Assignment4

Keliang Xu

11/30/2021

Task One

I choose The Hound of the Baskervilles by Arthur Conan Doyle. https://www.gutenberg.org/ebooks/2852

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

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

## Joining, by = "word"

## # A tibble: 5,159 x 2
##    word            n
##    <chr>       <int>
##  1 sir           351
##  2 holmes        187
##  3 moor          165
##  4 henry         147
##  5 watson        116
##  6 baskerville   113
##  7 dr            110
##  8 mortimer       87
##  9 time           86
## 10 stapleton      85
## # ... with 5,149 more rows
```

Task Two

All the functions and approach to sentiment analysis detailed in Text Mining with R. https://www.tidytextmining.com/sentiment.html

Sentiment analysis with inner join

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

tidy_Holmes %>%
  inner_join(nrc_joy) %>%
  count(word, sort = TRUE)
```

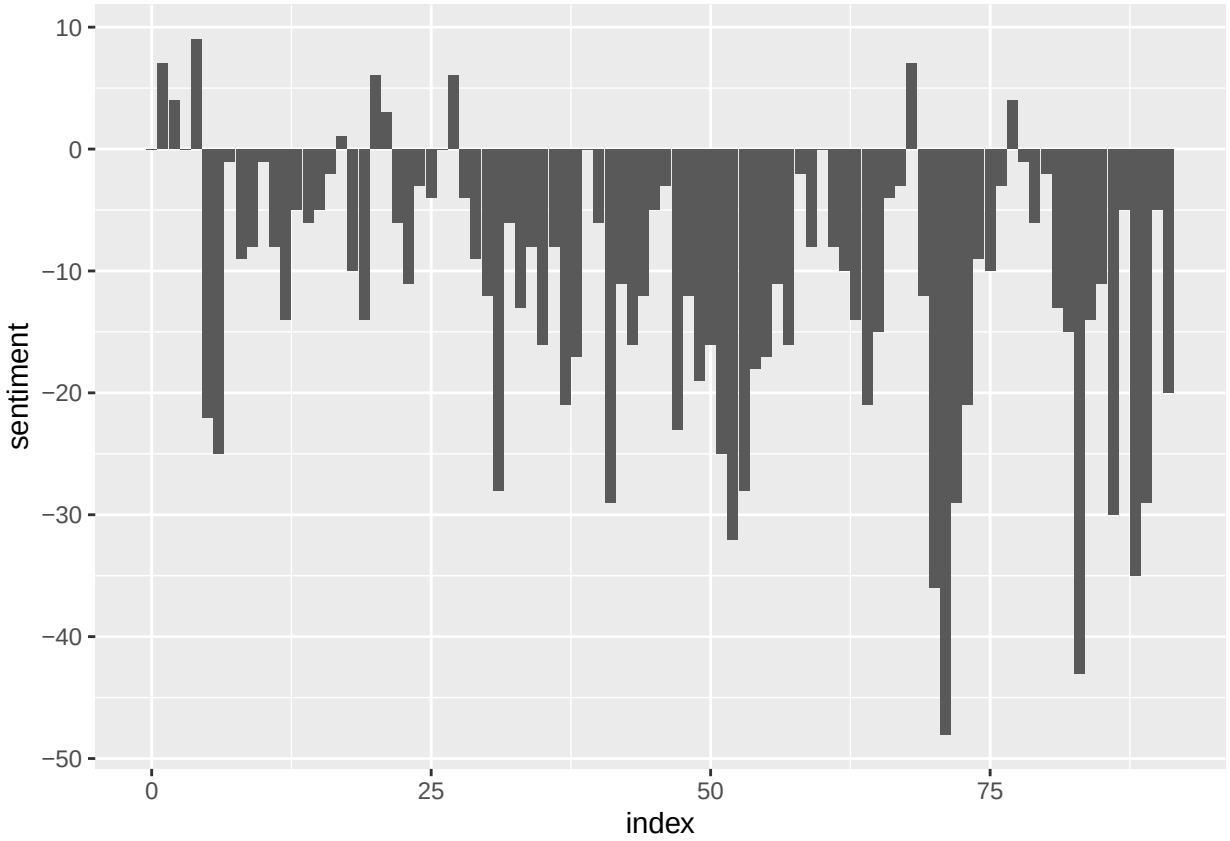
```
## Joining, by = "word"
```

```
## # A tibble: 190 x 2
##    word         n
##    <chr>    <int>
##  1 friend      58
##  2 found       44
##  3 glad        18
##  4 love        16
##  5 surprise    16
##  6 green       14
##  7 save        14
##  8 god         13
##  9 true        13
## 10 hope        12
## # ... with 180 more rows
```

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

```
## Joining, by = "word"
```

```
ggplot(Holmes_sentiment, aes(index, sentiment)) +
  geom_col(show.legend = FALSE)
```



2. The outputs show that the net sentiment (positive - negative) has a very high frequency.

This book mainly tells about Holmes's adventures in investigating the case. The atmosphere in the book will be created according to the case. This case is shrouded in negative elements such as curse, ignorance and death, so the sentiment dictionaries in the book are mostly negative.

