Some Simple SPOOKY Data Analysis

Alek Anichowski

February 2, 2018

Setup the libraries

```
packages.used <- c("ggplot2", "dplyr", "tibble", "tidyr",  "stringr", "tidytext", "topicmodels", "wordcl

# check packages that need to be installed.
packages.needed <- setdiff(packages.used, intersect(installed.packages()[,1], packages.used))

# install additional packages
if(length(packages.needed) > 0) {
  install.packages(packages.needed, dependencies = TRUE, repos = 'http://cran.us.r-project.org')
}

library(ggplot2)
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(tibble)
library(tidyr)
library(stringr)
library(tidytext)
library(topicmodels)
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
library(ggridges)
library(forcats)

source("../lib/multiplot.R")
```

Read in the data

```
spooky <- read.csv('../data/spooky.csv', as.is = TRUE)
```

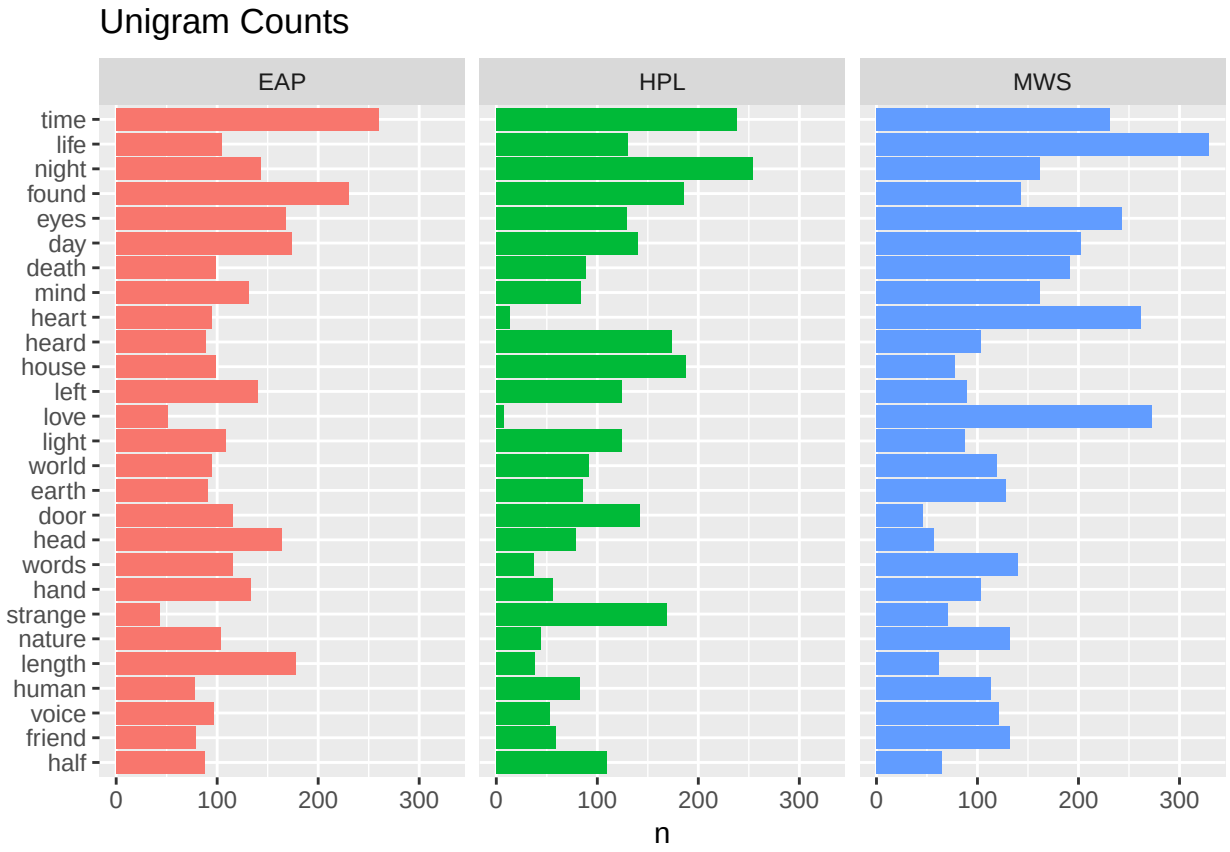
Data Cleaning

```
# Make a table with one word per row and remove `stop words` (i.e. the common words).
#1-gram
spooky_wrd <- unnest_tokens(spooky, word, text)
spooky_wrd <- anti_join(spooky_wrd, stop_words, by = "word")

#bigrams
spooky_bigrams <- unnest_tokens(spooky, bigram, text, token = "ngrams", n = 2)
bigrams_separated <- separate(spooky_bigrams,bigram, c("word1", "word2"), sep = " ")
bigrams_filtered <- bigrams_separated %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word)
bigrams_united <- bigrams_filtered %>%
  unite(bigram, word1, word2, sep = " ")

#trigrams
spooky_trigrams <- unnest_tokens(spooky, trigram, text, token = "ngrams", n = 3)
trigrams_separated <- separate(spooky_trigrams,trigram, c("word1", "word2", "word3"), sep = " ")
trigrams_filtered <- trigrams_separated %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  filter(!word3 %in% stop_words$word)
trigrams_united <- trigrams_filtered %>%
  unite(trigram, word1, word2, word3, sep = " ")
```

Unigram word counts

We see here that there are some common words to be found in the work of all the authors, “time”, “life”, “night”, “eyes”. MWS uses “love” a lot, and HPL uses “strange”

```
# Unigram
author_words <- count(group_by(spooky_wrd, word, author))

all_words    <- rename(count(group_by(spooky_wrd, word)), all = n)

author_words <- left_join(author_words, all_words, by = "word")
author_words <- arrange(author_words, desc(all))
author_words <- ungroup(head(author_words, 81))

#Plot the counts of the unigrams
ggplot(author_words) +
  geom_col(aes(reorder(word, all, FUN = min), n, fill = author)) +
  xlab(NULL) +
  coord_flip() +
  facet_wrap(~ author) +
  theme(legend.position = "none") +
  labs(title = 'Unigram Counts')
```




```
ggsave('unigram_counts.png',path='../figs')
```

```
## Saving 6.5 x 4.5 in image
```

Bigram Analysis

Here we take a look at the common bigrams that occur in the documents. Unlike the unigrams, there seem to be a lot less overlap between the authors and the bigrams they use - "short time" being a noteworthy one, maybe used to create a sense of urgency in all the texts.

Although there aren't a lot of common bigrams, we can use them to differentiate the authors. - Names like "lord raymond" or "madame lalande" or "dr armitage" are usually specific to only 1 author. - EAP and HPL are both fond of laughter, but they write it differently "ha ha" vs. "heh heh" - MWS deals with nature/animalistic themes - "fellow creatures", "native country" and "natural philosophy" are frequent - On the other hand, EAP likes material things - "chess players", "main compartment", and "tea pot" - HPL describes locations a lot, like "shunned house", "ancient house", or "tempest mountain"

