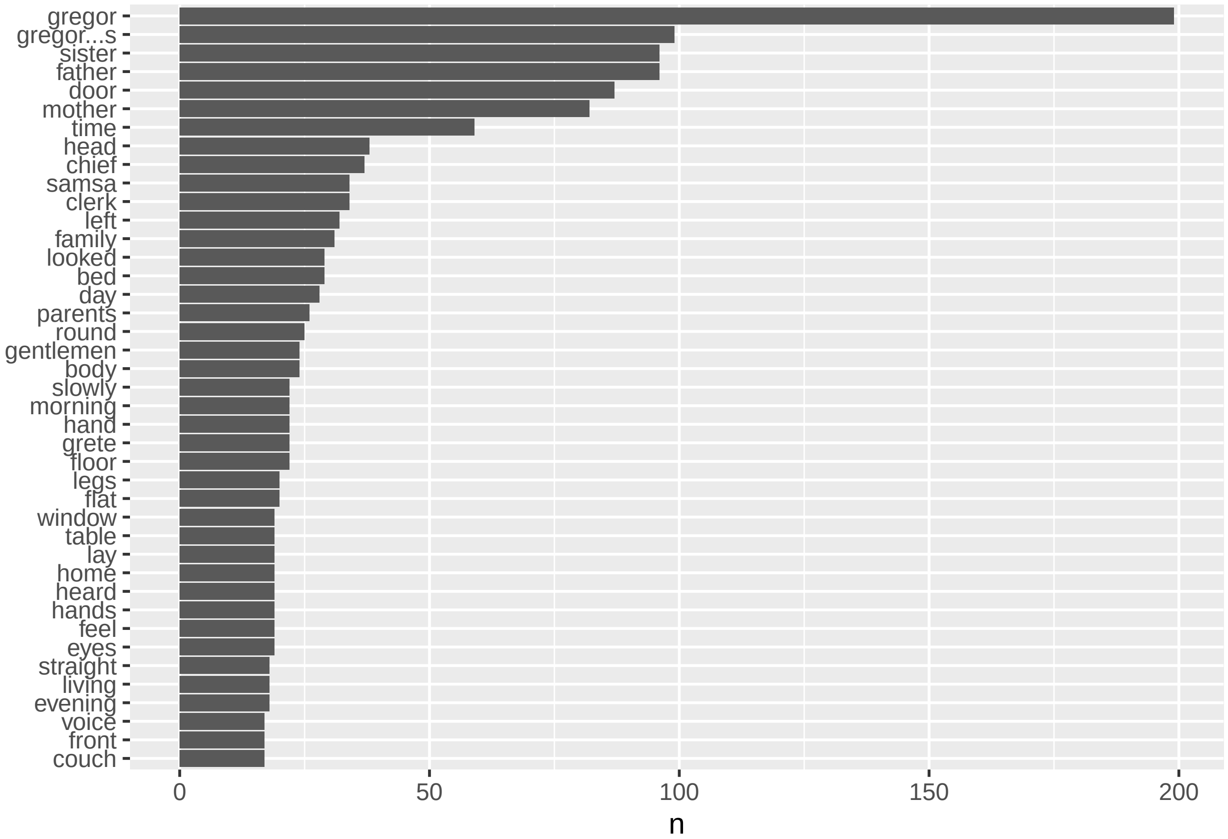
```
## Determining mirror for Project Gutenberg from http://www.gutenberg.org/robot/harvest
```

```
## Using mirror http://aleph.gutenberg.org
```

```r
# write.table(Metamorphosis, "Metamorphosis.txt") # add <> for every chapter in text file
Metamorphosis <- read.table("Metamorphosis.txt", header = T)
```

Tidy Metamorphosis by single-word tokenization

```r
Metamorphosis <- Metamorphosis %>%
  mutate(linenumber = row_number(),
         chapter = cumsum(str_detect(text,
                                     regex("^chapter [\\divxlc]",
                                           ignore_case = TRUE)))) %>%
  ungroup()

tidy_Metamorphosis <- Metamorphosis %>%
  unnest_tokens(word, text)
#########################################
data("stop_words")
tidy_Metamorphosis <- tidy_Metamorphosis %>% anti_join(stop_words)
#########################################
commonwords <- tidy_Metamorphosis %>% count(word, sort = TRUE)

tidy_Metamorphosis %>%
  count(word, sort = TRUE) %>%
  filter(n > 16) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word)) +
  geom_col() +
  labs(y = NULL)
```



Firstly, I convert the book "Metamorphosis" into the tidy text format: a table with one-token-per-row. In this table, for one row, we have one specific word, chapter number and line number of this word.