Comparing the three sentiment dictionaries

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

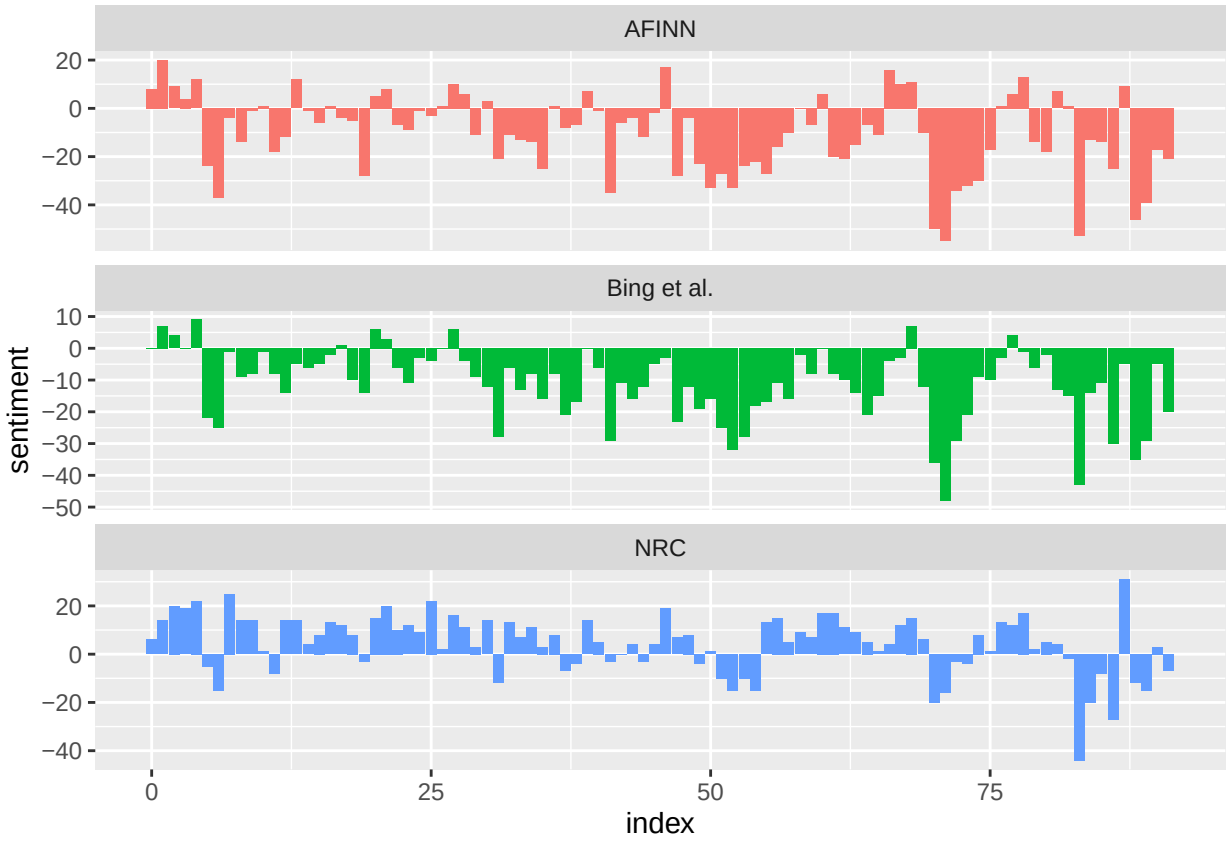
```
## Joining, by = "word"
```

```
bing_and_nrc <- bind_rows(
  tidy_Holmes %>%
    inner_join(get_sentiments("bing")) %>%
    mutate(method = "Bing et al."),
  tidy_Holmes %>%
    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, by = "word"
## Joining, by = "word"
```

```
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")
```



The three different lexicons for calculating sentiment give results that are different in an absolute sense but have similar relative trajectories through the novel. The first two plots are the same as the plot above. Actually I don't know why the third one is different from other.