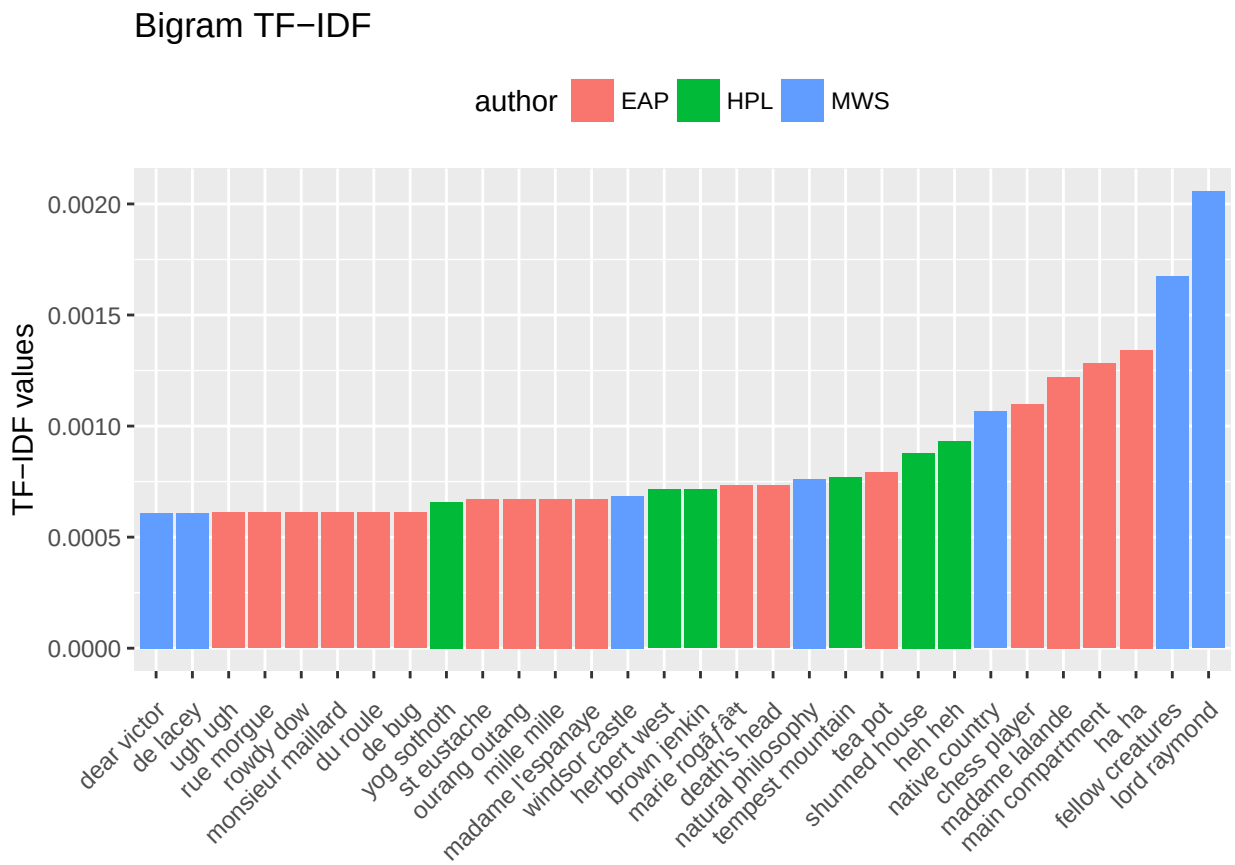
```
# Bigram

#Counts and Frequencies
author_bigrams <- count(group_by(bigrams_united, bigram, author))
all_bigrams    <- rename(count(group_by(bigrams_united, bigram)), all = n)
author_bigrams <- left_join(author_bigrams, all_bigrams, by = "bigram")
author_bigrams <- arrange(author_bigrams, desc(all))
author_bigrams <- author_bigrams[author_bigrams$all >10,]

bigrams_frequency <- count(bigrams_united, bigram, author)
```

```
bigrams_tf_idf    <- bind_tf_idf(bigrams_frequency, bigram, author, n)
bigrams_tf_idf    <- arrange(bigrams_tf_idf, desc(tf_idf))
bigrams_tf_idf    <- mutate(bigrams_tf_idf, bigram = factor(bigram, levels = rev(unique(bigram))))
bigrams_tf_idf_30 <- top_n(bigrams_tf_idf, 30, tf_idf)

#Plots
ggplot(author_bigrams) +
  geom_col(aes(reorder(bigram, all, FUN = min), n, fill = author)) +
  xlab(NULL) +
  coord_flip() +
  facet_wrap(~ author) +
  theme(legend.position = "none") +
  labs(title = "Bigram Counts by Author")
```


```
ggsave('bigram_counts.png',path='../figs')
```

```
## Saving 6.5 x 4.5 in image
```

```
ggplot(bigrams_tf_idf_30) +
  geom_col(aes(bigram, tf_idf, fill = author)) +
  labs(x = NULL, y = "TF-IDF values") +
  theme(legend.position = "top", axis.text.x  = element_text(angle=45, hjust=1, vjust=0.9)) +
  labs(title = "Bigram TF-IDF")
```




```
ggsave('bigram_tfidf.png',path='../figs')
```

