

We then use `spread()` so that we have negative and positive sentiment in separate columns, and lastly calculate a net sentiment (positive - negative). And Now we can plot these sentiment scores across the plot trajectory of each author. Notice that we are plotting against the index on the x-axis that keeps track of text.

```
head(spooky_wrd)
```

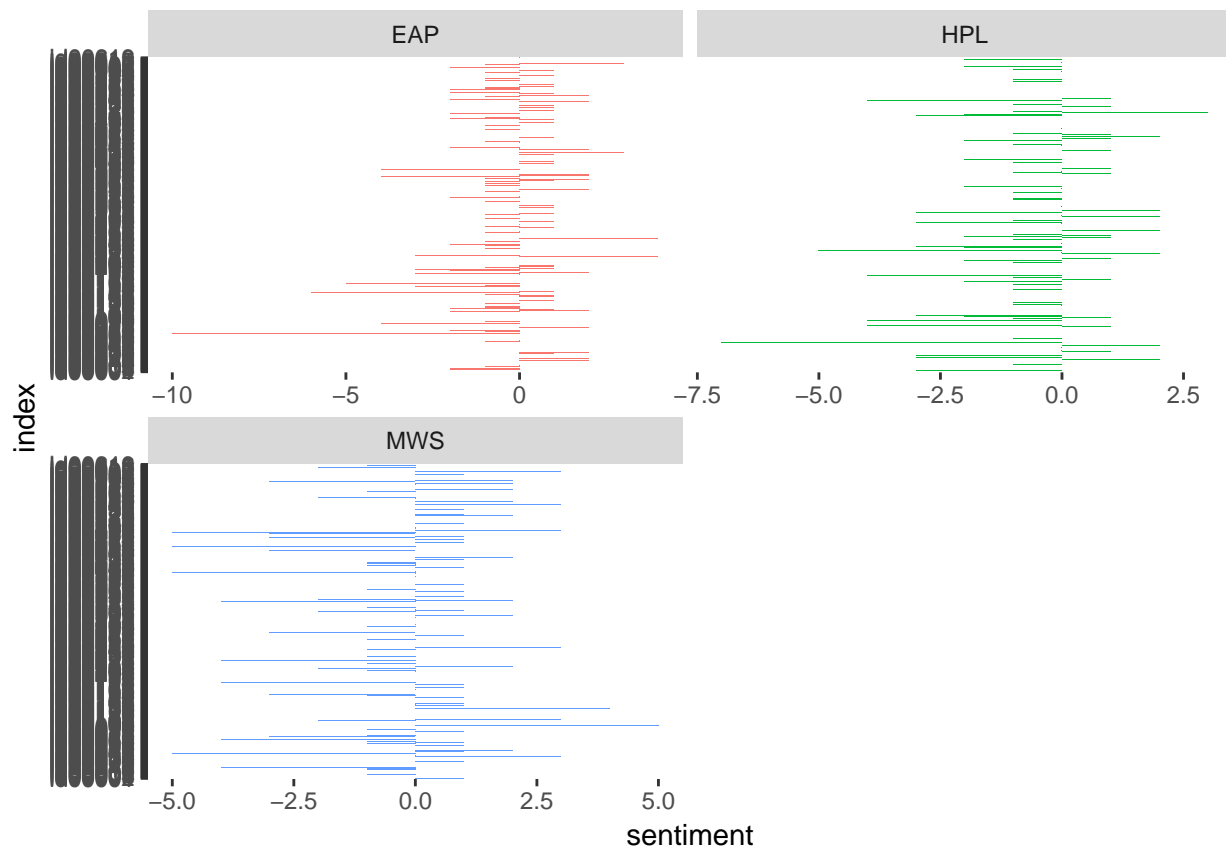
```
##           id author    word
## 1   id26305    EAP    this
## 1.1 id26305    EAP  process
## 1.2 id26305    EAP  however
## 1.3 id26305    EAP  afforded
## 1.4 id26305    EAP     me
## 1.5 id26305    EAP     no
```

```
specialsentiment<-spooky_wrd%>%inner_join(get_sentiments("bing")) %>%
  count(index=id, author, sentiment)%>%
  spread(sentiment, n, fill = 0) %>%
  mutate(sentiment = positive - negative)
```

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

```
specialsentiment_300<-head(specialsentiment,300)
```

```
ggplot(specialsentiment_300, aes(index , sentiment, fill = author)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~author, ncol = 2, scales = "free_x")+
  coord_flip()
```



We can see how the plot of each author changes toward more positive or negative sentiment over the trajectory of the story.

