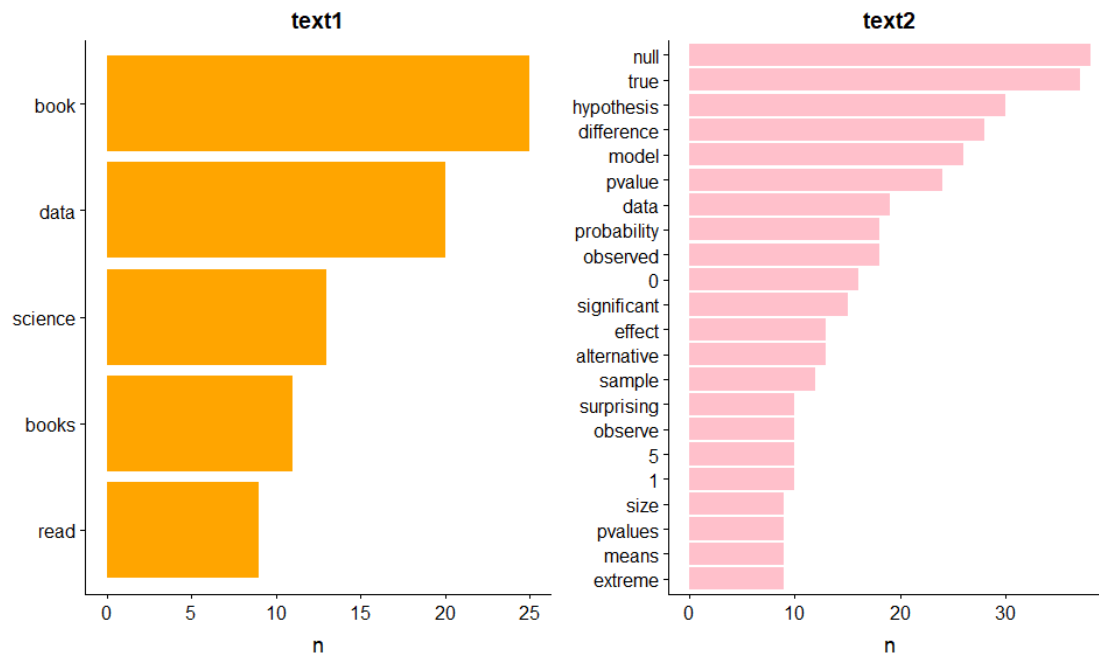


Introduction

We scraped two articles from <https://www.correlaid.org/bog>. “DATA SCIENCE IS NOT JUST ABOUT DATA SCIENCE”, and “ABOUT P-VALUES”. Based on these two articles, we analyzed tidy format, frequency, sentiment, relationships between words, and topics.

Words frequency



Words that occur over 9 times in each text

The barplot shows words that occur over 9 times in each text. We can see that they both have “data”

number of words about trust in text1

| word | n |
|---------|---|
| machine | 4 |
| system | 4 |
| found | 3 |

| | |
|---------------|---|
| related | 3 |
| architecture | 2 |
| calls | 2 |
| crucial | 2 |
| hope | 2 |
| influential | 2 |
| statistical | 2 |
| account | 1 |
| communicate | 1 |
| doubt | 1 |
| economy | 1 |
| guide | 1 |
| inspired | 1 |
| level | 1 |
| policy | 1 |
| professor | 1 |
| rational | 1 |
| scientific | 1 |
| understanding | 1 |
| wealth | 1 |