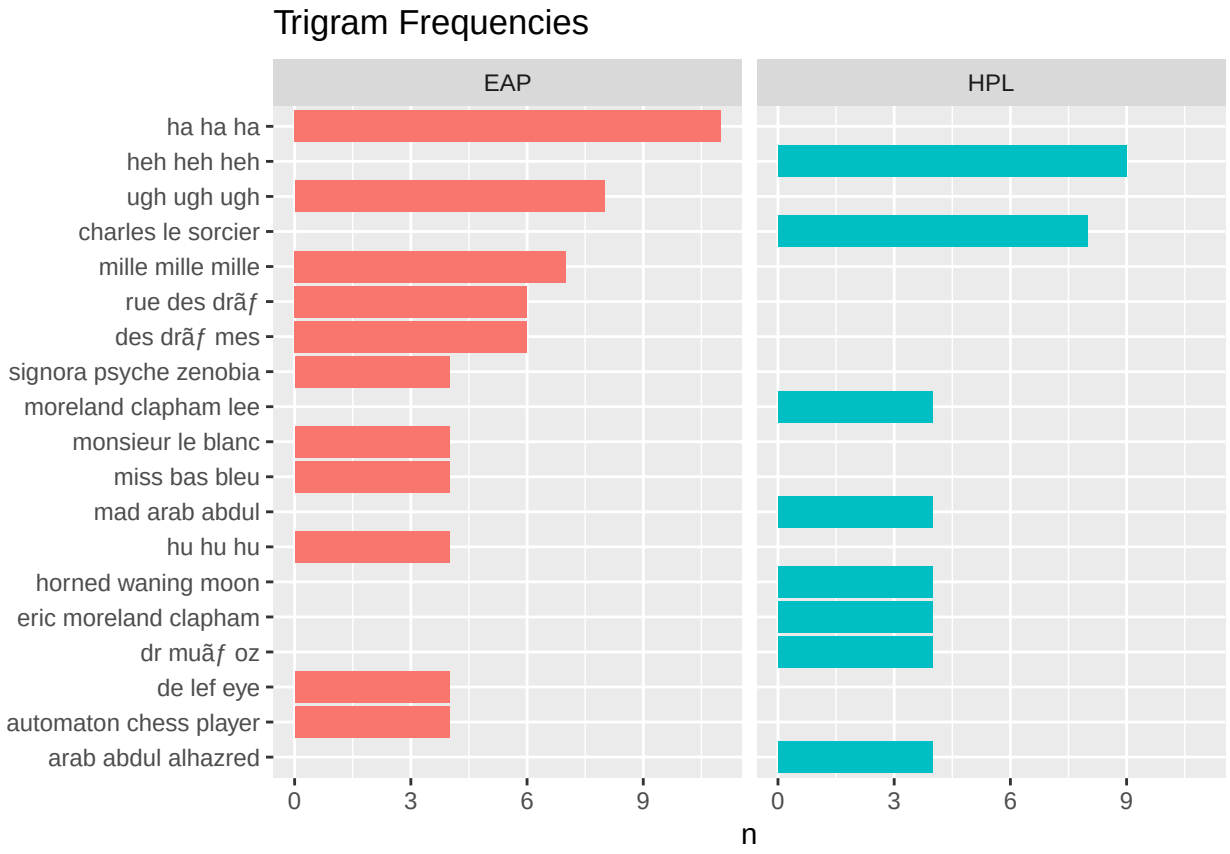
```
## Saving 6.5 x 4.5 in image
```

Trigram Analysis

It seems that only EAP and HPL have trigrams that they like to repeat, at least in this dataset. - The most common trigrams are names, specific to each author, like “charles le sorcier” or “moreland clapham lee” - EAP still likes chess players, in fact they are usually “automaton chess players” - We see laughter again, still differentiated by spelling “ha ha ha” vs “heh heh heh”. HPL’s “heh” sounds more snide

```
# Trigram

#Counts and Frequencies
author_trigrams <- count(group_by(trigrams_united, trigram, author))
all_trigrams    <- rename(count(group_by(trigrams_united, trigram)), all = n)
author_trigrams <- left_join(author_trigrams, all_trigrams, by = "trigram")
author_trigrams <- ungroup(author_trigrams[author_trigrams$all>3,])

trigrams_frequency <- count(trigrams_united, trigram, author)
trigrams_tf_idf    <- bind_tf_idf(trigrams_frequency, trigram, author, n)
trigrams_tf_idf    <- arrange(trigrams_tf_idf, desc(tf_idf))
trigrams_tf_idf    <- mutate(trigrams_tf_idf, trigram = factor(trigram, levels = rev(unique(trigram))))
trigrams_tf_idf_30 <- top_n(trigrams_tf_idf, 20, tf_idf)

#Plots
```

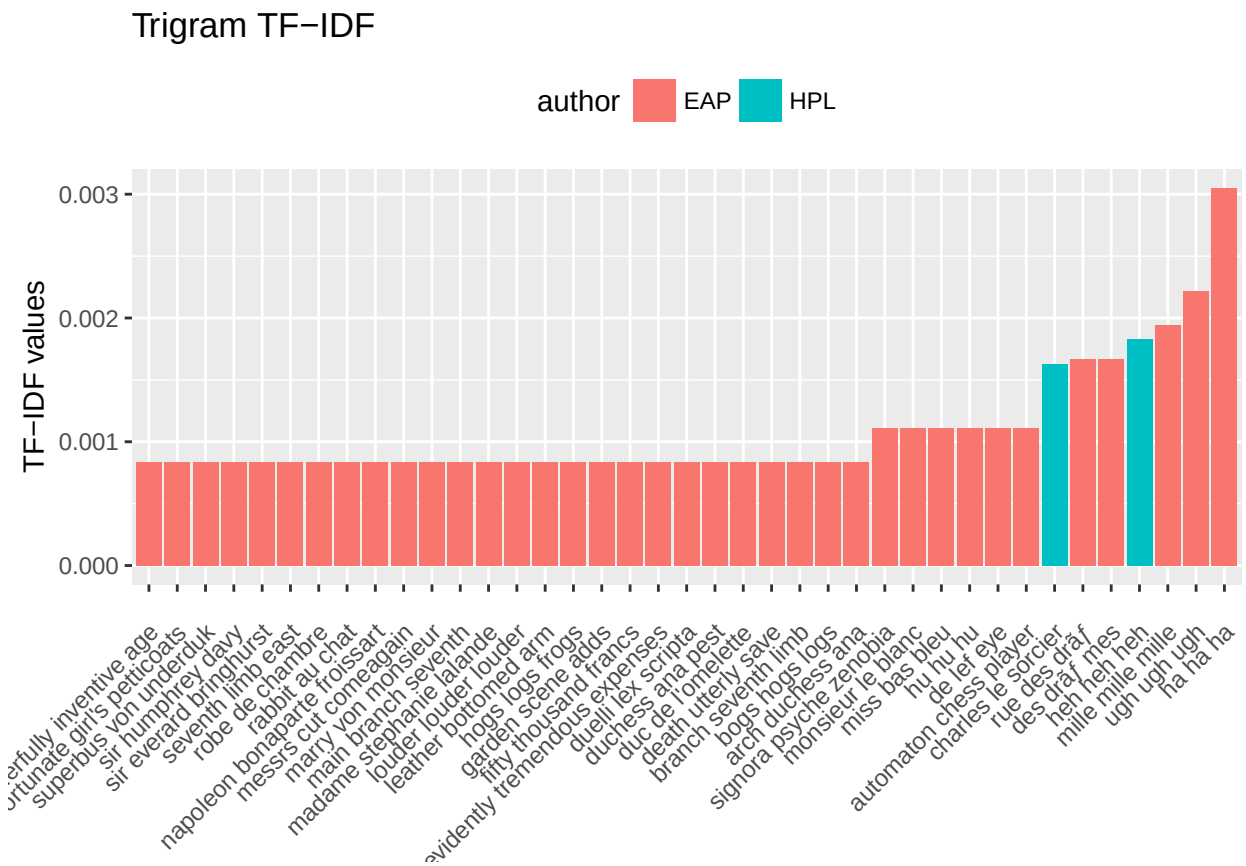
```
ggplot(author_trigrams) +
  geom_col(aes(reorder(trigram, all, FUN = min), n, fill = author)) +
  xlab(NULL) +
  coord_flip() +
  facet_wrap(~ author) +
  theme(legend.position = "none") +
  labs(title = "Trigram Frequencies")
```




```
ggsave('trigram_counts.png',path='../figs')
```

```
## Saving 6.5 x 4.5 in image
```

```
ggplot(trigrams_tf_idf_30) +
  geom_col(aes(trigram, tf_idf, fill = author)) +
  labs(x = NULL, y = "TF-IDF values") +
  theme(legend.position = "top", axis.text.x  = element_text(angle=45, hjust=1, vjust=0.9)) +
  labs(title = "Trigram TF-IDF")
```




```
ggsave('trigram_tfidf.png',path='../figs')
```