Secondly, I removed the stop words, words have no meaning, from the tidy table.

Then, I explored the common words of this book by count single word's frequency. And I found the most common words were noun, like "gregor", "father", "sister", "door" and etc. Here, I visualized the words with frequency more than 16.

From this plot, we can easily see the most common words and their frequencies in Metamorphosis.

Sentiment analysis with Metamorphosis

```
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
## # ... with 2,467 more rows
```

```r
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
## # ... with 6,776 more rows
```

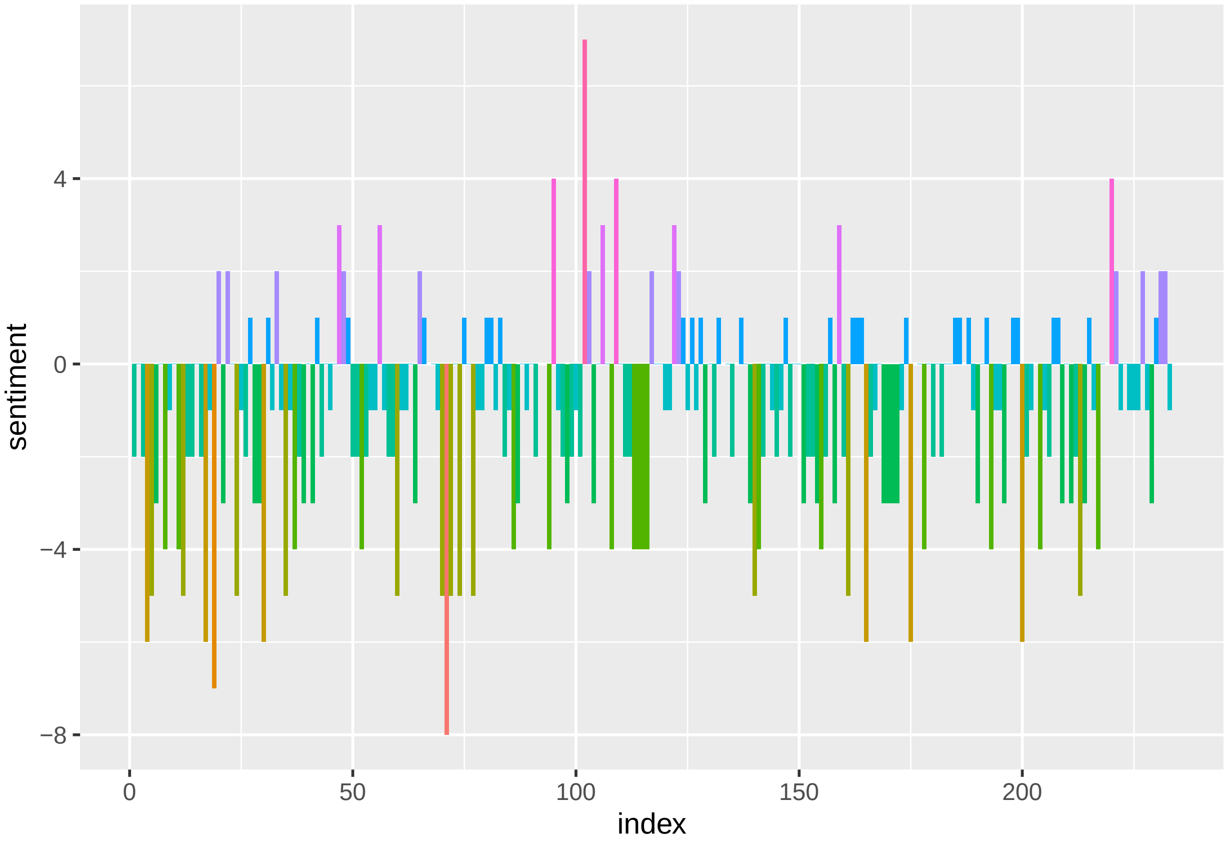
```r
get_sentiments("nrc")
```

```
## # A tibble: 13,875 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
## # ... with 13,865 more rows
```

From the tidytext package, we can find three sentiment lexicons based on single words:

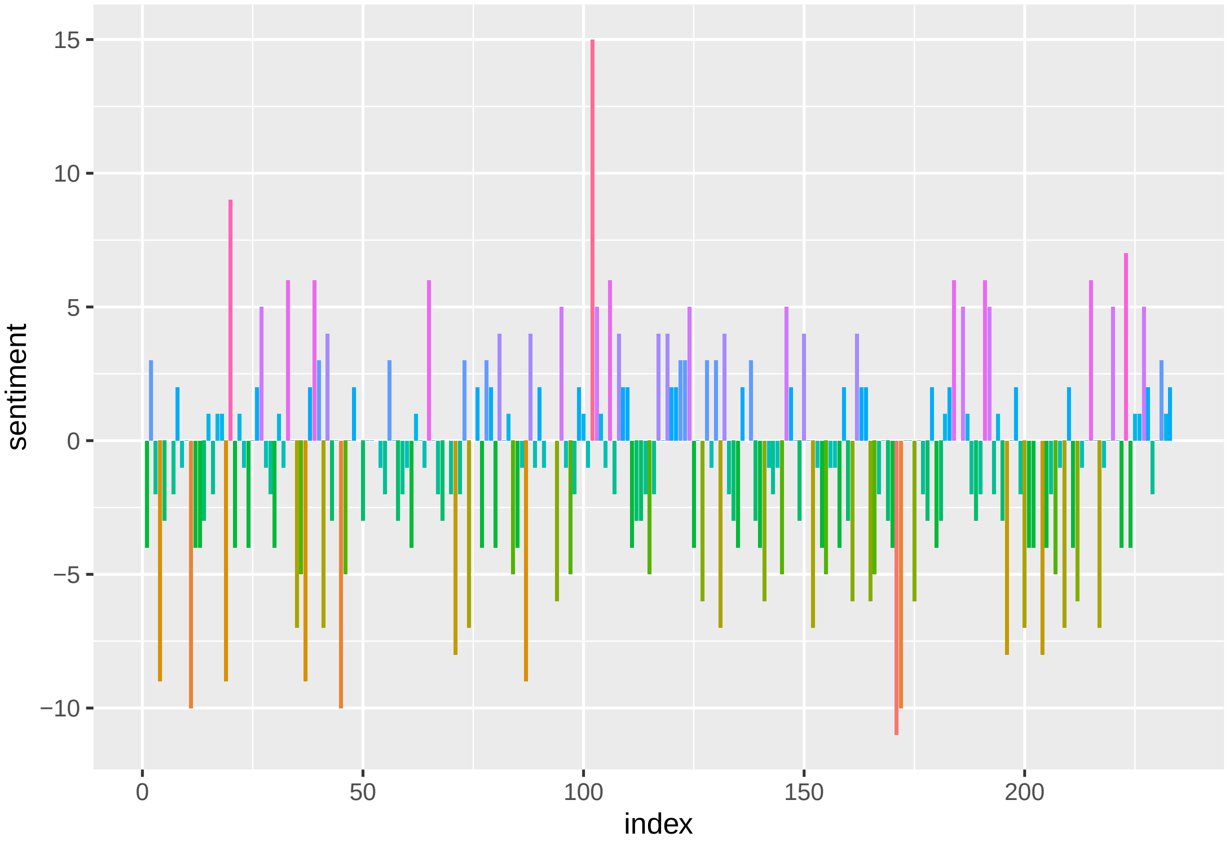
- afinn: assign words with a score form -5 to 5
- bing: categorize words in a binary fashion, positive and negative.
- nrc : assign words into emotion categories of positive, negative, anger, anticipation, disgust, fear, joy, sadness, surprise, and trust.

```r
### analysis based on "bing" ###
Metamorphosis_sentiment_bing <- tidy_Metamorphosis %>%
  inner_join(get_sentiments("bing")) %>%
  count(index = linenumber %/% 8, sentiment) %>%
  pivot_wider(names_from = sentiment, values_from = n, values_fill = 0) %>%
  mutate(sentiment = positive - negative)

ggplot(Metamorphosis_sentiment_bing, aes(index, sentiment, fill=factor(sentiment))) +
  geom_col(show.legend = FALSE)
```