Most common positive and negative words

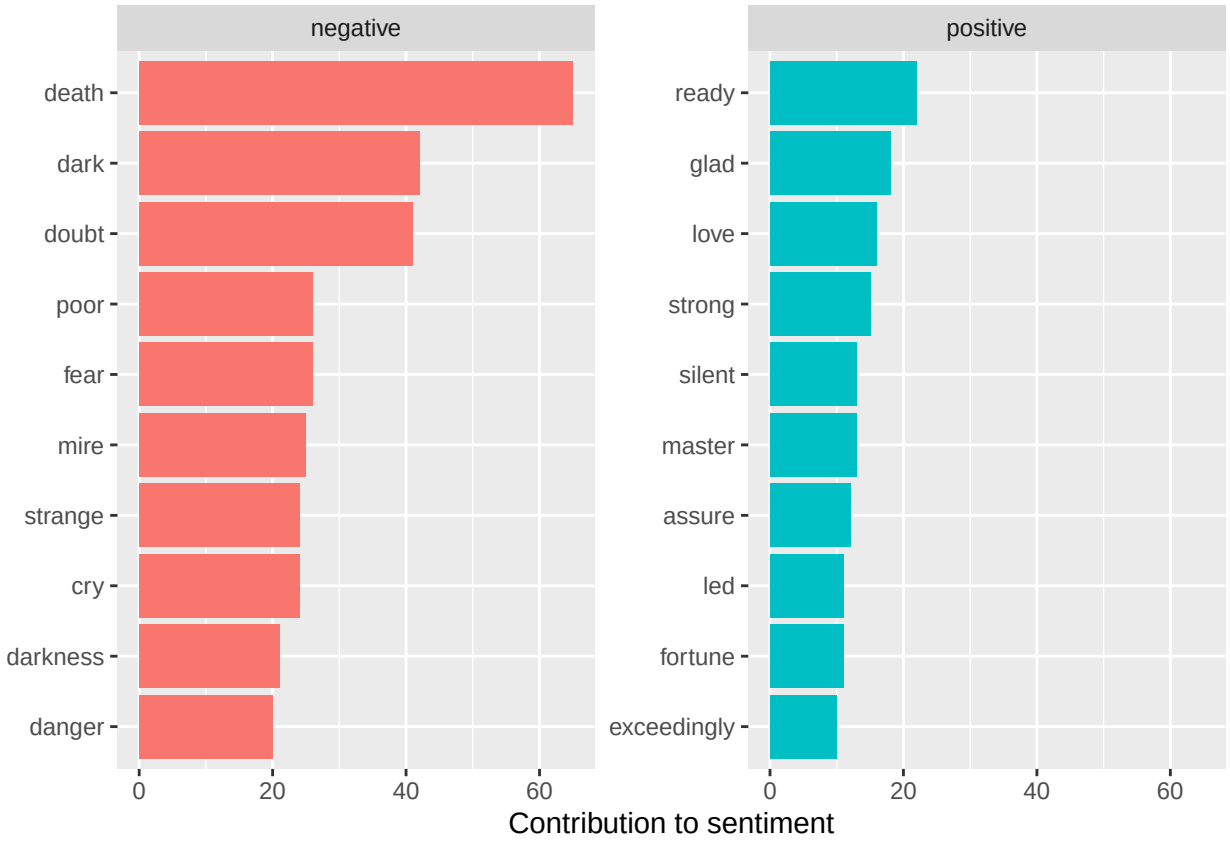
```
bing_word_counts <- tidy_Holmes %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  ungroup()
```

```
## Joining, by = "word"
```

```
bing_word_counts
```

```
## # A tibble: 1,041 x 3
##    word      sentiment     n
##    <chr>     <chr>     <int>
##  1 death     negative     65
##  2 dark      negative     42
##  3 doubt     negative     41
##  4 fear      negative     26
##  5 poor      negative     26
##  6 mire      negative     25
##  7 cry       negative     24
##  8 strange   negative     24
##  9 ready     positive     22
## 10 darkness negative     21
## # ... with 1,031 more rows
```

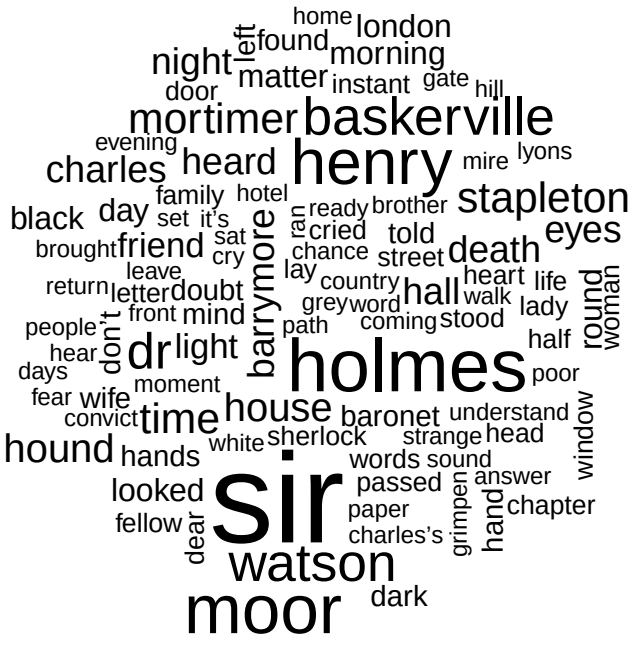
```
bing_word_counts %>%
  group_by(sentiment) %>%
  slice_max(n, n = 10) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word, fill = sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y") +
  labs(x = "Contribution to sentiment",
     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



Wordclouds

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

```
## Joining, by = "word"
```




```
#positive and negative
tidy_Holmes %>%
  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, by = "word"
```

All the words analysis shows that this text – Story of Holmes has a lot of negative words. The frequency of negative words is higher rank than positive, and there are more negative words at wordclouds.

Looking at units beyond just words

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

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

wordcounts <- tidy_Holmes %>%
  group_by( chapter) %>%
  summarize(words = n())

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

```
## Joining, by = "word"
```

```
## # A tibble: 15 x 4
##    chapter negativewords words  ratio
##      <int>         <int> <int>  <dbl>
##  1       1            30   760 0.0395
##  2       2           127  1333 0.0953
##  3       3            87   886 0.0982
##  4       4            85  1255 0.0677
##  5       5            73  1048 0.0697
##  6       6           133  1325 0.100
##  7       7           199  1527 0.130
##  8       8            91   863 0.105
##  9       9           256  1860 0.138
## 10      10           113  1034 0.109
## 11      11           151  1408 0.107
## 12      12           236  1496 0.158
## 13      13            75  1016 0.0738
## 14      14           188  1473 0.128
## 15      15           166  1310 0.127
```

Another Lexicon

I find loughran lexicon. And I try to use that to reflect on the results of the book.