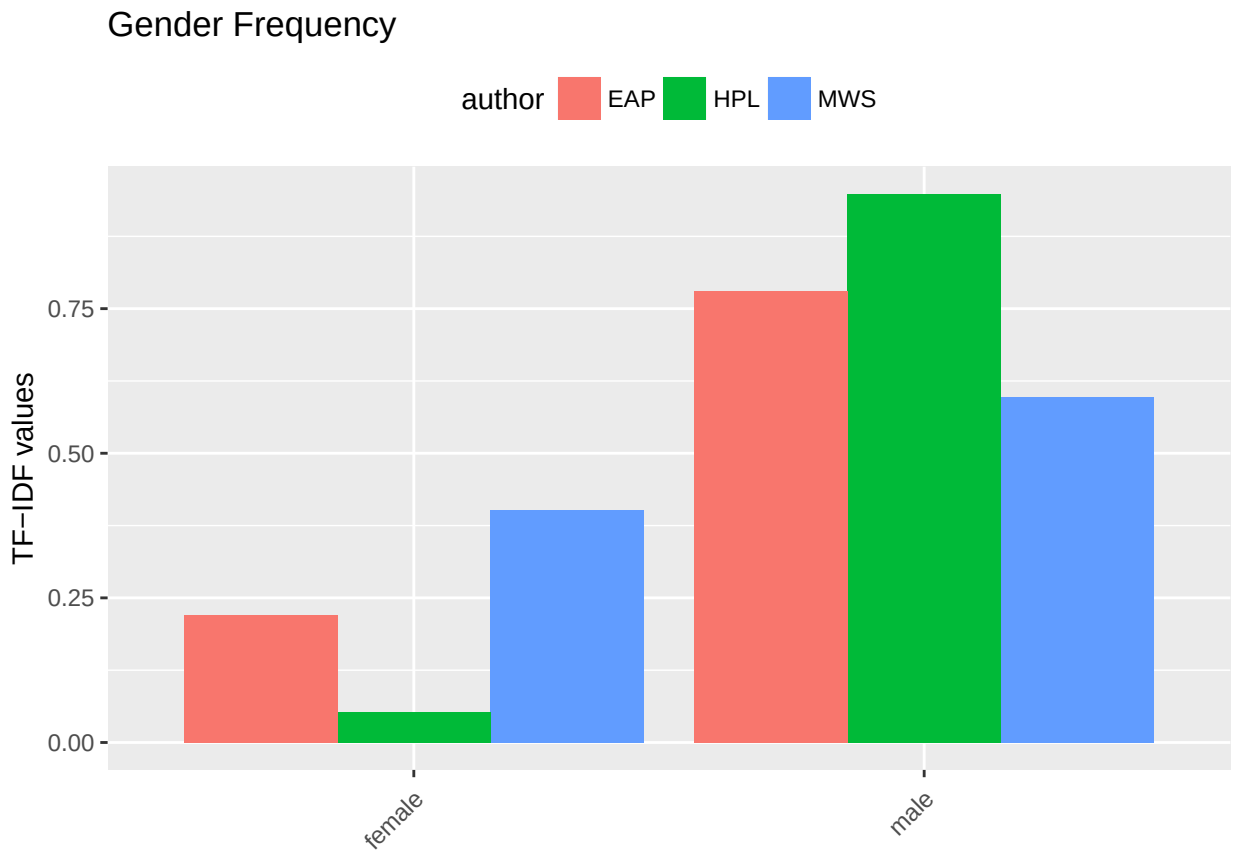
```
## Saving 6.5 x 4.5 in image
```

Genders

The gender of pronouns also differentiates the authors. Looking at the counts of gendered pronouns, MWS uses far more female words than the other two-in fact, she uses far more pronouns in general, male and female. Looking at the frequency, this analysis holds up-MWS is balanced in her use of female to male characters (around 0.4 female vs 0.6 male). EAP is more male skewed, while HPL has a large bias towards using male characters.

```
#Finding words associated with a gendered pronoun
gender_wrd = unnest_tokens(spooky, word, text) %>%
        mutate(male = (word == 'he'| word == 'him'| word == 'his'|word =='man' |word == 'gentleman'|wor
        mutate(female = (word == 'she' | word == 'her'| word == 'hers'|word =='woman'|word == 'lady'|wo
        unite(sex, male, female) %>%
        mutate(sex = fct_recode(as.factor(sex), male = "TRUE_FALSE", female = "FALSE_TRUE", na = "FALSE_
        filter(sex != "na")

#Plotting the frequencies of each gender
gender_frequency <- count(gender_wrd, sex, author)
gender_tf_idf    <- bind_tf_idf(gender_frequency, sex, author, n)

ggplot(gender_tf_idf) +
  geom_col(aes(sex, tf, fill = author),position = "dodge") +
```

```
  labs(x = NULL, y = "TF-IDF values") +
  theme(legend.position = "top", axis.text.x  = element_text(angle=45, hjust=1, vjust=0.9)) +
  labs(title = "Gender Frequency")
```




```
ggsave('gender_frequency.png',path='../figs')
```

```
## Saving 6.5 x 4.5 in image
```

Gender Sentiment Analysis

Here we isolate the sentences that mention male or females, then do sentiment analysis on the words in that sentence. First we exclude sentences that mention both male and female genders, as we are interested in the difference betweent the two.

Interestingly, since there are less female sentences overall(1837 female vs 5176 male), the 525 shared sentences(with both men and women in them) constitute a larger percentage of the female sentences. 28% of the female sentences also had men in them, while only 10% of the male sentences had women, meaning the females are mentioned along with men with higher frequency than men were mentioned with women.

Going on to the sentiment analysis, the counts are not that useful, since there are so many more male sentences, so the counts are also biased. The graph shows the frequency of sentiments across the authors, specifically the difference between the male frequency and female frequency of a certain sentiment. A positive difference means that the male sentences were more associated with this sentiment, and negative female sentences. A few trends emerge:

More significant differences: -EAP trusts men more than women, this seems to be a major trait of his sentiments towards men. -EAP and MWS associate joy and sadness with women more than men, so they are

more emotional in the corpus. -EAP's men are more positive, while the women more negative - the other author's don't have this difference between gendered pos and neg. -MWS's treatment of the genders is fairly balanced

Less significant differences: -Only in HPL are men more sad than women-in EAP and MWS sadness is associated with women more. -HPL's women are also more suprised, angry and disgusted than the men.

```
#Isolating the sentences that mention men or women
male_wrd <- gender_wrd %>%
  group_by(id) %>%
  filter(sex =='male')
male_groups <- unique(male_wrd[,1])

female_wrd <- gender_wrd %>%
  group_by(id) %>%
  filter(sex =='female')
female_groups <- unique(female_wrd[,1])

male_only <- anti_join(male_groups,female_groups)
```

```
## Joining, by = "id"
```

```
female_only <- anti_join(female_groups,male_groups)
```

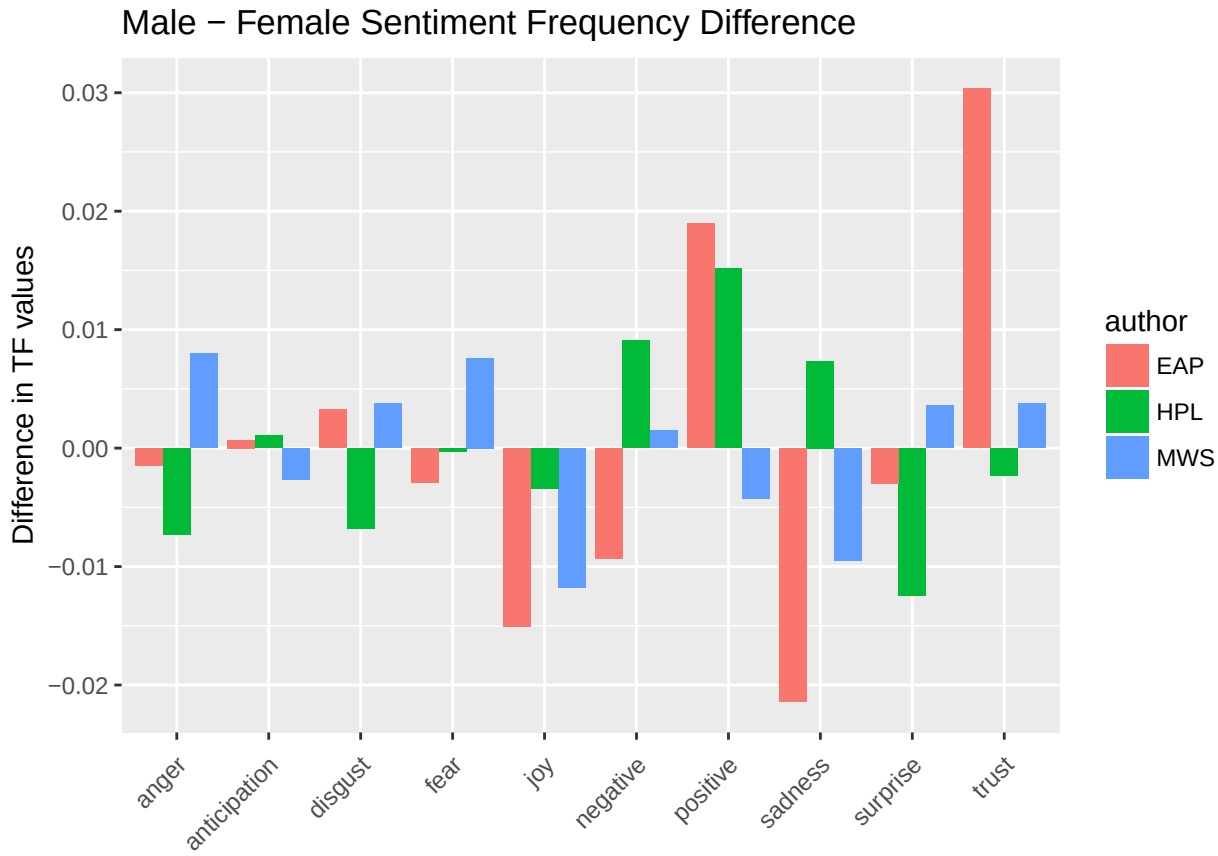
```
## Joining, by = "id"
```

```
male_lines <- spooky %>%
  filter(id %in% male_only$id)
female_lines <- spooky %>%
  filter(id %in% female_only$id)

#Tagging with sentiments
get_sentiments('nrc')
```

```
## # A tibble: 13,901 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 13,891 more rows
```

```
male_wrd <- unnest_tokens(male_lines, word, text)
female_wrd <- unnest_tokens(female_lines, word, text)

male_sentiments <- inner_join(male_wrd, get_sentiments('nrc'), by = "word")
female_sentiments <- inner_join(female_wrd, get_sentiments('nrc'), by = "word")

#Getting sentiment Frequencies
male_sentiment_frequency <- count(male_sentiments, sentiment, author)
```

```
male_tf_idf    <- bind_tf_idf(male_sentiment_frequency, sentiment, author, n)

female_sentiment_frequency <- count(female_sentiments, sentiment, author)
female_tf_idf    <- bind_tf_idf(female_sentiment_frequency, sentiment, author, n)

male_tf_idf$diff <- male_tf_idf$tf-female_tf_idf$tf

ggplot(male_tf_idf) +
  geom_col(aes(sentiment, diff, fill = author),position = "dodge") +
  labs(x = NULL, y = "Difference in TF values") +
  theme(axis.text.x  = element_text(angle=45, hjust=1, vjust=0.9)) +
  labs(title = "Male - Female Sentiment Frequency Difference")
```




```
ggsave('gender_sentiments.png',path='../figs')
```

```
## Saving 6.5 x 4.5 in image
```