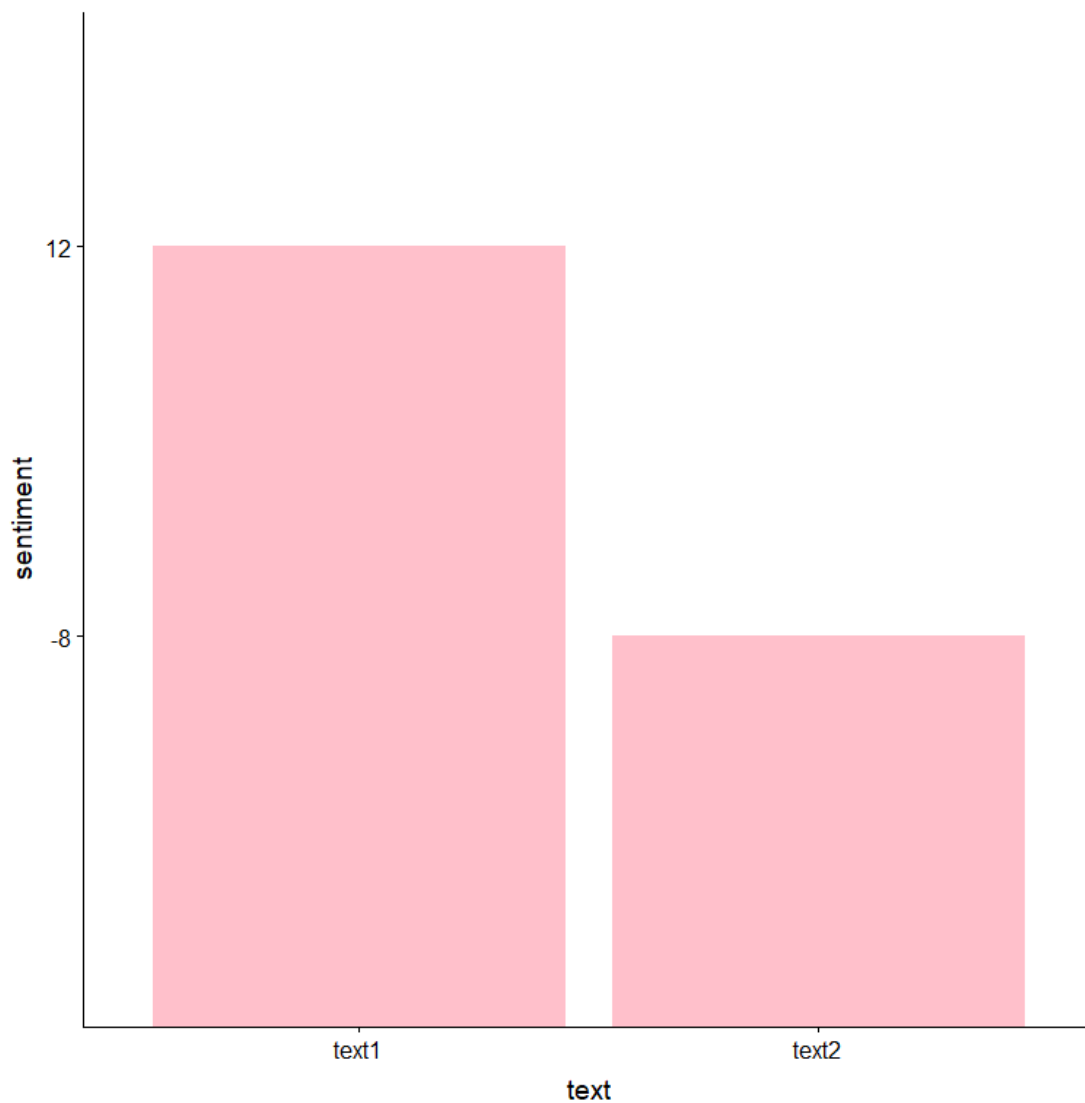
number of words about trust in text2

| word | n |
|------------|----|
| true | 37 |
| expect | 7 |
| omniscient | 6 |
| level | 4 |
| statement | 4 |
| explain | 3 |
| larger | 3 |
| theory | 3 |
| grammar | 2 |
| prefer | 2 |
| provide | 2 |
| complement | 1 |
| enable | 1 |
| finally | 1 |

| | |
|-------------|---|
| inform | 1 |
| label | 1 |
| recommend | 1 |
| series | 1 |
| statistical | 1 |

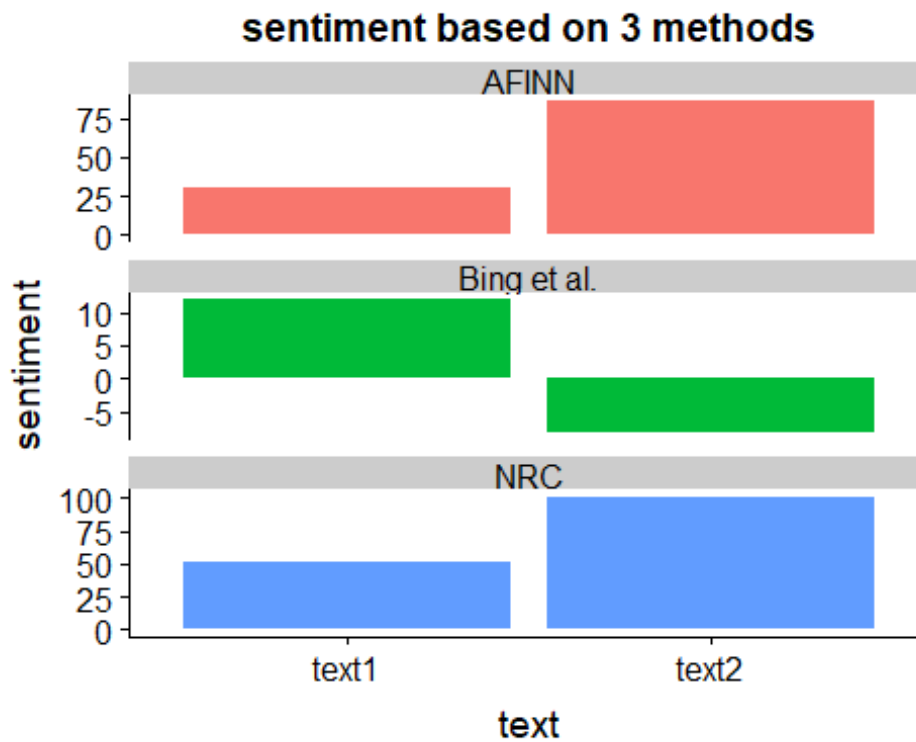
Sentiment

Overall sentiment



The plot shows the sentiment score of two texts, the text1 has higher score so that it has a better sentiment.

Sentiment based on 3 methods



sentiment based on 3 methods

The plot shows that sentiment of text2 based on Bing sentiment dataset has different result with other two sentiment dataset.

Top10 positive and negative words

top 10 positive and negative words for text1

| word | sentiment | n |
|-------------|-----------|---|
| popular | positive | 3 |
| critical | negative | 2 |
| criticism | negative | 2 |
| enjoyed | positive | 2 |
| fascinating | positive | 2 |
| fast | positive | 2 |
| influential | positive | 2 |
| master | positive | 2 |
| pessimistic | negative | 2 |
| bad | negative | 1 |
| beautiful | positive | 1 |