```
loughranwords<-get_sentiments("loughran")
table(loughranwords$sentiment)
```

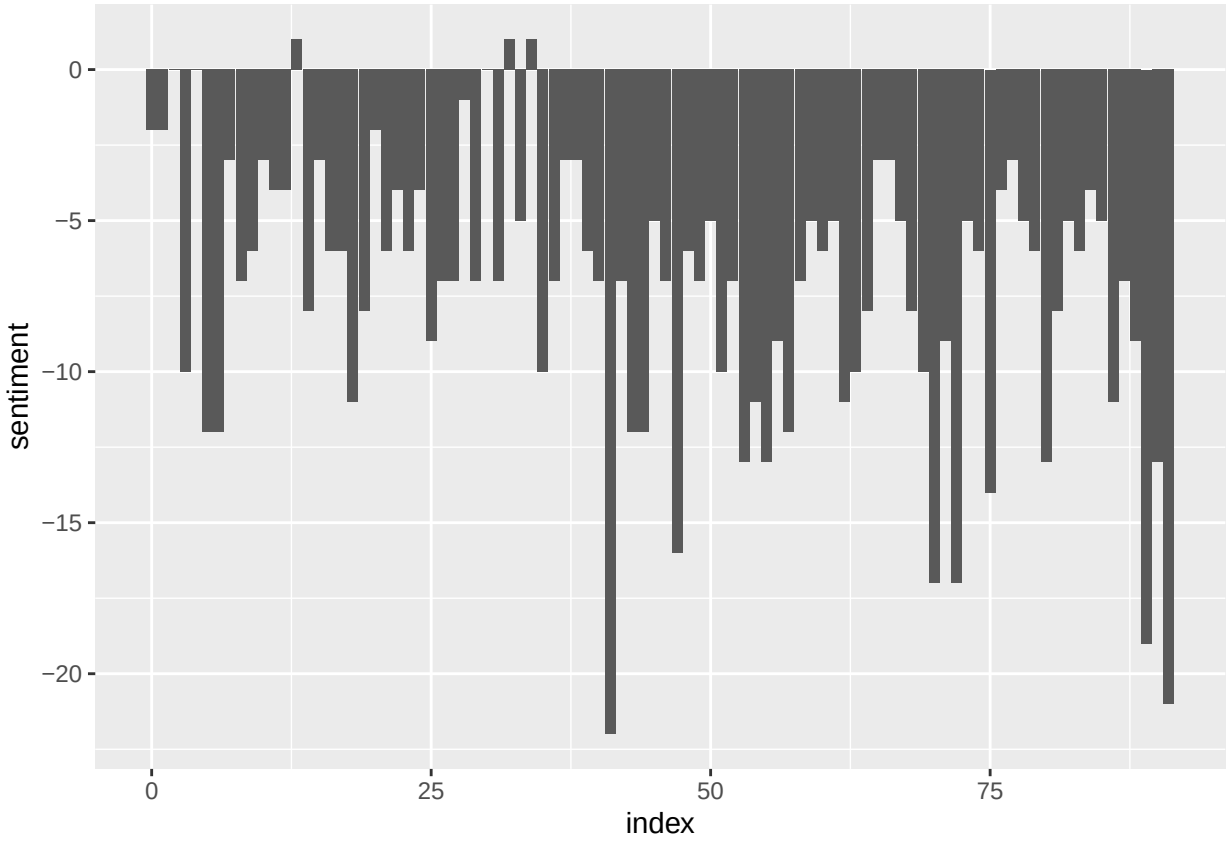
```
##
## constraining    litigious     negative     positive  superfluous  uncertainty
##          184          904         2355          354           56          297
```

It contains much more index of the words. But I try to use just the positive and negative.

```
Holmes_sentiment <- tidy_Holmes %>%
  inner_join(get_sentiments("loughran")) %>%
  count(index = linenumber %/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment, values_from = n, values_fill = 0) %>%
  mutate(sentiment = positive - negative)
```

```
## Joining, by = "word"
```

```
ggplot(Holmes_sentiment, aes(index, sentiment)) +
  geom_col(show.legend = FALSE)
```




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

```
## Joining, by = "word"
```

I find that it can be seen from the plot that the negative number of (positive-negative) words means that there are many negative words. But we also know that there are more negative words in this dictionary than positive words. Therefore, although he is still the same as the previous conclusions, this book has more negative vocabulary, but this dictionary can not be arbitrarily said to be effective.

verbal description

The Hound of the Baskervilles is a suspenseful novel that outlines the process in which the protagonist Holmes and his assistant Watson encountered a suspenseful incident and resolved it. The whole writing uses a suspenseful atmosphere, so all the word analysis shows that there are many negative words in the article.

Task Three

Tnum ingester