Gender Wordcloud

We can look at word clouds to see what specific words are associated with each gender. This is across all the authors. Eyes, time and life are common to both, but male sentences feature the "night" while female sentences are more heavily associated with "love", "heart","beauty". Male sentences also seem to have structures in them, like "house" and "city"

```
#Finding male/female sentences again, but using stopwords removed so that the clouds are more interesti
spooky_wrd <- unnest_tokens(spooky, word, text)
```

```
spooky_wrd <- anti_join(spooky_wrd, stop_words, by = "word")

male_lines <- spooky_wrd %>%
  filter(id %in% male_groups$id)
female_lines <- spooky_wrd %>%
  filter(id %in% female_groups$id)

png('../figs/male_wordcloud.png')
male_words <- names(table(male_lines$word))
male_freqs <- table(male_lines$word)
wordcloud(male_words, male_freqs, max.words = 50, color = c("purple4", "red4", "black"))
dev.off()
```

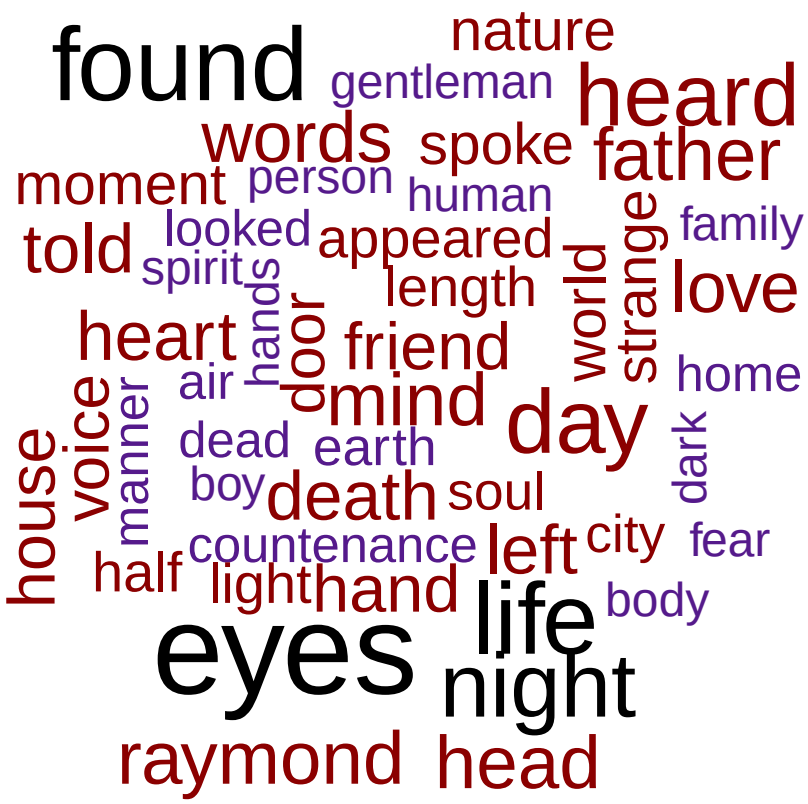
```
## pdf
##   2
```

```
png('../figs/female_wordcloud.png')
female_words <- names(table(female_lines$word))
female_freqs <- table(female_lines$word)
wordcloud(female_words, female_freqs, max.words = 50, color = c("purple4", "red4", "black"))
dev.off()
```

```
## pdf
##   2
```

```
wordcloud(male_words, male_freqs, max.words = 50, color = c("purple4", "red4", "black"))
```

```
## Warning in wordcloud(male_words, male_freqs, max.words = 50, color =
## c("purple4", : time could not be fit on page. It will not be plotted.
```




```
wordcloud(female_words, female_freqs, max.words = 50, color = c("purple4", "red4", "black"))
```