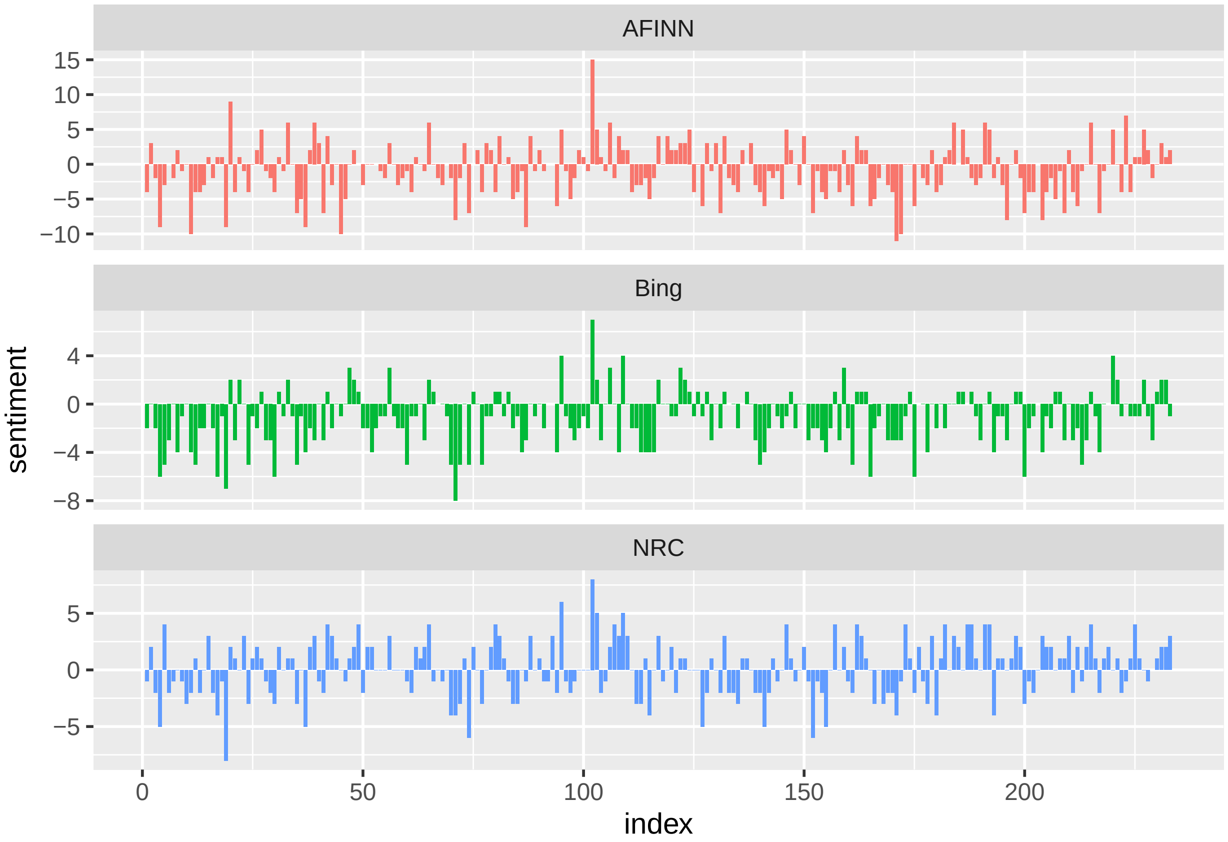
- Firstly, I did sentiment analysis with Metamorphosis based on “bing”, the plot shows each line’s positive or negative sentiment, and because x-axis is novel’s line number index, we can also track the sentiment trend through narrative time.
- We can see from this plot, Metamorphosis shows more negative sentiment, at the end of this novel, it shows positive ending.

```
### analysis based on "afinn" ###
Metamorphosis_sentiment_afinn <- tidy_Metamorphosis %>%
  inner_join(get_sentiments("afinn")) %>%
  group_by(index = linenumber %/% 8) %>%
  summarise(sentiment = sum(value)) %>%
  mutate(method = "AFINN")

ggplot(Metamorphosis_sentiment_afinn, aes(index, sentiment,fill=factor(sentiment))) +
  geom_col(show.legend = FALSE)
```