```
tnum.authorize("mssp1.bu.edu")
tnum.setSpace("test2")

source("Book2TN-v6A-1.R")

#Holmes<-gutenberg_download(2852)
#write.table(Holmes, file = "holmes.txt", sep = "\t",row.names = F, col.names = T)

Holmes_txt<-read.table("Holmes.txt",header = T)
#tnBooksFromLines(Holmes_txt$text, "holmes/hound")

tnum.getDBPathList(taxonomy="subject", levels=1)
```

```
##  [1] ""                   "wells"              "Twain"
##  [4] "twain"              "cervantes"          "Dickens"
##  [7] "dickens"            "well"               "hw_time_1"
## [10] "hw_time_2"          "Alcott"             "alcott"
## [13] "Kafka"              "kafka"              "bronte"
## [16] "Thomas"             "thomas"             "wells5"
## [19] "Kipling"            "kipling"            "jane_eyre"
## [22] "wells6"             "wells7"             "wells8"
## [25] "wells9"             "barrie"             "wilde"
## [28] "gatsby"             "zweig"              "tnum_pp"
## [31] "randb"              "Zweig"              "glasser"
## [34] "wells10"            "wells11"            "wells12"
## [37] "hssb"               "Glasser"            "jv_ttluts"
## [40] "Austen"             "austen"             "hvsb"
## [43] "hvdsb"              "MachiavelliPrince"  "machiavelliprince"
## [46] "Machiavelli_Prince" "machiavelli_prince" "Jane_Austen"
## [49] "jane_austen"        "anton"              "william"
## [52] "howard"             "homer"              "Homer"
## [55] "gdm"                "xihao"              "je"
## [58] "Montgomery"         "montgomery"         "ballantyne"
## [61] "yuanming"           "johanna"            "Johanna"
## [64] "ballantyner"        "BallantyneR"        "holmes"
## [67] "charlse_dickens"    "Charlse_Dickens"    "love_paddington"
## [70] "mobydick"           "Maupassant"         "maupassant"
## [73] "peter_pan"          "Alexandre_Dumas"    "alexandre_dumas"
## [76] "Kipling2"           "kipling2"
```

```
#tnum.getDBPathList(taxonomy="subject", levels=2)

q4 <- tnum.query("holmes/hound# has text", max = 15)
df4 <- tnum.objectsToDf(q4)
head(df4)
```

```
##                                                 subject property
## 1                              holmes/hound/heading:0001     text
## 2 holmes/hound/section:0001/paragraph:0003/sentence:0001     text
## 3 holmes/hound/section:0001/paragraph:0003/sentence:0002     text
## 4 holmes/hound/section:0001/paragraph:0003/sentence:0003     text
## 5 holmes/hound/section:0001/paragraph:0003/sentence:0004     text
```

```
## 6 holmes/hound/section:0001/paragraph:0005/sentence:0001     text
##
## 1
## 2                                    "Mr Sherlock Holmes         Mr Sherlock Holmes, who was usually
## 3
## 4                                                          "It was a fine, thick piece of wood,       bu
## 5 ""To James Mortimer, M.R.C.S., from his friends of the       C.C.H.," was engraved upon it, with t
## 6
##   numeric.value error unit tags       date                                  guid
## 1            NA    NA   NA      2021-12-07 4c211630-3d07-4a7c-ba4c-3807ebd1ba61
## 2            NA    NA   NA      2021-12-07 58754a34-4cd7-4c7f-8cec-8e353b733e56
## 3            NA    NA   NA      2021-12-07 88138e44-699b-471e-9997-4eb6850f157e
## 4            NA    NA   NA      2021-12-07 9083c1c5-8049-40dd-937f-0f520c8abb03
## 5            NA    NA   NA      2021-12-07 5dda405c-6d55-4226-a071-8776fdb0c247
## 6            NA    NA   NA      2021-12-07 44ee0d80-9387-4f7f-9b3c-d5ee036a5126
```

You can see that in the output result, there is the path holmes, which proves that I uploaded the file to test2.

Sentimerntr

```
para_text4 <- df4 %>% pull(string.value) %>%
                      str_replace_all("\"","") %>%
                      str_flatten(collapse = " ")

hound<-get_sentences(para_text4)
sentiment(hound)
```

```
##     element_id sentence_id word_count   sentiment
##  1:          1           1         33 -0.06092718
##  2:          1           2         21  0.18548521
##  3:          1           3         33  0.00000000
##  4:          1           4         41  0.27330408
##  5:          1           5         26  0.13728129
##  6:          1           6          8  0.00000000
##  7:          1           7         28  0.24567691
##  8:          1           8         13  0.06933752
##  9:          1           9         23  0.34404878
## 10:          1          10         53  0.61125451
## 11:          1          11         52  0.63097147
## 12:          1          12         26  0.27456259
## 13:          1          13         59  0.16924558
```

```
houndall <- tnum.query("holmes/hound# has text", max = 2870) %>%tnum.objectsToDf()
```