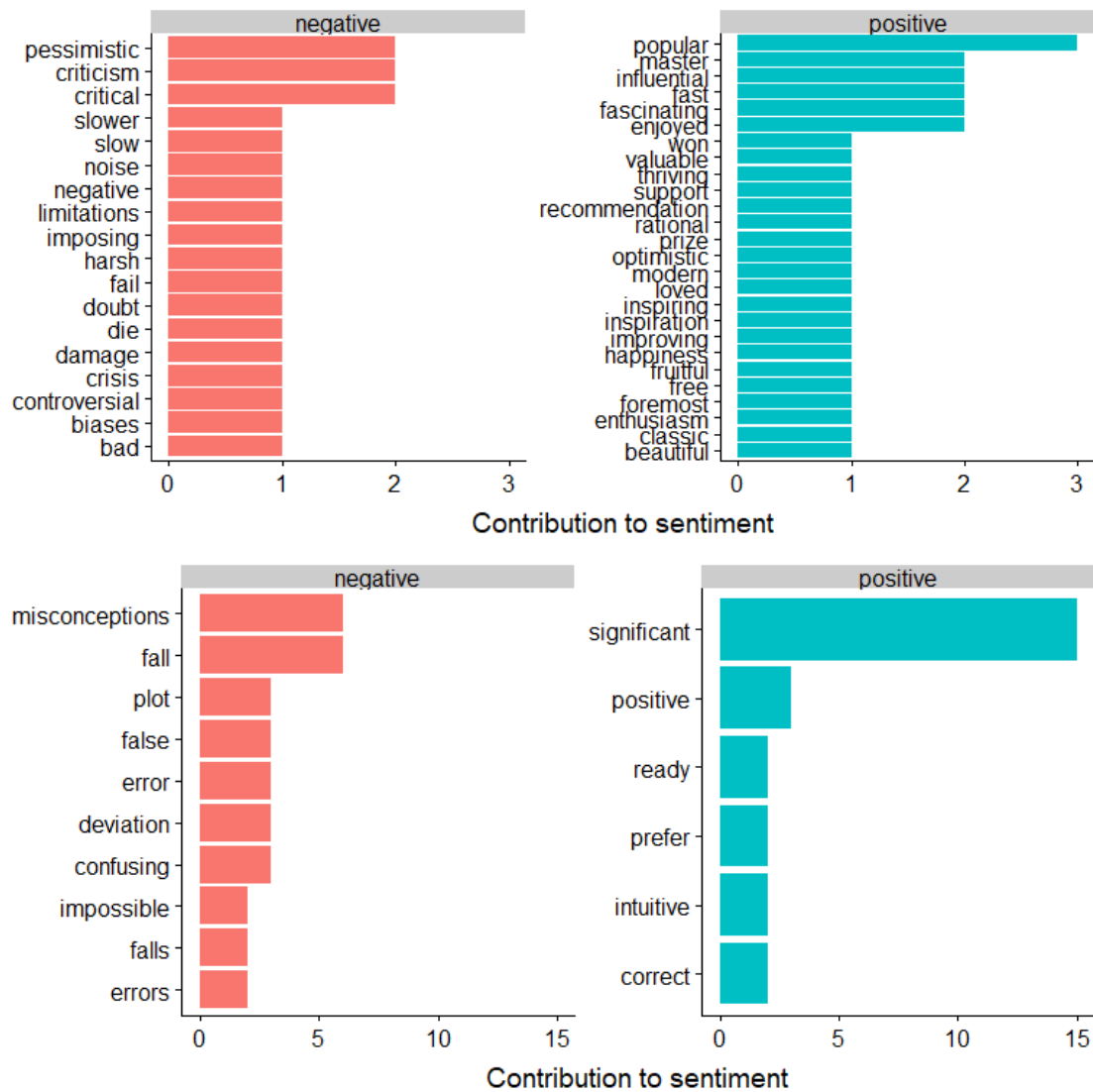
| | | |
|----------------|----------|---|
| biases | negative | 1 |
| classic | positive | 1 |
| controversial | negative | 1 |
| crisis | negative | 1 |
| damage | negative | 1 |
| die | negative | 1 |
| doubt | negative | 1 |
| enthusiasm | positive | 1 |
| fail | negative | 1 |
| foremost | positive | 1 |
| free | positive | 1 |
| fruitful | positive | 1 |
| happiness | positive | 1 |
| harsh | negative | 1 |
| imposing | negative | 1 |
| improving | positive | 1 |
| inspiration | positive | 1 |
| inspiring | positive | 1 |
| limitations | negative | 1 |
| loved | positive | 1 |
| modern | positive | 1 |
| negative | negative | 1 |
| noise | negative | 1 |
| optimistic | positive | 1 |
| prize | positive | 1 |
| rational | positive | 1 |
| recommendation | positive | 1 |
| slow | negative | 1 |
| slower | negative | 1 |
| support | positive | 1 |
| thriving | positive | 1 |
| valuable | positive | 1 |
| won | positive | 1 |

top 10 positive and negative words for text2

| word | sentiment | n |
|-------------|-----------|----|
| significant | positive | 15 |

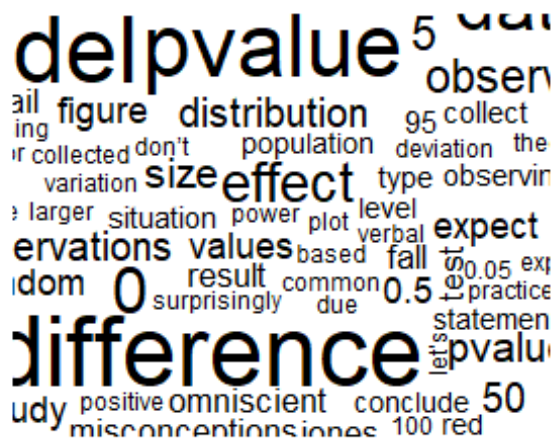
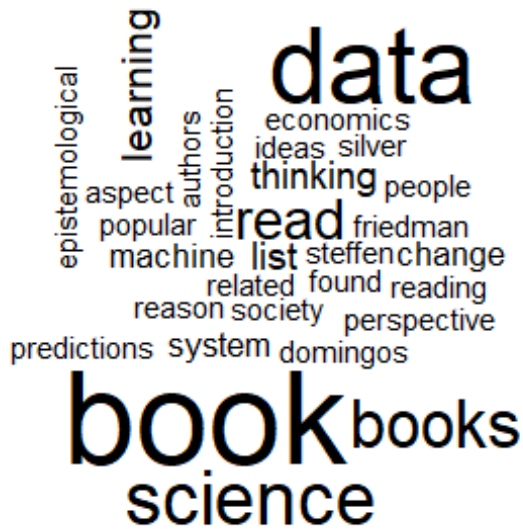
| | | |
|----------------|----------|---|
| fall | negative | 6 |
| misconceptions | negative | 6 |
| confusing | negative | 3 |
| deviation | negative | 3 |
| error | negative | 3 |
| false | negative | 3 |
| plot | negative | 3 |
| positive | positive | 3 |
| correct | positive | 2 |
| errors | negative | 2 |
| falls | negative | 2 |
| impossible | negative | 2 |
| intuitive | positive | 2 |
| prefer | positive | 2 |
| ready | positive | 2 |

