Text Mining and Sentiment Analysis

Prof. Annamaria Bianchi
A.Y. 2024/2025

Lecture 11

24 March 2025

UNIVERSITÀ DEGLI STUDI DI BERGAMO
Dipartimento di Scienze Economiche

Outline

Comparing SA based on the three sentiment dictionaries

Packages: **tidytext, tidyverse**

Functions: `base::scale()`

Sentiment Analysis

The case study I am showing today is based on the Boston Airbnb comments.

We will be using the datasets:

- bos.airbnb [with variables ID and comments]
- tidy.bos.airbnb [with variables ID and word]
- bos.pol

AFINN and NRC sentiment datasets

Let us use the NRC lexicon to answer the question:

What are the most common trust words used in the comments?

```
> nrctrust = nrc |>
+ filter(sentiment == "trust")
> tidy.bos.airbnb |>
+ inner_join(nrctrust) |>
+ count(word, sort = T)
Joining with `by = join_by(word)`
```

```
# A tibble: 235 x 2
   word           n
   <chr>      <int>
 1 clean        346
 2 recommend    218
 3 perfect      158
 4 helpful      145
 5 friendly     114
 6 wonderful    105
 7 lovely       100
 8 excellent     78
 9 found         56
10 safe          54
# ... with 225 more rows
```

Comparing the three sentiment dictionaries

With several options for sentiment lexicons, you might want some more information on which one is most appropriate for your purposes. Let us use all three sentiment lexicons and then compare the results.

Let us first add a column to the polarity score computed according to the Bing lexicon:

```
> sentiment.B = bos.pol |>
+ select(ID, sentiment) |>
+ mutate(method = "bing")
```

	ID	sentiment	method
1	1	4	Bing
2	2	3	Bing
3	3	3	Bing
4	4	6	Bing
5	5	2	Bing

Comparing the three sentiment dictionaries

Let us now compute the polarity score according to the *NRC* lexicon:

```
> sentiment.N = tidy.bos.airbnb |>
+ inner_join(nrc) |>
+ filter(sentiment %in% c("positive", "negative")) |>
+ count(ID, sentiment) |>
+ pivot_wider(names_from = sentiment, values_from = n, values_fill =0) |>
+ mutate(sentiment = positive - negative, method = "nrc") |>
+ select(ID, sentiment, method)
> View(sentiment.N)
```

	ID	sentiment	method
1	1	5	nrc
2	2	3	nrc
3	3	2	nrc
4	4	1	nrc
5	5	0	nrc
6	6	0	nrc

Comparing the three sentiment dictionaries

Polarity score according to the AFINN lexicon

```
> sentiment.A = tidy.bos.airbnb |>
+ inner_join(afinn) |>
+ group_by(ID) |>
+ summarise(sentiment=sum(value)) |>
+ mutate(method = "AFINN")
Joining with `by = join_by(word)`
> View(sentiment.A)
```

	ID	sentiment	method
1	1	11	AFINN
2	2	6	AFINN
3	3	10	AFINN
4	4	13	AFINN

Comparing the three sentiment dictionaries

```
> sentiments_all = bind_rows(sentiment.A,
+ sentiment.B,
+ sentiment.N)
> sentiments_all
# A tibble: 2,816 × 3
      ID sentiment method
   <int>     <dbl> <chr>
 1     1        11 AFINN
 2     2         6 AFINN
 3     3        10 AFINN
 4     4        13 AFINN
 5     5         6 AFINN
 6     6         6 AFINN
 7     7         4 AFINN
 8     8         5 AFINN
 9     9         8 AFINN
10    10        19 AFINN
# … 2,806 more rows
```

Comparing the three sentiment dictionaries

Exercise. Compute the number of comments with associated sentiment for each lexicon. What do you notice? How could we explain this?

Comparing the three sentiment dictionaries

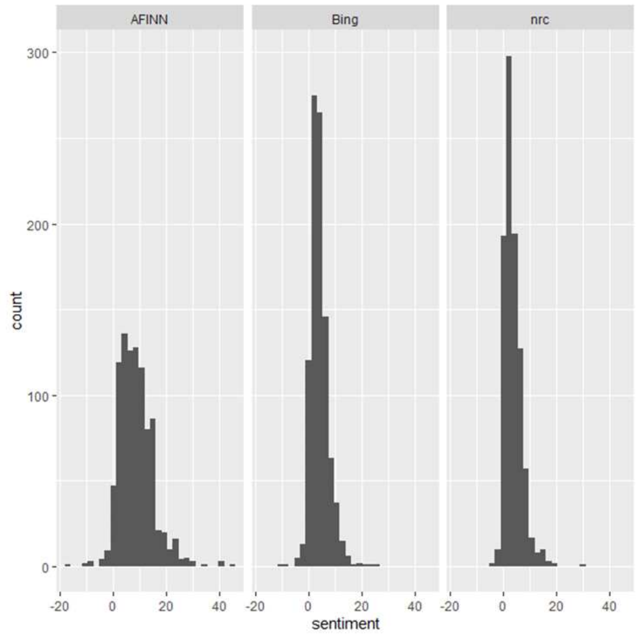
The sentiments computed have different ranges of values

```
> sentiments_all |>
+ group_by(method) |>
+ summarise(mean(sentiment), sd(sentiment), n()) |>
+ ungroup()
# A tibble: 3 × 4
method `mean(sentiment)` `sd(sentiment)` `n()`
 <chr>              <dbl>           <dbl> <int>
1 AFINN              8.59            6.38   941
2 bing               4.35            3.34   953
3 nrc                3.83            3.28   922
```