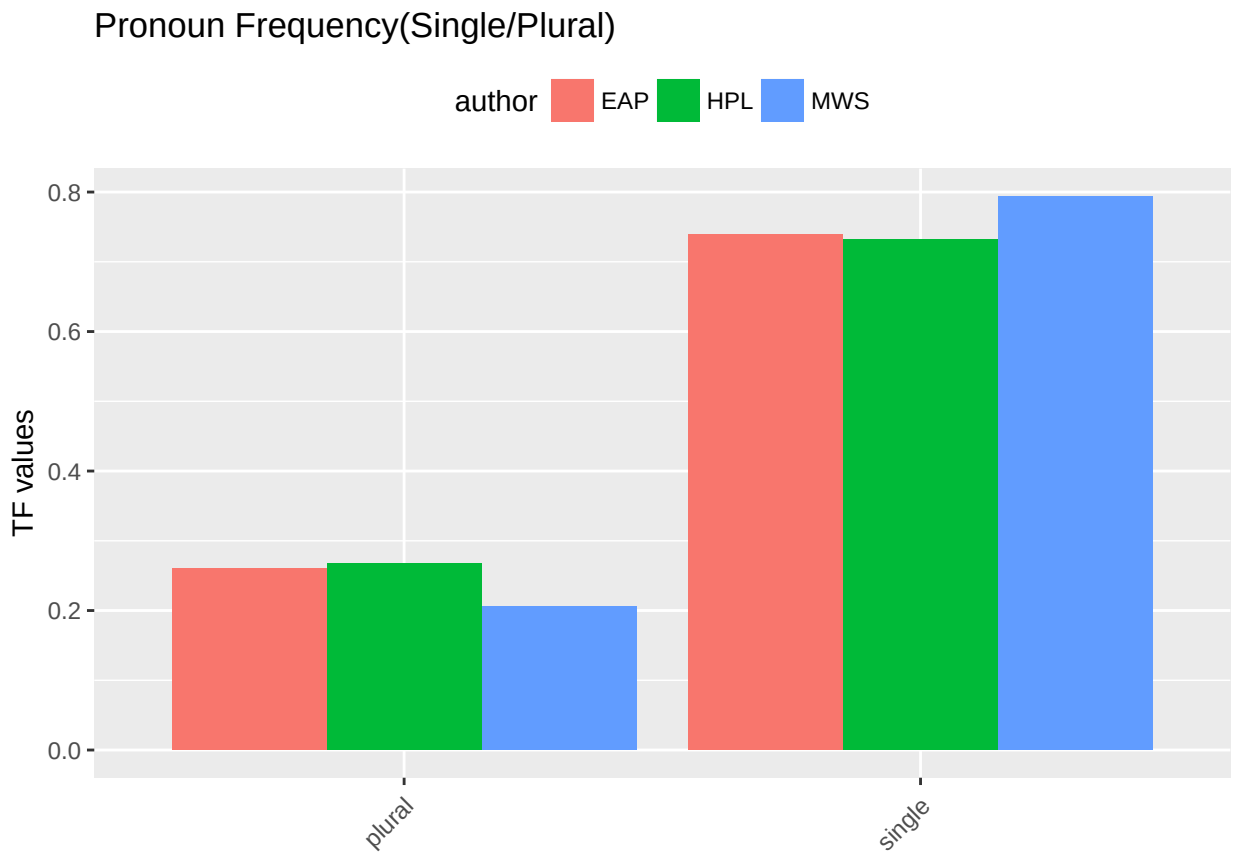
```
## Warning in wordcloud(female_words, female_freqs, max.words = 50, color =
## c("purple4", : woman could not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(female_words, female_freqs, max.words = 50, color =
## c("purple4", : love could not be fit on page. It will not be plotted.
```

Singular vs Plural Pronouns

Looking at singular vs plural pronouns, we see that all authors use the singular more than the plural, at almost the same rate in fact. MWS does tend to use the singular a bit more (80/20 split instead of 70/30 like EAP and HPL)

```
#Finding single/plural pronouns
pronoun_wrd = unnest_tokens(spooky, word, text) %>%
        mutate(single = (word == 'he'| word == 'him'| word == 'his'|word =='she' |word == 'her'|word ==
        mutate(plural = (word == 'we'|word=='us'|word=='our'|word=='ours'|word=='they'|word=='them'|wor
        unite(pronoun, single, plural) %>%
        mutate(pronoun = fct_recode(as.factor(pronoun), single = "TRUE_FALSE", plural = "FALSE_TRUE", na
        filter(pronoun != "na")

pronoun_frequency <- count(pronoun_wrd, pronoun, author)
pronoun_tf_idf    <- bind_tf_idf(pronoun_frequency, pronoun, author, n)

ggplot(pronoun_tf_idf) +
  geom_col(aes(pronoun, tf, fill = author),position = "dodge") +
  labs(x = NULL, y = "TF values") +
  theme(legend.position = "top", axis.text.x  = element_text(angle=45, hjust=1, vjust=0.9)) +
  labs(title = "Pronoun Frequency(Single/Plural)")
```




```
ggsave('pronoun_frequency.png',path='../figs')
```

```
## Saving 6.5 x 4.5 in image
```

Single/Plural Pronoun Sentiment Analysis

Like with gender, we take sentences that exclusively use single or plural pronouns and analyze the difference in frequency of various sentiments in those sentences.

-In EAP the plural sentences are more positive than the single, but for HPL it's reversed, single is more positive than plural. -EAP trusts groups more, wheras HPL and MWS trust the individual. -HPL and MWS associate fear with the plural more than the single pronoun.

```
#Single/Plural Sentiment Analysis
single_wrd <- pronoun_wrd %>%
  group_by(id) %>%
  filter(pronoun =='single')
single_groups <- unique(single_wrd[,1])

plural_wrd <- pronoun_wrd %>%
  group_by(id) %>%
  filter(pronoun =='plural')
plural_groups <- unique(plural_wrd[,1])

single_only <- anti_join(single_groups,plural_groups)
```

```
## Joining, by = "id"
```

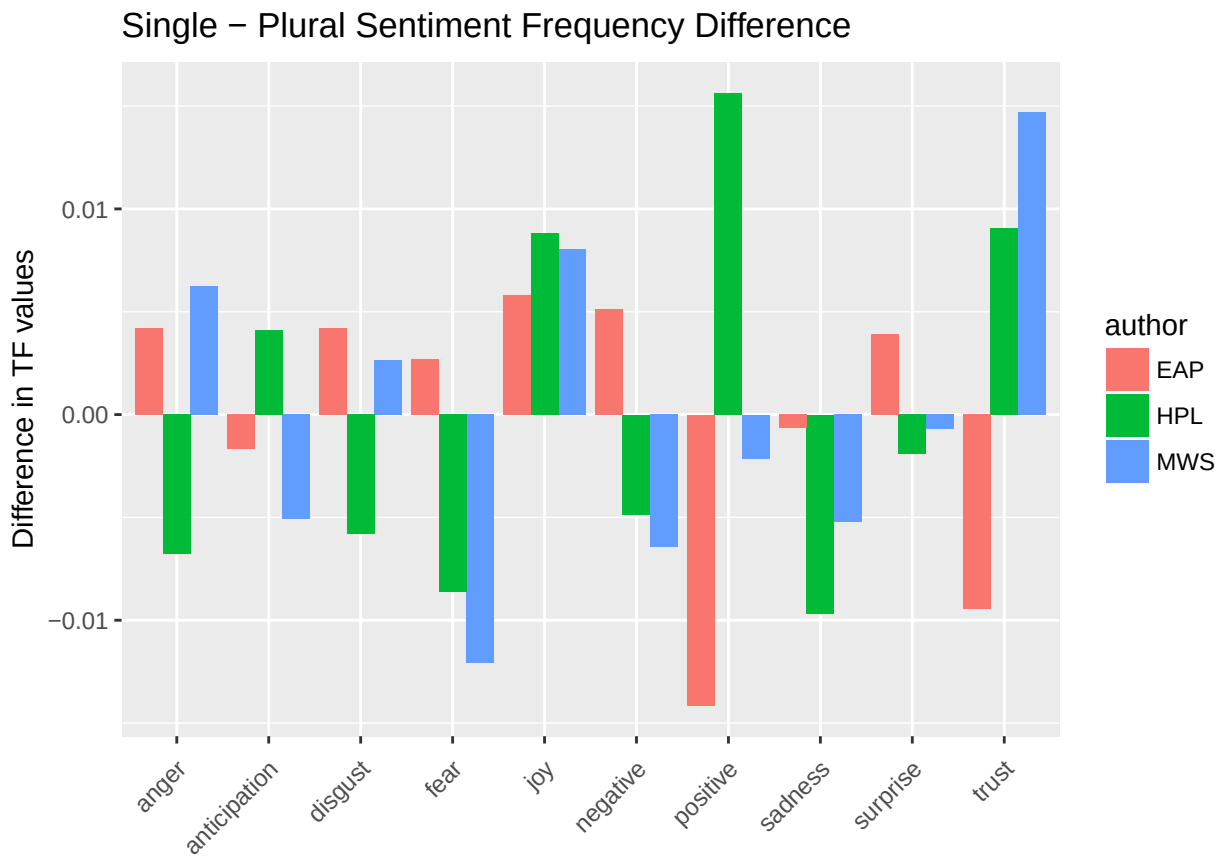
```
plural_only <- anti_join(plural_groups,single_groups)
```

```
## Joining, by = "id"
```

```
single_lines <- spooky %>%
  filter(id %in% single_only$id)
plural_lines <- spooky %>%
  filter(id %in% plural_only$id)

get_sentiments('nrc')
```

```
## # A tibble: 13,901 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 13,891 more rows
```

```
single_wrd <- unnest_tokens(single_lines, word, text)
plural_wrd <- unnest_tokens(plural_lines, word, text)

single_sentiments <- inner_join(single_wrd, get_sentiments('nrc'), by = "word")
plural_sentiments <- inner_join(plural_wrd, get_sentiments('nrc'), by = "word")

single_sentiment_frequency <- count(single_sentiments, sentiment, author)
single_tf_idf    <- bind_tf_idf(single_sentiment_frequency, sentiment, author, n)

plural_sentiment_frequency <- count(plural_sentiments, sentiment, author)
plural_tf_idf    <- bind_tf_idf(plural_sentiment_frequency, sentiment, author, n)

single_tf_idf$diff <- single_tf_idf$tf-plural_tf_idf$tf

ggplot(single_tf_idf) +
  geom_col(aes(sentiment, diff, fill = author),position = "dodge") +
  labs(x = NULL, y = "Difference in TF values") +
  theme(axis.text.x  = element_text(angle=45, hjust=1, vjust=0.9)) +
  labs(title = "Single - Plural Sentiment Frequency Difference")
```