- Secondly, I did sentiment analysis with Metamorphosis based on "afinn", the plot shows similar result as "bing" method.

```
### compare three sentiment dictionaries ###
Metamorphosis_sentiment_bing_and_nrc <- bind_rows(
  tidy_Metamorphosis %>%
    inner_join(get_sentiments("bing")) %>%
    mutate(method = "Bing"),
 tidy_Metamorphosis %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive",
                                         "negative"))
    ) %>%
    mutate(method = "NRC")) %>%
  count(method, index = linenumber %/% 8, sentiment) %>%
  pivot_wider(names_from = sentiment,
              values_from = n,
              values_fill = 0) %>%
  mutate(sentiment = positive - negative)

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

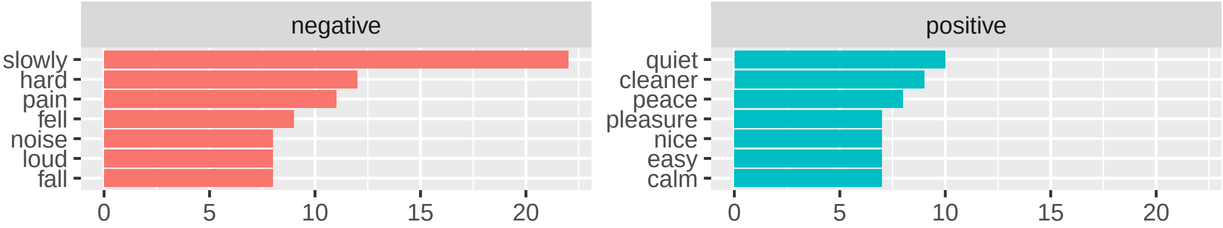


- Finally, I compared sentiment analysis results on three different sentiment lexicons. We can see the three lexicons shows similar sentiment trajectories through the novel, but the absolutely values are different. This is because different lexicons have different sentiment words, so they show different accuracy when analyse one specific book.

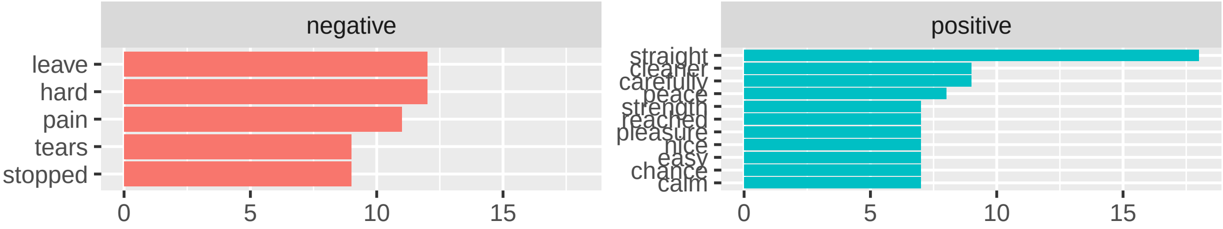
Check most common positive and negative words in Metamorphosis

```
tidy_Metamorphosis_less <- select(tidy_Metamorphosis,4)

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

afinn_word_counts <- tidy_Metamorphosis_less %>%
  inner_join(get_sentiments("afinn")) %>%
  mutate(sentiment=ifelse(value>=0,"positive","negative")) %>%
  count(word, sentiment, sort = TRUE) %>%
  ungroup()

nrc_word_counts <- tidy_Metamorphosis_less %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive",
                                         "negative"))
    ) %>%
 count(word, sentiment, sort = TRUE) %>%
  ungroup()
```

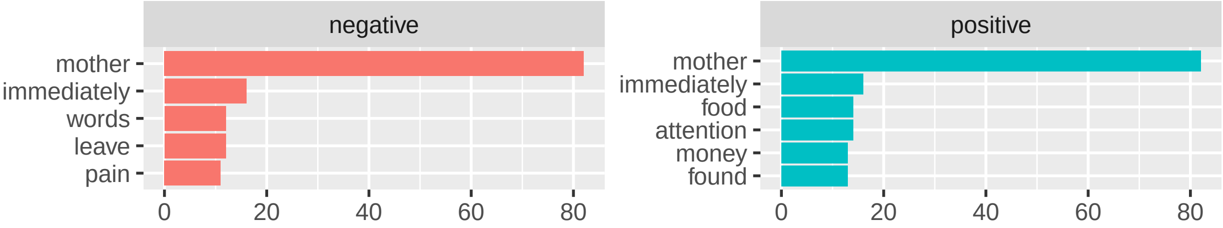
```r
# word_counts_Metamorphosis <- bind_rows(
#   (tidy_Metamorphosis_less) %>%
#   inner_join(get_sentiments("bing")) %>%
#   count(word, sentiment, sort = TRUE) %>%
#   ungroup() %>% mutate(method="Bing"),
#   tidy_Metamorphosis_less %>%
#   inner_join(get_sentiments("afinn")) %>%
#   mutate(sentiment=ifelse(value>=0,"positive","negative")) %>%
#   count(word, sentiment, sort = TRUE) %>%
#   ungroup() %>% mutate(method = "AFINN"),
#   tidy_Metamorphosis_less %>%
#   inner_join(get_sentiments("nrc") %>% filter(sentiment %in% c("positive","negative"))) %>%
#   count(word, sentiment, sort = TRUE) %>% ungroup() %>% mutate(method="NRC"))