Comparing the three sentiment dictionaries

We shall also compare the distribution of the scores thorugh histograms

```
> sentiments_all |> ggplot(aes(sentiment))+
+ geom_histogram()+
+ facet_wrap(~method)
```



All distributions are concentrated on positive values

The distribution for AFINN is less concentrated than the others and more shifted towards positive values

Comparing the three sentiment dictionaries

In order to compare results from different lexicons (but also to understand the scaling effect on outcomes), it is a good practice to **scale** the sentiment scores. We shall do so using the `scale()` function, whose default method centers and scales the columns of a numeric matrix.

The `scale()` function is used for scaling and centering of matrix-like objects.

```
scale(x, center = TRUE, scale = TRUE)
```

`x`	a numeric matrix(like object).
`center`	either a logical value or numeric-alike vector of length equal to the number of columns of x
`scale`	either a logical value or a numeric-alike vector of length equal to the number of columns of x.

Comparing the three sentiment dictionaries

```
> sentiments_all = sentiments_all |>
+ group_by(method) |>
+ mutate(sentiment.std = scale(sentiment)) |>
+ ungroup()
> View(sentiments_all)
```

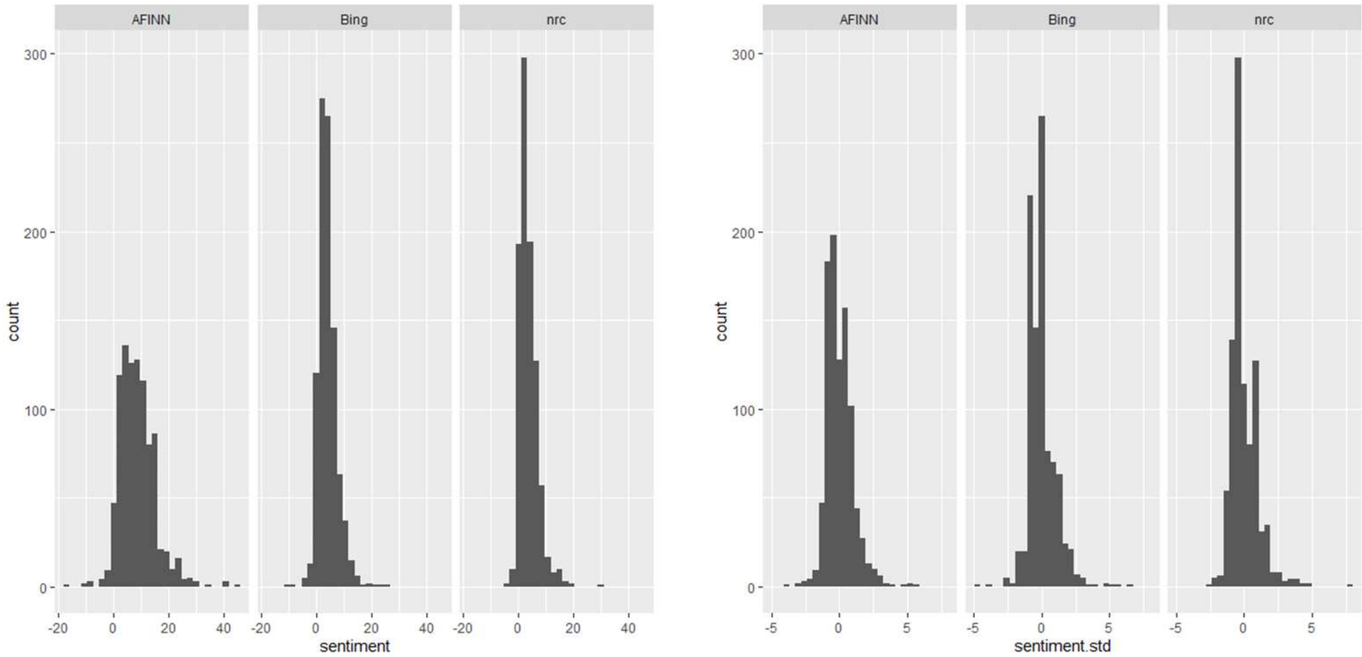
	ID	sentiment	method	sentiment.std
1	1	11	AFINN	0.37810832
2	2	6	AFINN	-0.40559196
3	3	10	AFINN	0.22136826
4	4	13	AFINN	0.69158843