Words that contribute to positive and negative sentiment



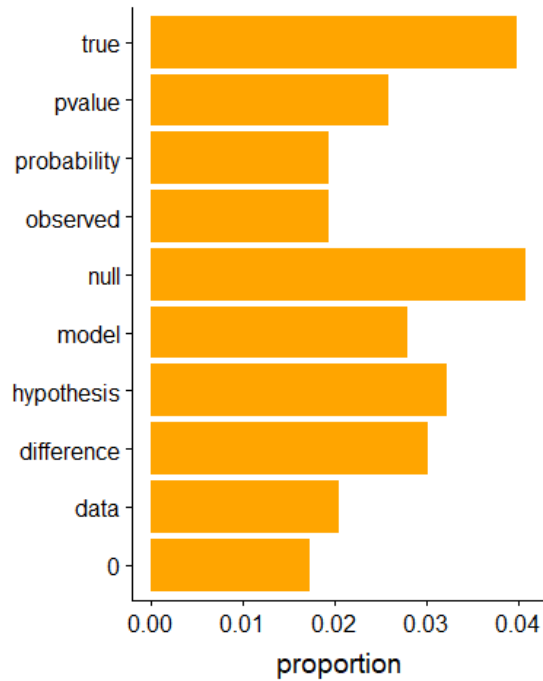
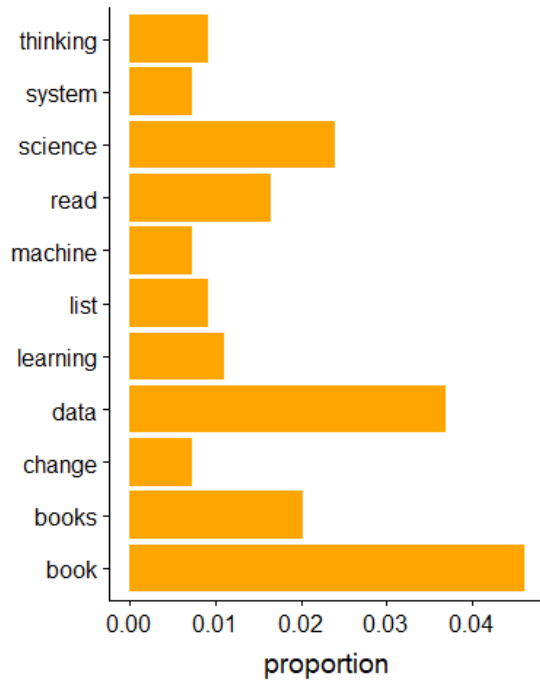
The plot shows positive and negative words on both texts.

Wordscloud



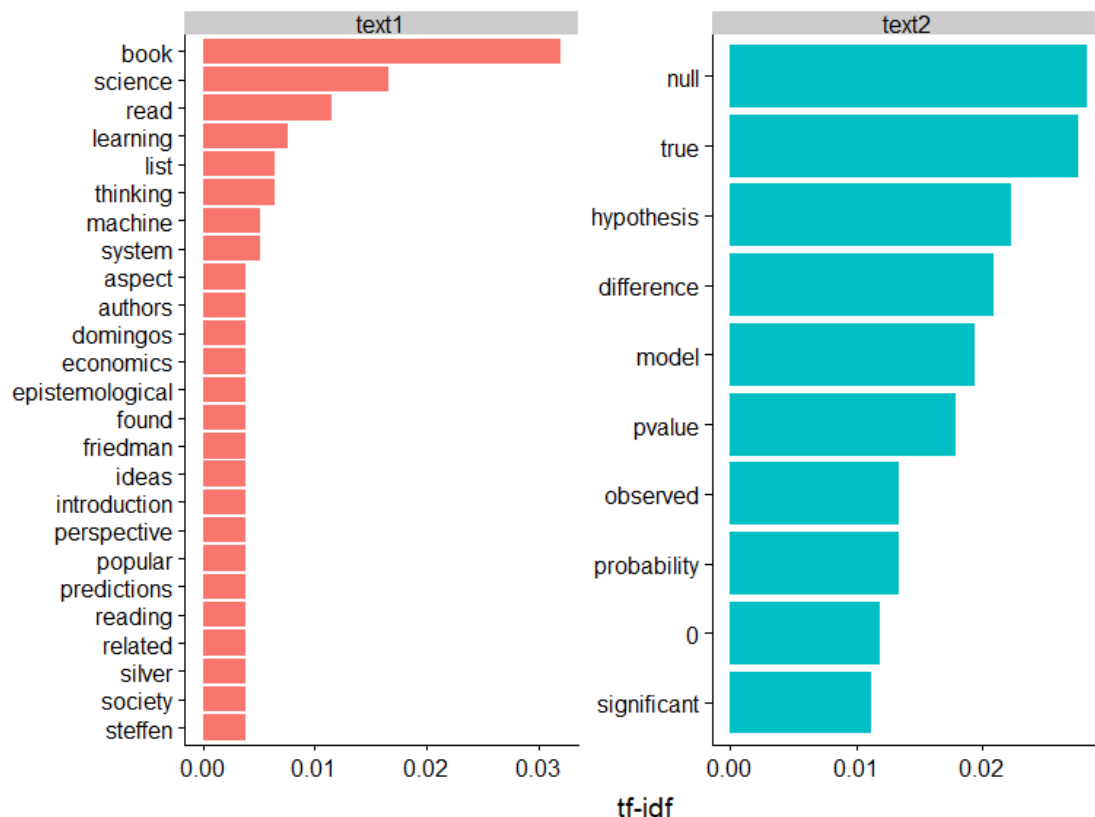
The wordcloud shows that “book”, “science”, “data”, “read” are the most frequent in text1; “true”, “hypothesis”, “pvalue”, “model”, “data”, “effect”. “probability” are the most frequent in text2.

Word and document frequency



Term Frequency Distribution

The plot shows term frequency of top 10 frequent words of each text.



how important a word is to a document in 2 documents

The plot shows term frequency based on tf_idf of top 10 frequent words of both text. We can see that the result is almost same with the result of term frequency based on simple term frequency(tf).

Relationships between words: n-grams and correlations

tokenize into consecutive sequences of words(token = “ngrams”)

Counting and filtering n-grams

```
## # A tibble: 6 x 2
##   bigram          n
##   <chr>          <int>
## 1 data science      9
## 2 machine learning  3
## 3 choice architecture 2
## 4 data analysts     2
## 5 enjoyed reading   2
## 6 learning techniques 2

## # A tibble: 6 x 2
##   bigram          n
```

```
##   <chr>                <int>
## 1 null hypothesis      21
## 2 null model          15
## 3 alternative model    7
## 4 omniscient jones     6
## 5 sample size         6
## 6 significant result   6
```

We can see that “data science” and “null hypothesis” are the most common pairs in the two books separately, which are close to the topic of books.