```
## Returned 1 thru 2865 of 2865 results
```

```
houndall_sen<-get_sentences(houndall)
sentiment(houndall_sen)
```

```
##                                                    subject property
##    1:                                holmes/hound/heading:0001     text
##    2: holmes/hound/section:0001/paragraph:0003/sentence:0001     text
##    3: holmes/hound/section:0001/paragraph:0003/sentence:0002     text
##    4: holmes/hound/section:0001/paragraph:0003/sentence:0003     text
##    5: holmes/hound/section:0001/paragraph:0003/sentence:0004     text
##   ---
## 3357: holmes/hound/section:0015/paragraph:0028/sentence:0002     text
## 3358: holmes/hound/section:0015/paragraph:0028/sentence:0002     text
## 3359: holmes/hound/section:0015/paragraph:0028/sentence:0003     text
## 3360: holmes/hound/section:0015/paragraph:0029/sentence:0001     text
## 3361: holmes/hound/section:0015/paragraph:0029/sentence:0002     text
##
##    1:
##    2:                                  "Mr Sherlock Holmes         Mr Sherlock Holmes, who was usua
##    3:
##    4:                                                      "It was a fine, thick piece of wood,
##    5: ""To James Mortimer, M.R.C.S., from his friends of the       C.C.H.," was engraved upon it, wi
##   ---
## 3357:
## 3358:
## 3359:                                                     "If Stapleton came       into the s
## 3360:                                                      "How could he claim it without cau
## 3361:
##       numeric.value error unit tags       date
##    1:            NA    NA   NA      2021-12-07
##    2:            NA    NA   NA      2021-12-07
##    3:            NA    NA   NA      2021-12-07
##    4:            NA    NA   NA      2021-12-07
##    5:            NA    NA   NA      2021-12-07
##   ---
## 3357:            NA    NA   NA      2021-12-08
## 3358:            NA    NA   NA      2021-12-08
## 3359:            NA    NA   NA      2021-12-08
## 3360:            NA    NA   NA      2021-12-08
## 3361:            NA    NA   NA      2021-12-08
##                                    guid element_id sentence_id word_count
##    1: 4c211630-3d07-4a7c-ba4c-3807ebd1ba61          1           1          1
##    2: 58754a34-4cd7-4c7f-8cec-8e353b733e56          2           1         32
##    3: 88138e44-699b-471e-9997-4eb6850f157e          3           1         21
##    4: 9083c1c5-8049-40dd-937f-0f520c8abb03          4           1         33
##    5: 5dda405c-6d55-4226-a071-8776fdb0c247          5           1         41
##   ---
## 3357: 55de1f20-84c6-459a-9d0d-9255b02866df       2862           2         NA
## 3358: 55de1f20-84c6-459a-9d0d-9255b02866df       2862           3          5
## 3359: e919fb1f-3852-40ed-b9df-c3a7cd3a9111       2863           1         28
## 3360: 16df3357-c84d-461f-bf74-eb27b9c4c66c       2864           1         30
## 3361: d4d959bb-62ac-4740-ac47-01328c68a4db       2865           1         27
##         sentiment
##    1:  0.00000000
##    2: -0.06187184
##    3:  0.18548521
##    4:  0.00000000
##    5:  0.27330408
```

```
##   ---
## 3357:  0.00000000
## 3358: -0.04472136
## 3359:  0.00000000
## 3360: -0.62075223
## 3361: -0.06014065
```

```
houndall_with_pol <- houndall %>%
  get_sentences() %>%
  sentiment() %>%
  mutate(polarity_level = ifelse(sentiment < 0.2, "Negative",
                                 ifelse(sentiment > 0.2, "Positive","Neutral")))