bing <- bing_word_counts %>%
  group_by(sentiment) %>%
  slice_max(n, n = 5) %>%
  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 = "Bing",
       y = NULL)
afinn <- afinn_word_counts %>%
  group_by(sentiment) %>%
  slice_max(n, n = 5) %>%
  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 = "AFINN",
       y = NULL)
nrc <- nrc_word_counts %>%
  group_by(sentiment) %>%
  slice_max(n, n = 5) %>%
  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 = "NRC",
       y = NULL)

grid.arrange(bing,afinn,nrc, ncol=1)
```



Bing



AFINN



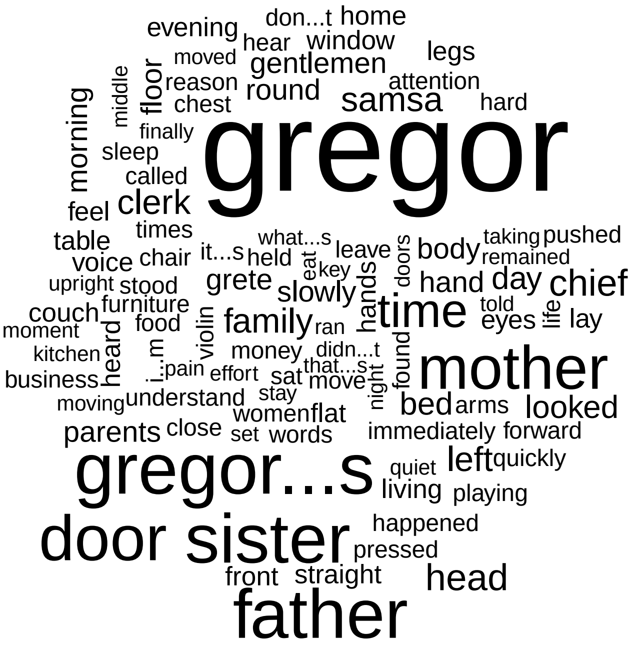
NRC

Here I compared the most common positive and negative words detected by three sentiment lexicons. We can see NRC assign "mother" as both positive and negative, it's not suitable for Metamorphosis.

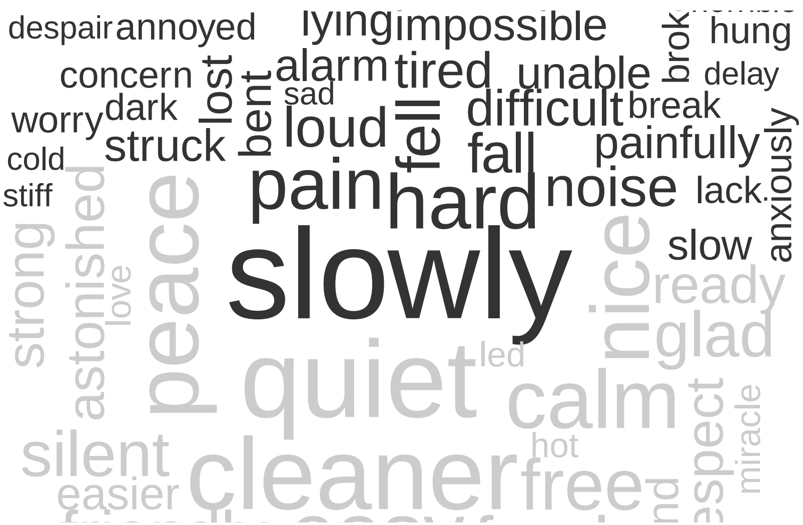
For other two lexicons, we can see Bing detects more positive words, and Afinn detects more positive words.

Wordclouds

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