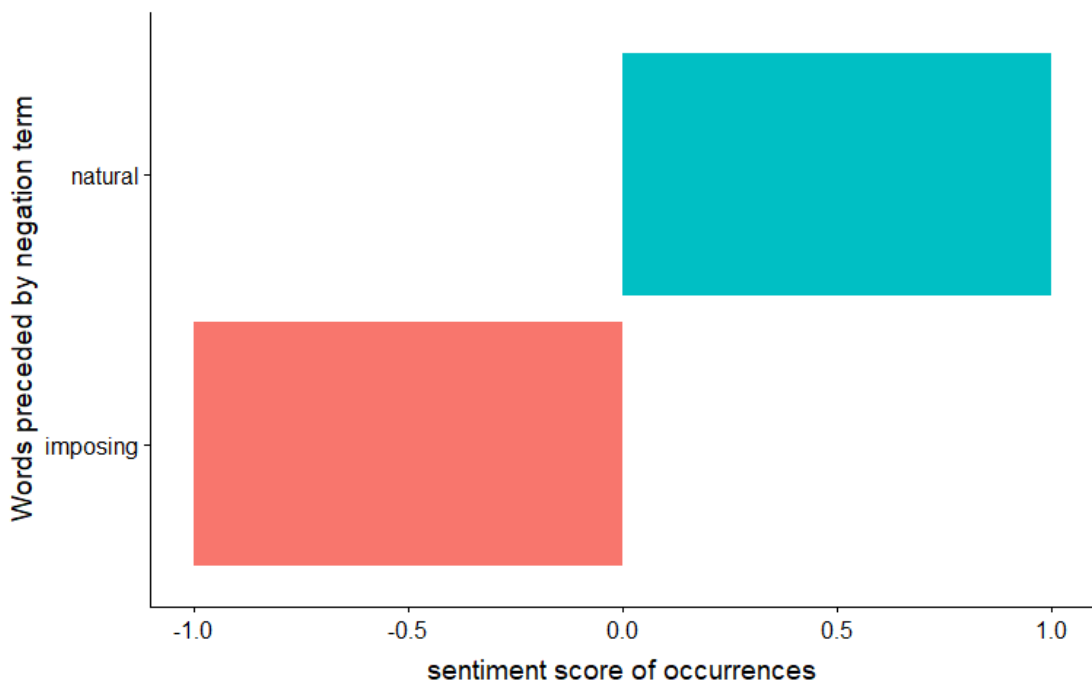
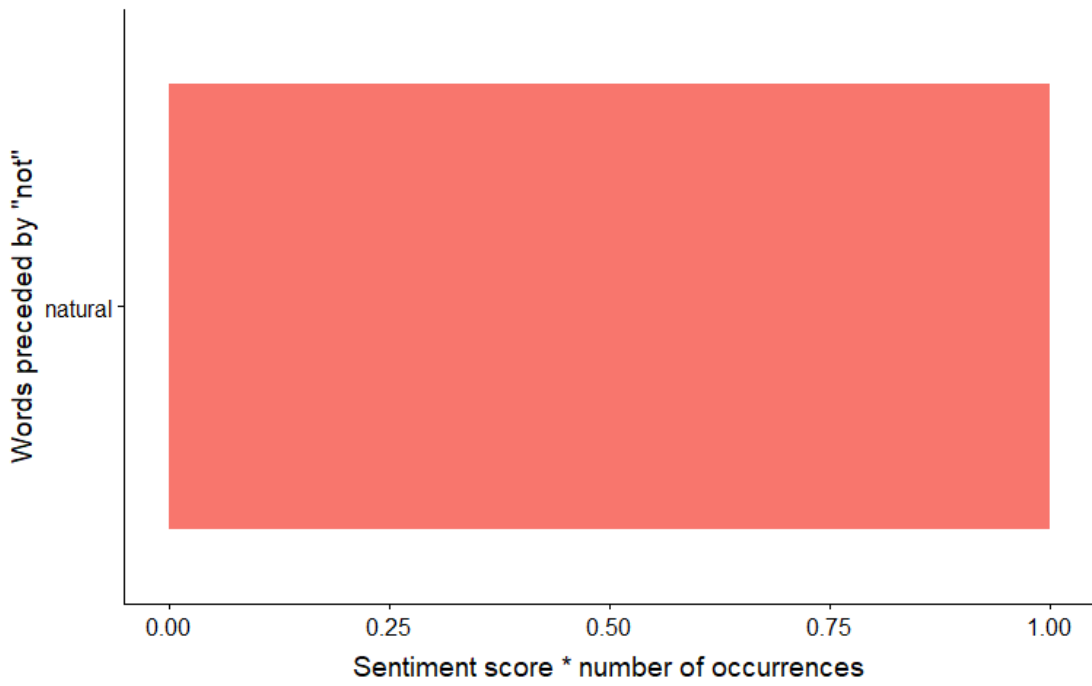
Analyzing bigrams