Comparing the three sentiment dictionaries

```
> sentiments_all |>
+ group_by(method) |>
+ summarise(mean(sentiment.std), sd(sentiment.std), n()) |>
+ ungroup()
# A tibble: 3 × 4
  method `mean(sentiment.std)` `sd(sentiment.std)` `n()`
  <chr>                  <dbl>               <dbl> <int>
1 AFINN               3.65e-17                   1   941
2 bing                1.09e-16                   1   953
3 nrc                 2.02e-17                   1   922
```

Comparing the three sentiment dictionaries

```
> sentiments_all |>
+ ggplot(aes(sentiment.std))+
+ geom_histogram()+
+ facet_wrap(~method)
```



Comparing the three sentiment dictionaries

Let us now investigate the relationship existing among the scores.

Exercise. In order to study the relationship among the scores, we need modify the dataset and set in the wide format. Trasform the dataframe sentiments_all in wide format, so to have columns with sentiment according to the different lexicons.

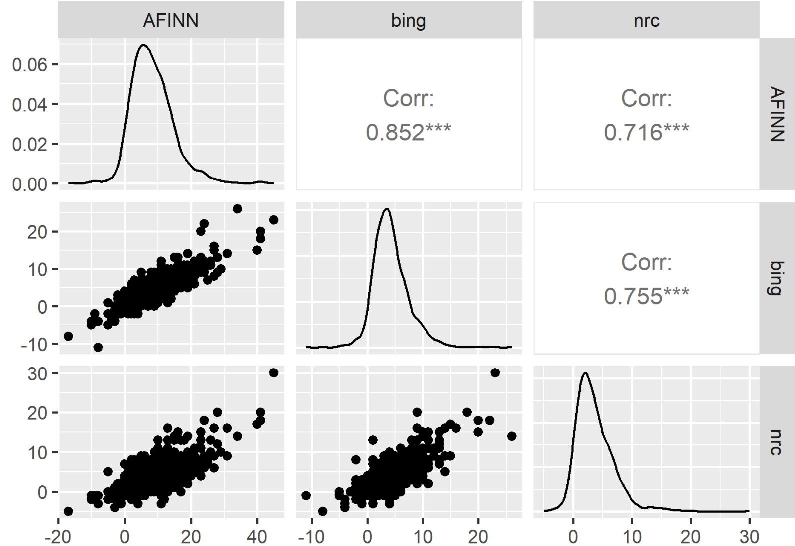
Comparing the three sentiment dictionaries

	comments	ID	afinn	bing	nrc
1	My daughter and I had a wonderful stay with Maura. She ke...	1	11	4	5
2	We stay at Elizabeth's place for 3 nights in October 2014. Th...	2	6	3	3
3	If you're staying in South Boston, this is a terrific place to ca...	3	10	3	2
4	Derian and Brian were great and prompt with	4	13	6	1
5	John and Dan were gracious hosts and the lo		6	2	0
6	The best thing about Sean's place is the locat		6	3	0
7	Tom was very welcoming and available for ea		4	5	4

If you're staying in South Boston, this is a terrific place to camp out. The apartment and bedroom are lovely, Ellie is an excellent host, and there is a lot within walking distance in a neighborhood on the rise.

Comparing the three sentiment dictionaries

```
> sentiments_all_wide |>
+ select(AFINN, bing, nrc) |>
+ cor()
          AFINN      bing       nrc
AFINN 1.0000000 0.8522714 0.7163972
bing  0.8522714 1.0000000 0.7552104
nrc   0.7163972 0.7552104 1.0000000
```




```
> install.packages('Ggally')
> library(Ggally)
> ggpairs(sentiments_all_wide, columns = c("AFINN", "bing", "nrc"))
```

Recommendation: When analyzing sentiment with tidytext, the recommendation is to compare results from the three lexicons
As you can see by each output generated, the lexicon will impact how you summarize and assess your project.

Exercise for you

Exercise 1

1. Consider the tibble sentiment.B. Which documents are not assigned a score by the bing lexicon? How do you explain that? For example, you may focus on comment n. 26. In answering the question, it might be useful to apply the function `left_join()`.
2. Consider the tibbles sentiment.B, sentiment.N, and sentiment.A. How many documents are assigned a score by all dictionaries? Are some documents assigned a score by one lexicon and not by others? How do you explain that? By way of example, you may consider comment n. 46.
3. Do you think it is possible to improve the analysis?

Exercise for you

Exercise 2
Using the function `comparison.cloud()`, build a wordcloud by emotion. If the plot does not fit the page, try playing with the arguments `scale` and `title.size`.

Exercise 3
Using ggplot2 package, build a bar plot to represent the number of words by emotion.

Exercise 4
How would you answer the original question: What quality properties are listed in positive or negative comments?