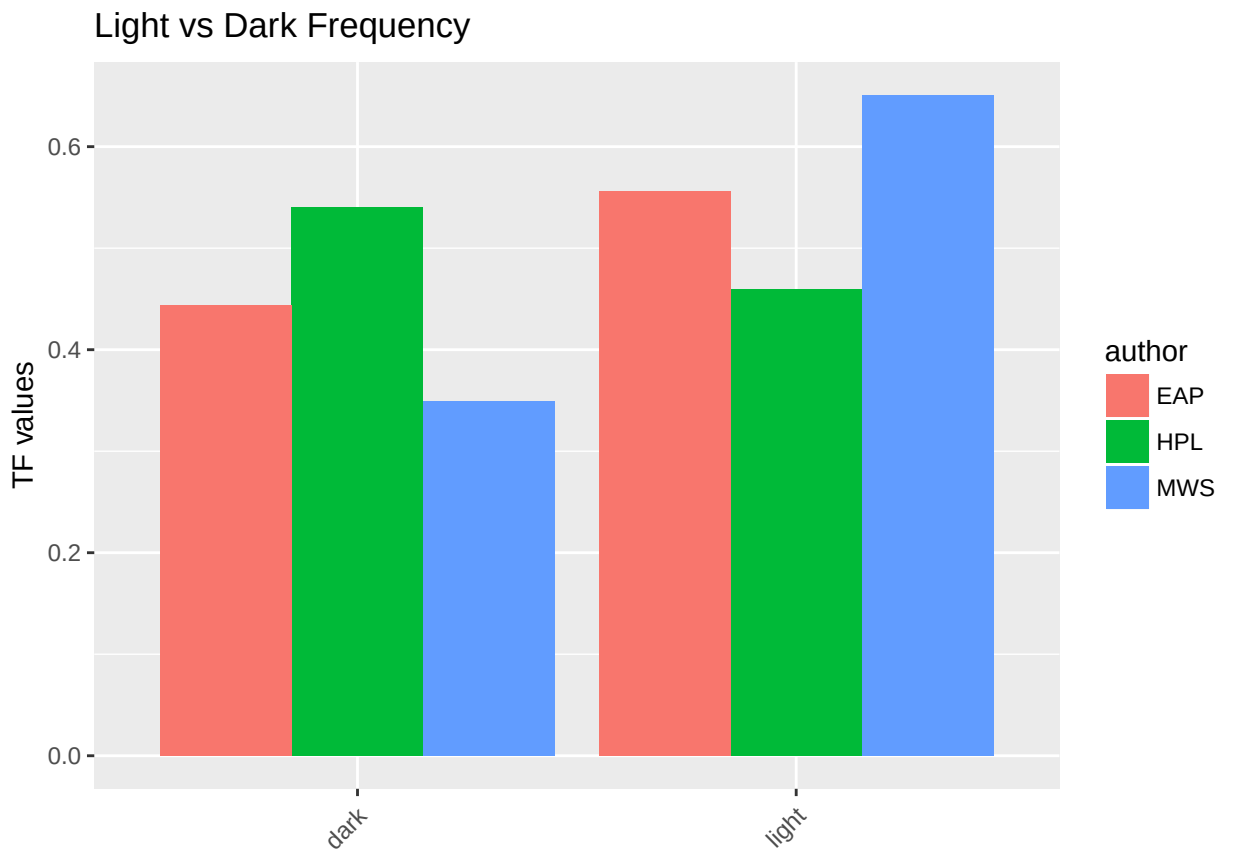

```
ggsave('pronoun_sentiments.png',path='../figs')
```

```
## Saving 6.5 x 4.5 in image
```

Light vs Dark

MWS and EAP use more light than dark imagery, and with MWS the difference is the greatest. HPL on the other hand uses more dark imagery in his works.

```
#Finding light/dark words
lights_wrd = unnest_tokens(spooky, word, text) %>%
        mutate(light = (word == 'light'|word =='bright'|word=='luminous'|word=='rich'|word=='shiny'|wor
        mutate(dark = (word == 'dark'|word=='black'|word=='grey'|word=='gray'|word=='dim'|word=='darken
        unite(light, light, dark) %>%
        mutate(light = fct_recode(as.factor(light), light = "TRUE_FALSE", dark = "FALSE_TRUE", na = "FA
        filter(light != "na")

light_frequency <- count(lights_wrd, light, author)
light_tf_idf    <- bind_tf_idf(light_frequency, light, author, n)

ggplot(light_tf_idf) +
  geom_col(aes(light, tf, fill = author),position = "dodge") +
  labs(x = NULL, y = "TF values") +
  theme(axis.text.x  = element_text(angle=45, hjust=1, vjust=0.9)) +
  labs(title = "Light vs Dark Frequency")
```




```
ggsave('lights_frequency.png',path='../figs')
```

```
## Saving 6.5 x 4.5 in image
```

Light/Dark Sentiments

When comparing light/dark sentiments, we pretty much get exactly what we hope for in terms of sentiment association, but there doesn't seem to be much differentiation between the authors. -anticipation, joy and positivity are associated with light. -negative and sadness are associated with dark. -EAP and MWS pair suprise with light, while HPL is unbiased -EAP has more disgust in his dark imagery -

```
#Light and Dark Sentiments
light_wrd <- lights_wrd %>%
  group_by(id) %>%
  filter(light =='light')
light_groups <- unique(light_wrd[,1])

dark_wrd <- lights_wrd %>%
  group_by(id) %>%
  filter(light =='dark')
dark_groups <- unique(dark_wrd[,1])

light_only <- anti_join(light_groups,dark_groups)
```

```
## Joining, by = "id"
```