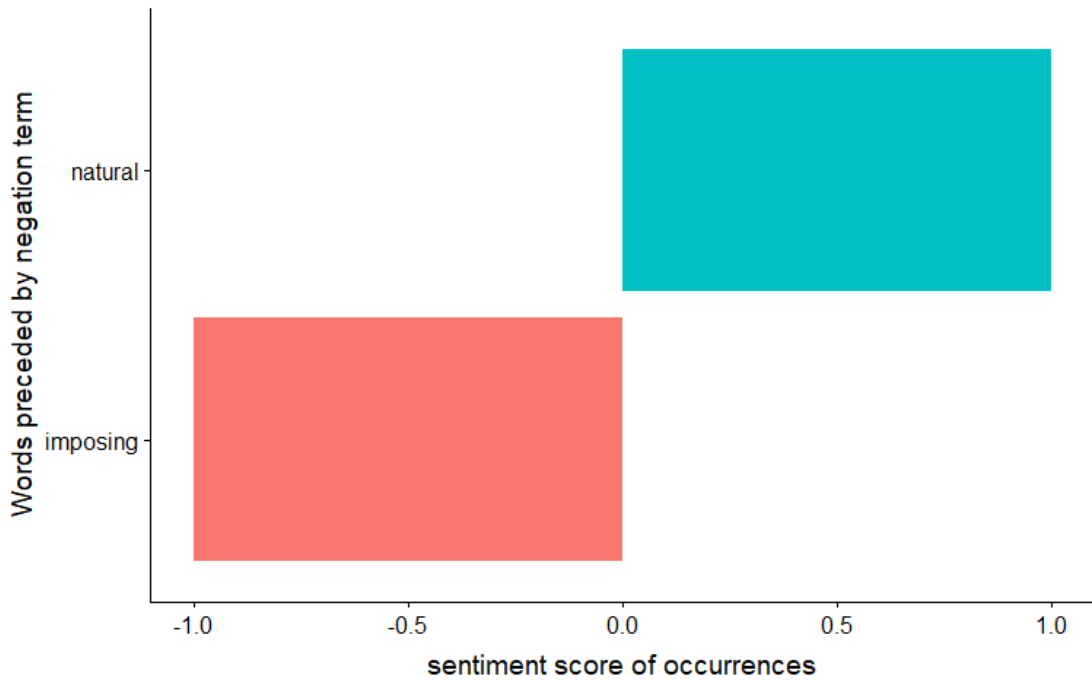
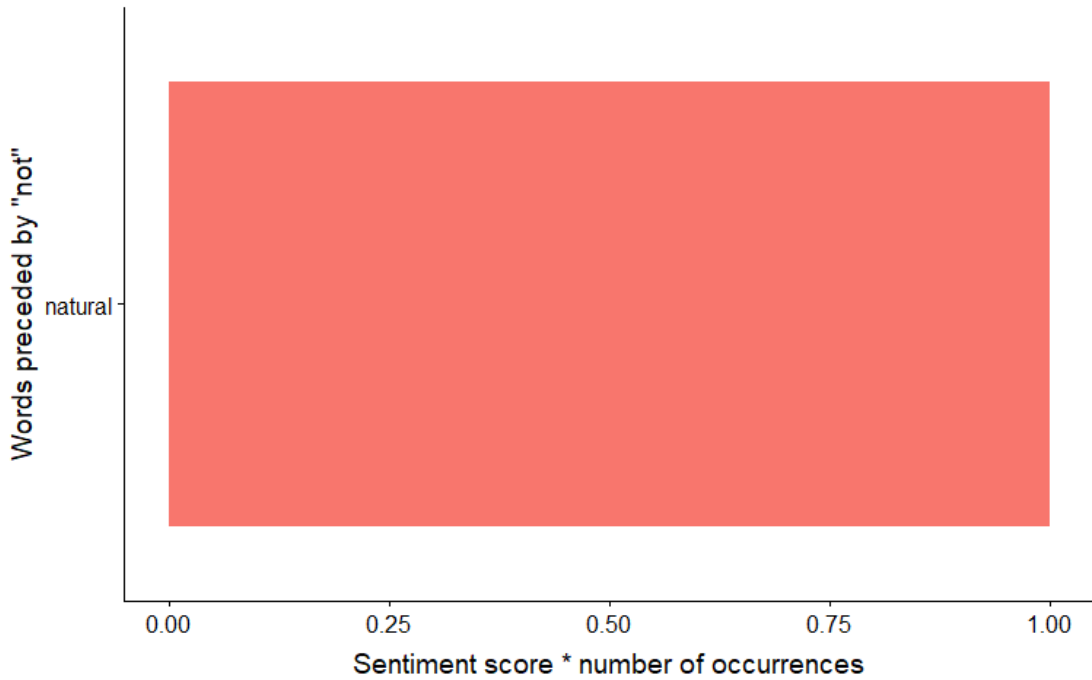
```
## # A tibble: 3 x 3
##   word1  word2      n
##   <chr>  <chr>  <int>
## 1 data   science    9
## 2 popular science    2
## 3 social science    1

## # A tibble: 3 x 3
##   word1  word2  nn
##   <chr>  <chr> <int>
## 1 data   science    1
## 2 popular science    1
## 3 social science    1
```

We can see that for both two books, the word1(s) are the same corresponding to word2 “science”, but with different frequencies.

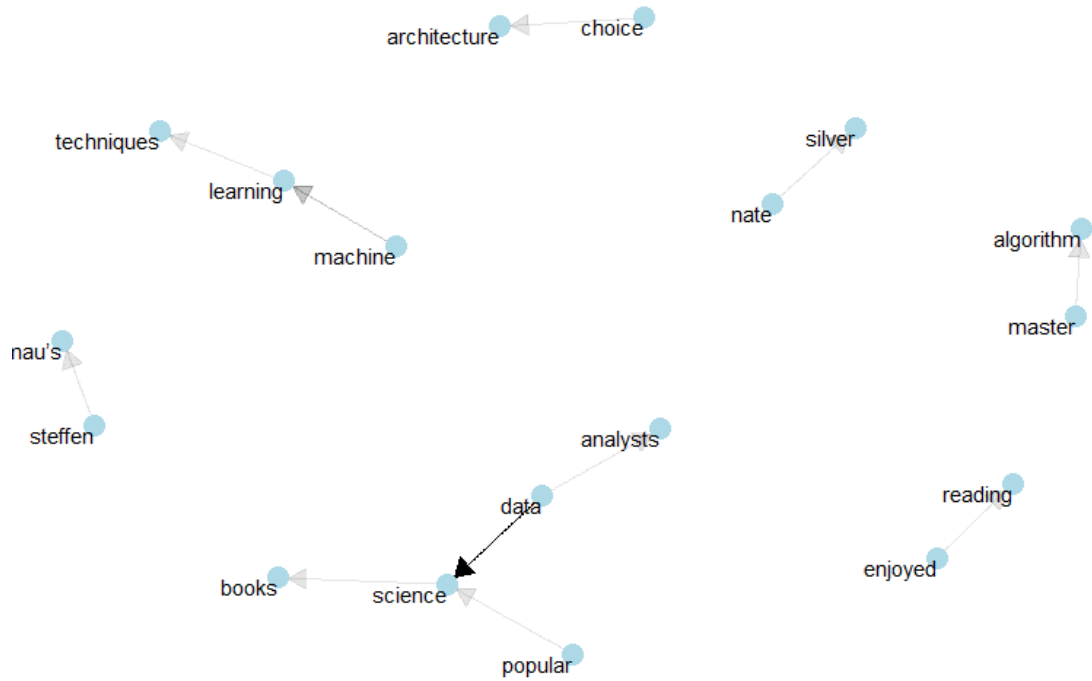
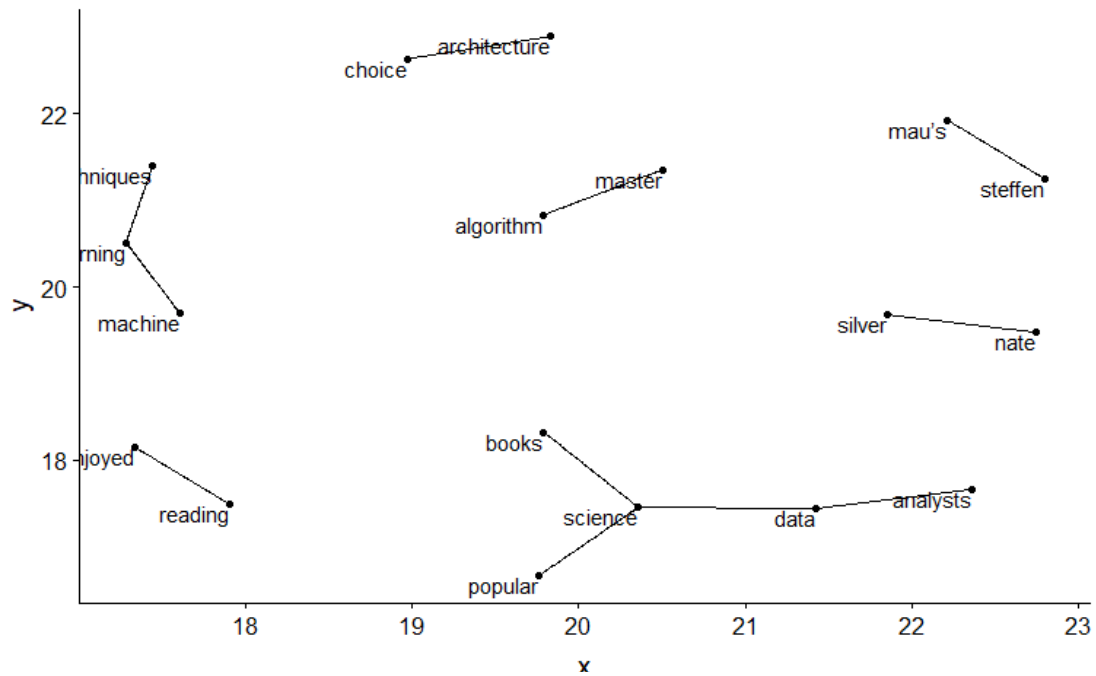
Using bigrams to provide context in sentiment analysis