houndall_with_pol %>% filter(polarity_level == "Negative") #%>% View()
```

```
##                                                  subject property
##    1:                               holmes/hound/heading:0001     text
##    2: holmes/hound/section:0001/paragraph:0003/sentence:0001     text
##    3: holmes/hound/section:0001/paragraph:0003/sentence:0002     text
##    4: holmes/hound/section:0001/paragraph:0003/sentence:0003     text
##    5: holmes/hound/section:0001/paragraph:0005/sentence:0001     text
##   ---
## 2730: holmes/hound/section:0015/paragraph:0028/sentence:0002     text
## 2731: holmes/hound/section:0015/paragraph:0028/sentence:0002     text
## 2732: holmes/hound/section:0015/paragraph:0028/sentence:0003     text
## 2733: holmes/hound/section:0015/paragraph:0029/sentence:0001     text
## 2734: holmes/hound/section:0015/paragraph:0029/sentence:0002     text
##
##    1:
##    2: "Mr Sherlock Holmes         Mr Sherlock Holmes, who was usually very late in the mornings,
##    3:                                                                                  "I stood up
##    4:                    "It was a fine, thick piece of wood,       bulbous-headed, of the sort which
##    5:                                                                 ""Well, Watson, what do you ma
##   ---
## 2730:
## 2731:
## 2732:                           "If Stapleton came       into the succession, how could he explain the
## 2733:                            "How could he claim it without causing suspicion and       inquiry?"
## 2734:                                                                 "The past and the present are
##       numeric.value error unit tags       date
##    1:            NA    NA   NA      2021-12-07
##    2:            NA    NA   NA      2021-12-07
##    3:            NA    NA   NA      2021-12-07
##    4:            NA    NA   NA      2021-12-07
##    5:            NA    NA   NA      2021-12-07
##   ---
## 2730:            NA    NA   NA      2021-12-08
## 2731:            NA    NA   NA      2021-12-08
## 2732:            NA    NA   NA      2021-12-08
## 2733:            NA    NA   NA      2021-12-08
## 2734:            NA    NA   NA      2021-12-08
##                                       guid element_id sentence_id word_count
##    1: 4c211630-3d07-4a7c-ba4c-3807ebd1ba61          1           1          1
##    2: 58754a34-4cd7-4c7f-8cec-8e353b733e56          2           1         32
```

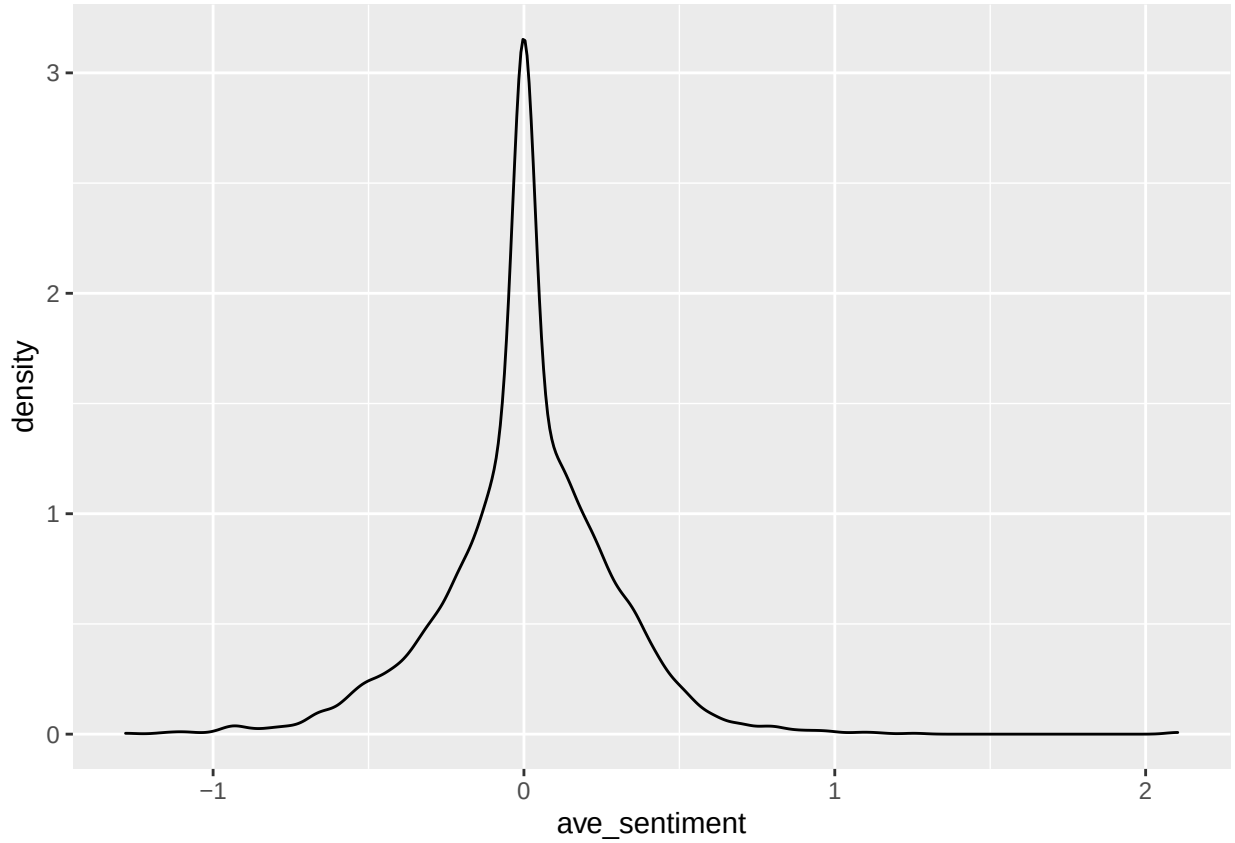
```
##    3: 88138e44-699b-471e-9997-4eb6850f157e          3          1         21
##    4: 9083c1c5-8049-40dd-937f-0f520c8abb03          4          1         33
##    5: 44ee0d80-9387-4f7f-9b3c-d5ee036a5126          6          1         26
##   ---
## 2730: 55de1f20-84c6-459a-9d0d-9255b02866df       2862          2         NA
## 2731: 55de1f20-84c6-459a-9d0d-9255b02866df       2862          3          5
## 2732: e919fb1f-3852-40ed-b9df-c3a7cd3a9111       2863          1         28
## 2733: 16df3357-c84d-461f-bf74-eb27b9c4c66c       2864          1         30
## 2734: d4d959bb-62ac-4740-ac47-01328c68a4db       2865          1         27
##         sentiment polarity_level
##    1:  0.00000000       Negative
##    2: -0.06187184       Negative
##    3:  0.18548521       Negative
##    4:  0.00000000       Negative
##    5:  0.13728129       Negative
##   ---
## 2730:  0.00000000       Negative
## 2731: -0.04472136       Negative
## 2732:  0.00000000       Negative
## 2733: -0.62075223       Negative
## 2734: -0.06014065       Negative
```