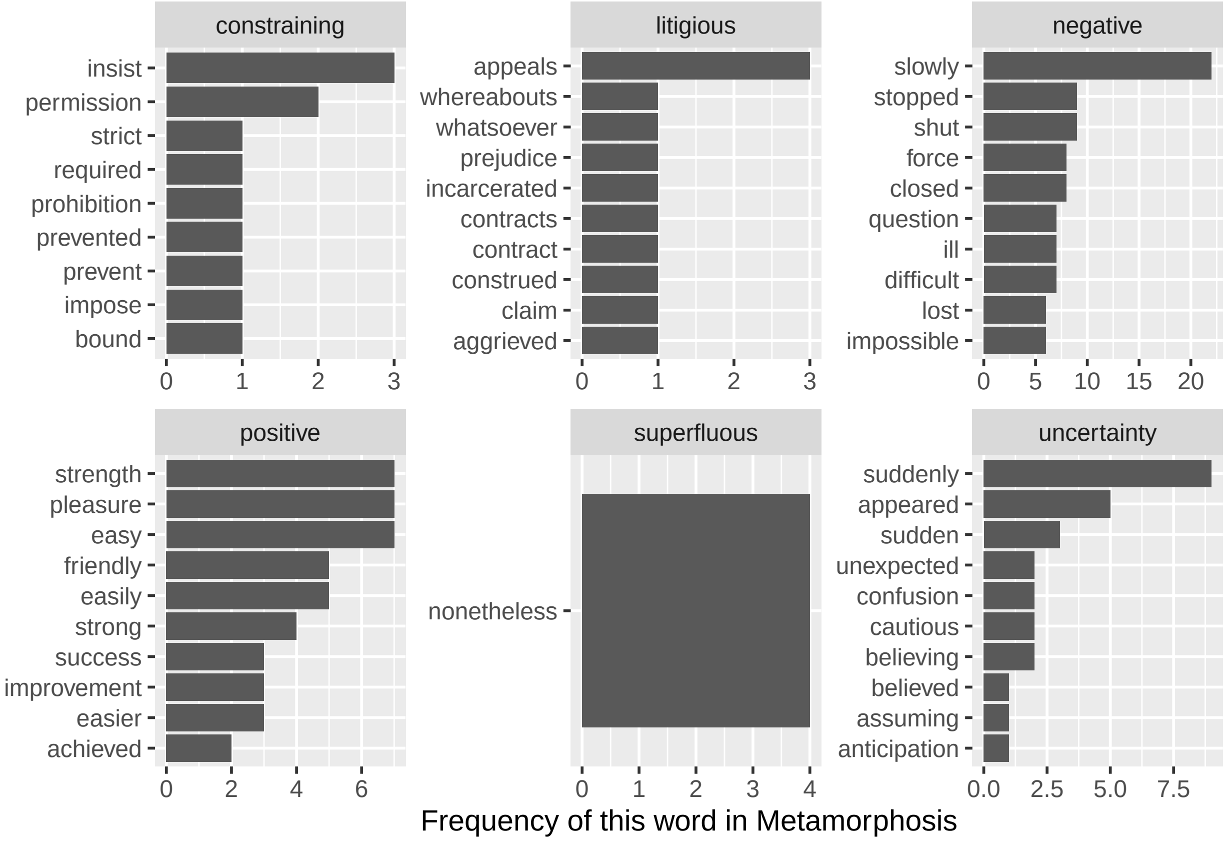
```
tidy_Metamorphosis %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("gray20", "gray80"),
                   max.words = 100)
```



Extra sentiment lexicons "loughran"

Here we try to analyse Metamorphosis based on the sentiment dictinary "loughran", the Loughran divides words into six sentiments: "positive", "negative", "litigious", "uncertain", "constraining", and "superfluous".

```
tidy_Metamorphosis %>%
  count(word) %>%
  inner_join(get_sentiments("loughran"), by = "word") %>%
  group_by(sentiment) %>%
  slice_max(n, n = 10, with_ties = FALSE) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word)) +
  geom_col() +
  facet_wrap(~ sentiment, scales = "free") +
  labs(x = "Frequency of this word in Metamorphosis", y = NULL)
```



From this plot, we can see the most common words for all six sentimental categories in dictionary "loughran". In Metamorphosis, most words belong to positive and uncertainty.