We can see the sentiment scores of occurrences of Words preceded by negation term are the same in the two books. Maybe because the two texts are both downloaded from the same blog.

Visualizing a network of bigrams with ggraph




```
## 8 data true 1
## 9 data goal 1
## 10 data blog 1
## # ... with 292 more rows
```

We can easily find the words that occur with “data” in the two texts

5 Latent Dirichlet allocation

```
## A LDA_VEM topic model with 2 topics.
```

```
## # A tibble: 1,264 x 3
##   topic term      beta
##   <int> <chr>    <dbl>
## 1     1 1 1      0.00653
## 2     2 2 1      0.00976
## 3     3 1 2      0.00343
## 4     4 2 2      0.00198
## 5     5 1 3      0.00129
## 6     6 2 3      0.00142
## 7     7 1 accelerations 0.000560
## 8     8 2 accelerations 0.000798
## 9     9 1 account      0.000928
## 10    2 account      0.000425
## # ... with 1,254 more rows
```

The model computes the probability of each term generated by each topic. For example, the term accelerations has $5.598958e-04$ probability of being generated from topic 1, but a $7.975985e-04$ probability of being generated from topic 2.

Word-topic probabilities



We use `top_n`

function to find the 10 most common terms for each topic. The most common words in topic 1 include “probability”, “model”, “pvalue” and so on, which shows that it maybe related to statistics. Those most common in topic 2 include “data”, “null”, and “significant”, also suggesting that this topic is related to statistics.

```
## # A tibble: 364 x 4
##   term      topic1  topic2 log_ratio
##   <chr>      <dbl>    <dbl>    <dbl>
## 1 0      0.00457  0.0172     1.91
## 2 0.01    0.00117  0.00154     0.398
## 3 0.05    0.00340  0.000648    -2.39
## 4 0.35    0.00112  0.00160     0.518
## 5 0.5     0.00714  0.00232    -1.62
## 6 05      0.000652  0.00207     1.67
## 7 1       0.00653  0.00976     0.581
## 8 1.5     0.000228  0.00113     2.32
## 9 100     0.00245  0.00161    -0.603
## 10 150    0.0000746 0.00129     4.11
## # ... with 354 more rows
```

We can see that the words more common in topic 1 include bayesian, which seems more theoretical. Topic 2 has more word like error, errors, centered, key, interpretation seems much more practical.

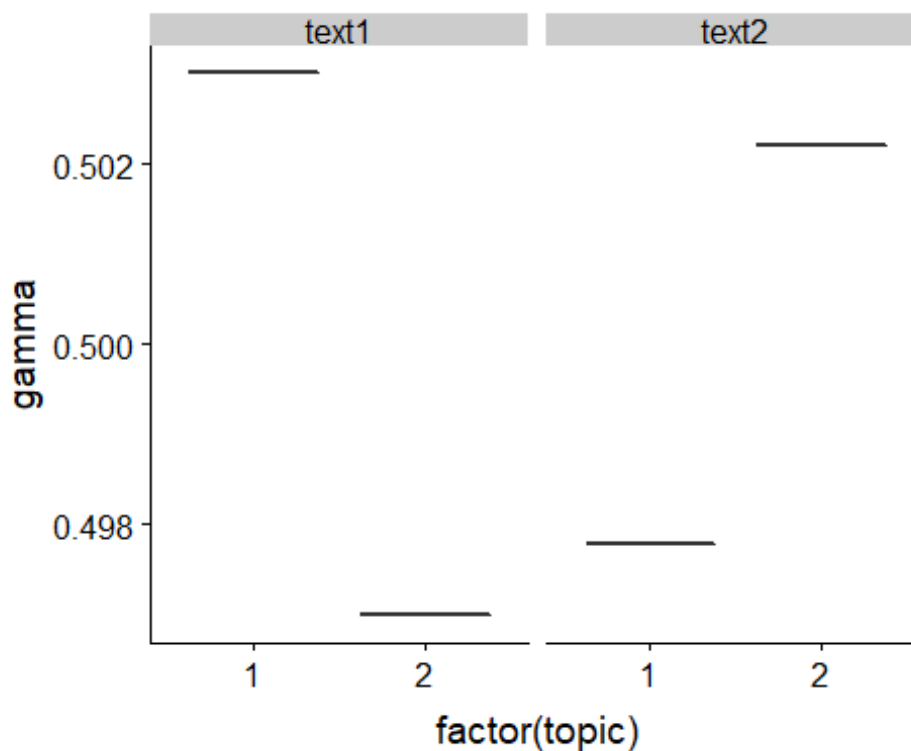
Document-topic probabilities

```
## # A tibble: 4 x 3
##   document topic gamma
##   <chr>    <int> <dbl>
## 1 text1      1 0.503
## 2 text2      1 0.498
## 3 text1      2 0.497
## 4 text2      2 0.502

## # A tibble: 4 x 4
##   title chapter topic gamma
##   <chr> <lgl>    <int> <dbl>
## 1 text1 NA      1 0.503
## 2 text2 NA      1 0.498
## 3 text1 NA      2 0.497
## 4 text2 NA      2 0.502
```

We can see the proportion of words from that document that are generated from that topic. For example, the model estimates that nearly more than half of the words in document 1 were generated from topic 1. Nearly more than half of the words in document 2 were generated from topic 2 we how topics are associated with each document. However, these articles does not have chapters so that their values are NA.