```
Holmes_sentiment <- tidy_Holmes %>%
  inner_join(get_sentiments("bing")) %>%
  count(index = linenumber , sentiment) %>%
  pivot_wider(names_from = sentiment, values_from = n, values_fill = 0) %>%
  mutate(sentiment = positive - negative)
```

```
## Joining, by = "word"
```

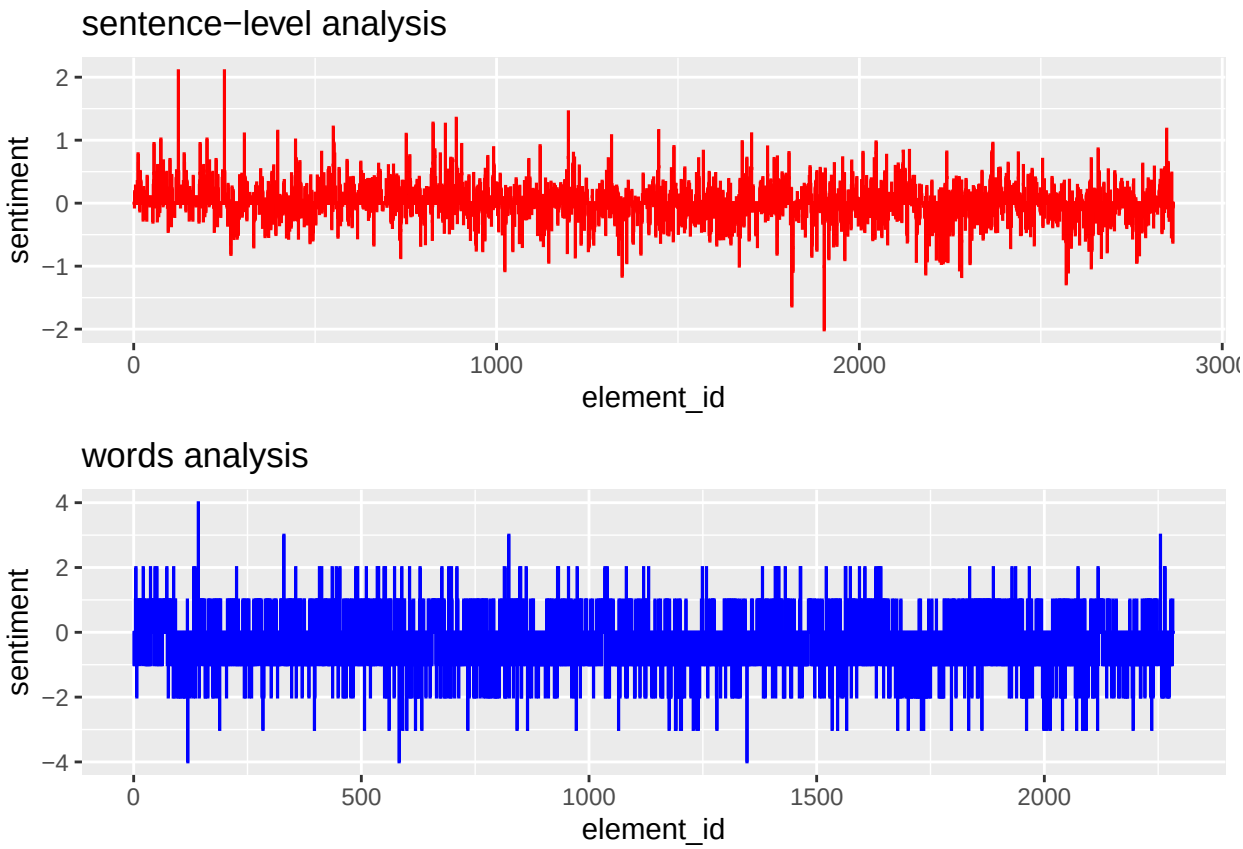
```
houndword <- Holmes_sentiment %>% mutate(element_id=NA)
for(i in 1:length(Holmes_sentiment$index)){
  houndword$element_id[i]=i
}

houndall %>%
  get_sentences() %>%
  sentiment_by(by = NULL) %>% #View()
  ggplot() + geom_density(aes(ave_sentiment))
```




```
p1<-ggplot(houndall_with_pol ) +
  geom_col(aes(element_id, sentiment),show.legend = FALSE,color="RED") +
  ggtitle("sentence-level analysis")

p2<-ggplot(houndword, aes(element_id, sentiment)) +
  geom_col(show.legend = FALSE,color="BLUE")  +
  ggtitle("words analysis")
ggpubr::ggarrange(p1,p2,nrow=2,ncol=1)
```



It can be seen from these two plots that sentence-level analysis has more results. Although the absolute value of both is within 4, the results of words analysis are all integers. The upward and downward trends are actually relatively consistent, so I think this result is meaningful.