4: Comparing the three sentiment dictionaries

With several options for sentiment lexicons, you might want some more information on which one is appropriate for your purposes. Let's use all three sentiment lexicons and examine how the sentiment changes across the author.

```
afinn_method<-spooky_wrd%>%
  inner_join(get_sentiments("afinn")) %>%
  group_by(index =id) %>%
  summarise(sentiment = sum(score)) %>%
  mutate(method = "AFINN")

## Joining, by = "word"

afinn_300<-head(afinn_method,300)
bing_method<-spooky_wrd%>%
  inner_join(get_sentiments("bing")) %>%
  mutate(method = "Bing et al.") %>%
  count(method, index =id, sentiment) %>%
  spread(sentiment, n, fill = 0) %>%
  mutate(sentiment = positive - negative)

## Joining, by = "word"

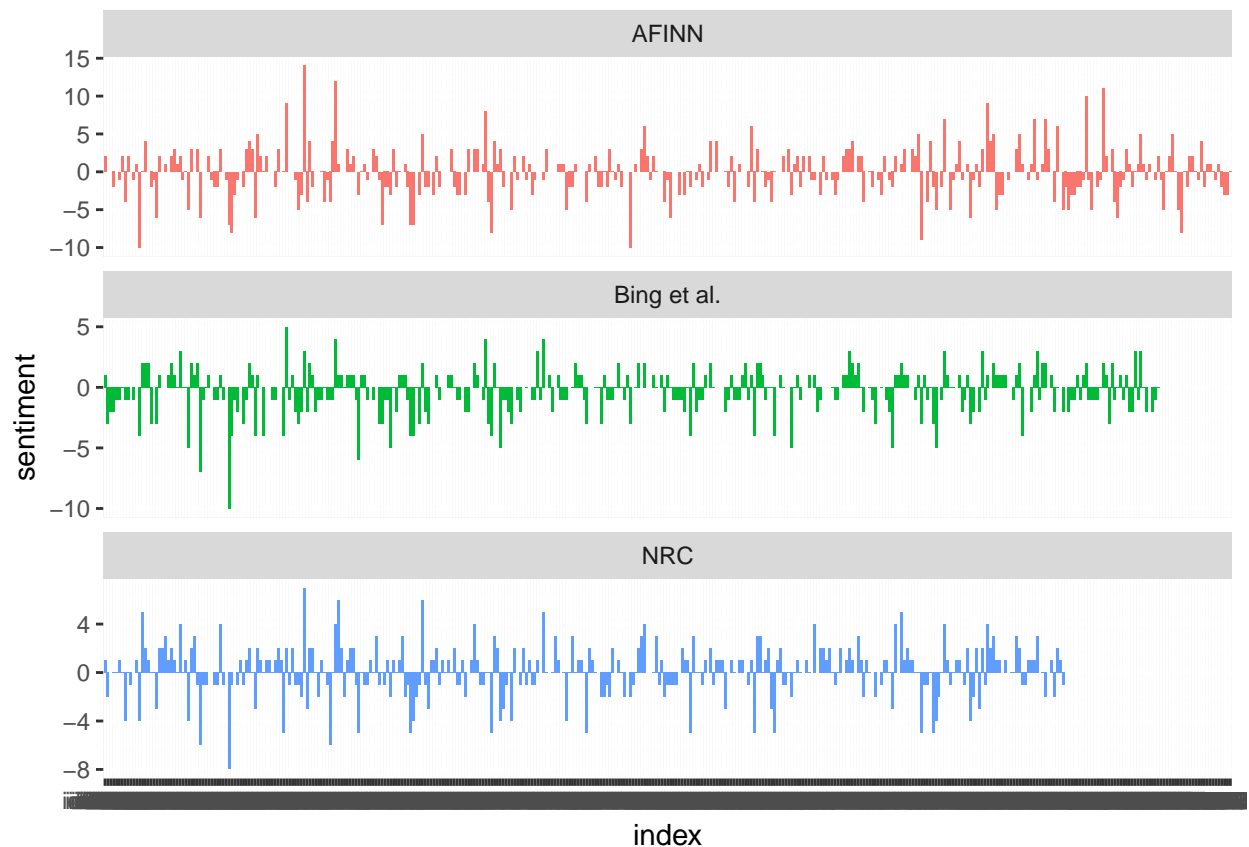
bing_300<-head(bing_method,300)
nrc_method<-spooky_wrd%>%
  inner_join(get_sentiments("nrc")) %>%
  filter(sentiment %in% c("positive","negative"))%>%
  mutate(method = "NRC")) %>%
  count(method, index =id, sentiment) %>%
  spread(sentiment, n, fill = 0) %>%
  mutate(sentiment = positive - negative)

## Joining, by = "word"

nrc_300<-head(nrc_method,300)
```

We now have an estimate of the net sentiment (positive - negative) in each chunk of the text for each sentiment lexicon. Let's bind them together and visualize them.

```
bind_rows(afinn_300,
  bing_300, nrc_300) %>%
  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 author. We see similar dips and peaks in sentiment at about the same places in the author, but the absolute values are significantly different. The AFINN lexicon gives the largest absolute values, with high positive values. The lexicon from Bing et al. has lower absolute values and seems to label larger blocks of contiguous positive or negative text. The NRC results are shifted higher relative to the other two, labeling the text more positively, but detects similar relative changes in the text. Sentiment appears to find longer stretches of similar text, but all three agree roughly on the overall trends in the sentiment through a narrative arc.