Per-document classification



```
## # A tibble: 2 x 4
##   title chapter topic gamma
##   <chr> <lgl>   <int> <dbl>
## 1 text1 NA         1 0.503
## 2 text2 NA         2 0.502

## # A tibble: 0 x 5
## # ... with 5 variables: title <chr>, chapter <lgl>, topic <int>,
## #   gamma <dbl>, consensus <chr>

## # A tibble: 2 x 4
##   title chapter topic gamma
##   <chr> <lgl>   <int> <dbl>
## 1 text1 NA         1 0.503
## 2 text2 NA         2 0.502
```

We visualize the gamma probabilities by using box plot. We notice that these two articles were uniquely identified as a single topic each.

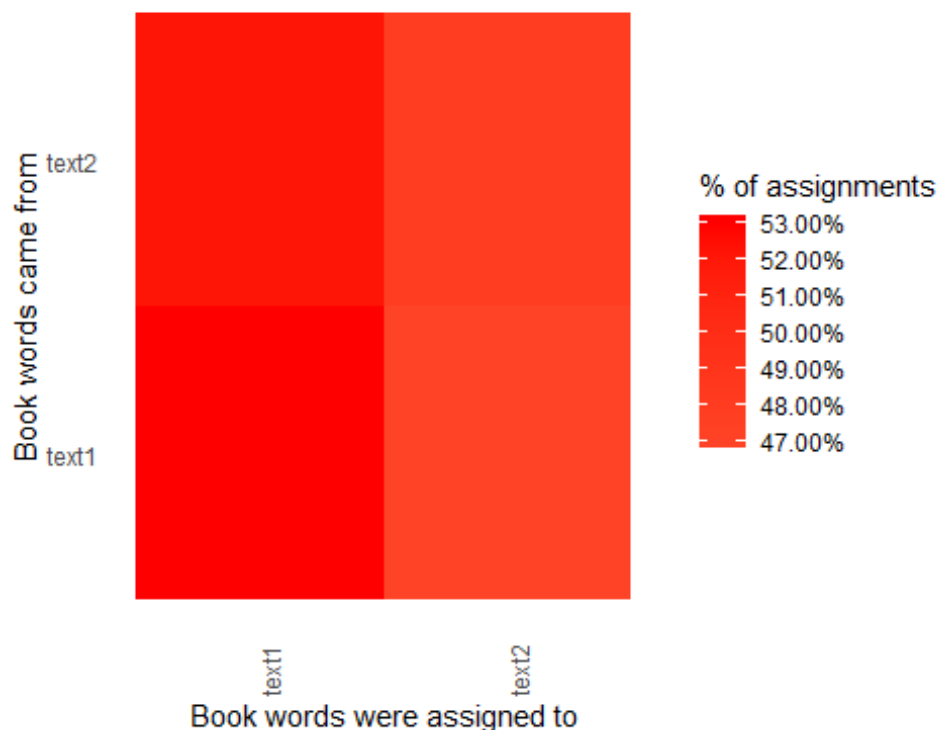
By word assignments: augment

```
## # A tibble: 655 x 4
##   document term          count .topic
##   <chr>   <chr>        <dbl> <dbl>
## 1 text1    1             2      2
## 2 text2    1          10      2
## 3 text1    2             2      1
## 4 text2    2             2      1
## 5 text1    3             1      2
## 6 text2    3             1      2
## 7 text1 accelerations 1      2
## 8 text1 account      1      1
## 9 text1 actual       1      1
## 10 text1 age          2      2
## # ... with 645 more rows

## # A tibble: 655 x 6
##   title chapter term          count .topic consensus
##   <chr> <lgl>   <chr>        <dbl> <dbl> <chr>
## 1 text1 NA      1             2      2 text2
## 2 text2 NA      1          10      2 text2
## 3 text1 NA      2             2      1 text1
## 4 text2 NA      2             2      1 text1
## 5 text1 NA      3             1      2 text2
## 6 text2 NA      3             1      2 text2
## 7 text1 NA      accelerations 1      2 text2
## 8 text1 NA      account      1      1 text1
## 9 text1 NA      actual       1      1 text1
## 10 text1 NA      age          2      2 text2
## # ... with 645 more rows
```

This script returns a tidy data frame for the counts of words. combine this assignments table with the consensus book titles so that we can find incorrect classified words. We can see in the list, the incoorrect classified words have been counted.

visualize a confusion matrix



By visualizing a confusion matrix, we can find how often words from one book were assigned to another. We can find that words are slightly more often assigned to text1.

Find out and count mistaken words

```
## # A tibble: 327 x 6
##   title chapter term      count .topic consensus
##   <chr> <lg1>  <chr>      <dbl> <dbl> <chr>
## 1 text1 NA      1          2      2 text2
## 2 text2 NA      2          2      1 text1
## 3 text1 NA      3          1      2 text2
## 4 text1 NA      accelerations 1          2 text2
## 5 text1 NA      age          2          2 text2
## 6 text1 NA      agnostic     1          2 text2
## 7 text1 NA      algorithm    2          2 text2
## 8 text1 NA      analysts    2          2 text2
## 9 text1 NA      architecture 2          2 text2
## 10 text1 NA      aren't      1          2 text2
## # ... with 317 more rows
```

```
## # A tibble: 327 x 4
##   title consensus term          n
##   <chr> <chr>      <chr>      <dbl>
## 1 text2 text1     null         38
## 2 text2 text1     true          37
## 3 text2 text1     difference      28
## 4 text2 text1     pvalue         24
## 5 text2 text1     data           19
## 6 text2 text1     probability     18
## 7 text1 text2     science         13
## 8 text2 text1     alternative      13
## 9 text2 text1     effect          13
## 10 text2 text1     extreme          9
## # ... with 317 more rows
```

We find the commonly mistaken words, such as null, true, difference and pvalue which are more than 20.