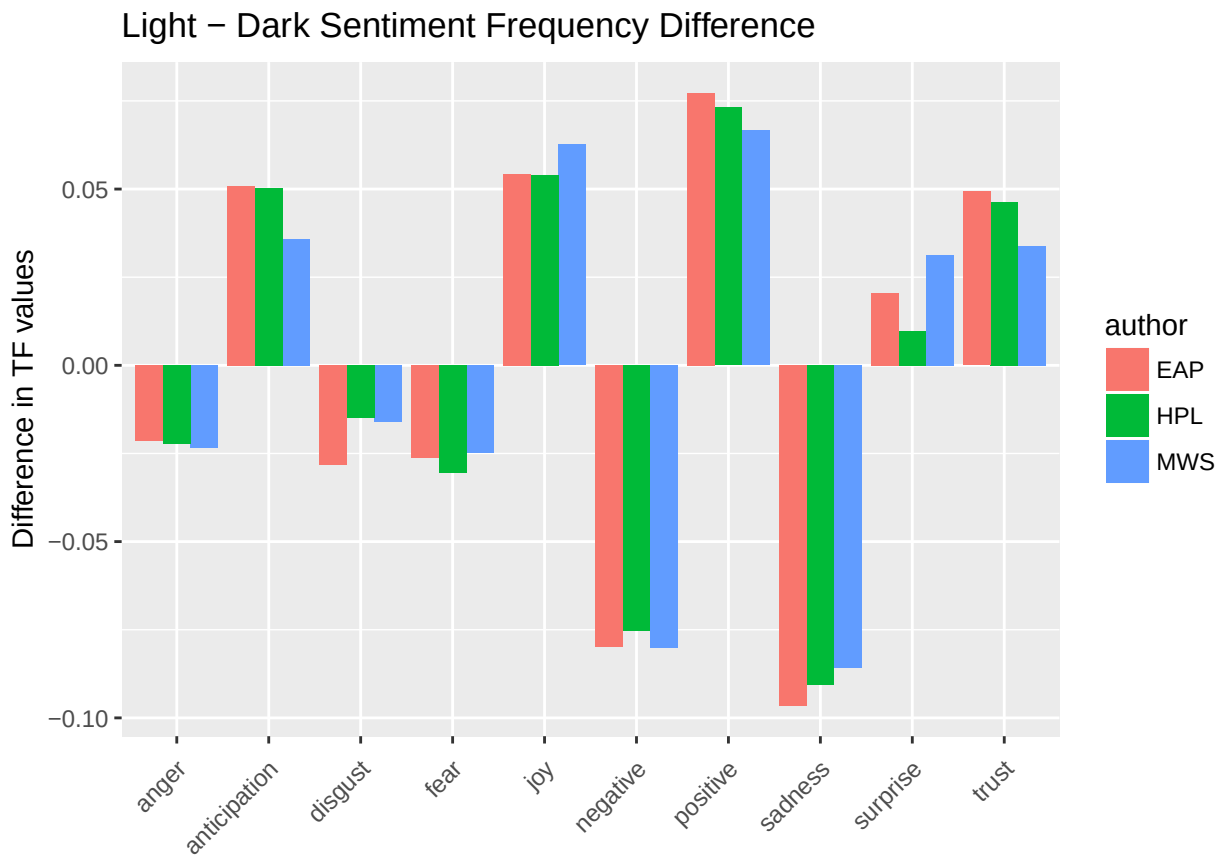
```
dark_only <- anti_join(dark_groups,light_groups)
```

```
## Joining, by = "id"
```

```
light_lines <- spooky %>%
  filter(id %in% light_only$id)
dark_lines <- spooky %>%
  filter(id %in% dark_only$id)

get_sentiments('nrc')
```

```
## # A tibble: 13,901 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 13,891 more rows
```

```
light_wrd <- unnest_tokens(light_lines, word, text)
dark_wrd <- unnest_tokens(dark_lines, word, text)

light_sentiments <- inner_join(light_wrd, get_sentiments('nrc'), by = "word")
dark_sentiments <- inner_join(dark_wrd, get_sentiments('nrc'), by = "word")

light_sentiment_frequency <- count(light_sentiments, sentiment, author)
light_tf_idf    <- bind_tf_idf(light_sentiment_frequency, sentiment, author, n)

dark_sentiment_frequency <- count(dark_sentiments, sentiment, author)
dark_tf_idf    <- bind_tf_idf(dark_sentiment_frequency, sentiment, author, n)

light_tf_idf$diff <- light_tf_idf$tf-dark_tf_idf$tf

ggplot(light_tf_idf) +
  geom_col(aes(sentiment, diff, fill = author),position = "dodge") +
  labs(x = NULL, y = "Difference in TF values") +
  theme(axis.text.x  = element_text(angle=45, hjust=1, vjust=0.9)) +
  labs(title = "Light - Dark Sentiment Frequency Difference")
```




```
ggsave('lights_sentiment.png',path='../figs')
```

```
## Saving 6.5 x 4.5 in image
```

Light/Dark Wordcloud

Here we look at the wordclouds again. Not suprisingly, the most dominant word in light sentences are "light", and for dark sentences "dark" and "black", since this was how we separated the sentences, but it's comforting to know the rest of the sentence is usually consistent. This is again the accumulation of all the author's sentences.

Often the words of the other kind are included with each other - "darkness" and "dark" in the light wordcloud and "day" or "light"in the dark wordcloud.

```
#Light and dark wordcloud, with stopwords removed
spooky_wrd <- unnest_tokens(spooky, word, text)
spooky_wrd <- anti_join(spooky_wrd, stop_words, by = "word")

light_lines <- spooky_wrd %>%
  filter(id %in% light_groups$id)
dark_lines <- spooky_wrd %>%
  filter(id %in% dark_groups$id)

png('../figs/light_wordcloud.png')
light_words <- names(table(light_lines$word))
light_freqs <- table(light_lines$word)
```

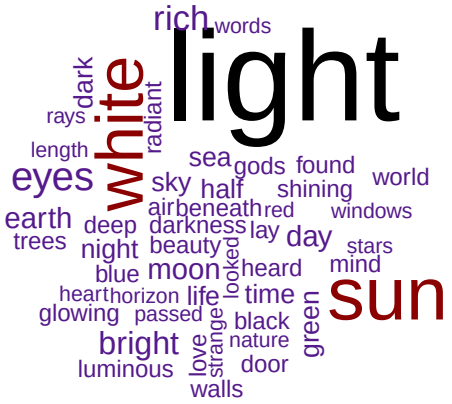
```
wordcloud(light_words, light_freqs, max.words = 50, color = c("purple4", "red4", "black"))
dev.off()
```

```
## pdf
##   2
```

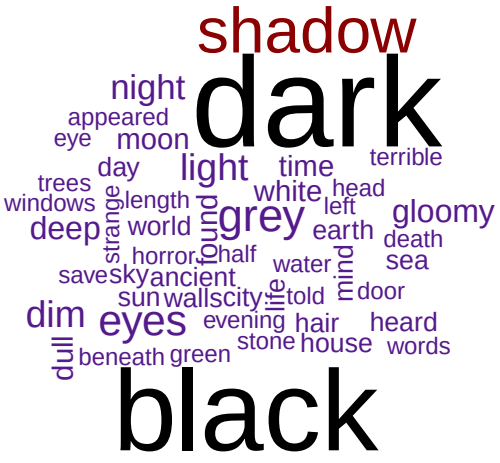
```
png('../figs/dark_wordcloud.png')
dark_words <- names(table(dark_lines$word))
dark_freqs <- table(dark_lines$word)
wordcloud(dark_words, dark_freqs, max.words = 50, color = c("purple4", "red4", "black"))
dev.off()
```

```
## pdf
##   2
```

```
wordcloud(light_words, light_freqs, max.words = 50, color = c("purple4", "red4", "black"))
```




```
wordcloud(dark_words, dark_freqs, max.words = 50, color = c("purple4", "red4", "black"))
```


shadow
dark
night
appeared
eye
moon
light
time
terrible
day
trees
white
head
windows
length
found
grey
left
gloomy
deep
world
earth
death
strange
horror
half
water
sea
mind
save
sky
ancient
sun
wall
city
told
door
dim
eyes
evening
hair
heard
dull
beneath
green
stone
house
words
black