

Assignment 10 by Team 3

Ashutosh Agarwal, Shun-Lung Chang, Pooja Natu

This study was conducted in R. The source code can be found [here](#).

1. Convert the html to text files and separate the individual news items. The individual press release items serve as documents.

```
colnames(text_df)

[1] "id" "text" "year"
text_df[2]

# A tibble: 638 x 1
                                text
                                <chr>
1 Wissenschaft jenseits von Science Fiction: Jacobs University beteiligt sich
2 Sozialer Mehrwert durch Musik? Begleitstudie der Jacobs University zur Symb
3 Leibniz-Preis für Jacobs-Professorin Antje Boetius Dec , Antje Boetius, sei
4 Neuer Förderpreis der Stiftung Mercator für Studierende der Jacobs Universi
5 Management mit Zukunft: TiasNimbas Business School und Jacobs University st
6 Deutscher Hochschulverband ernennt Katja Windt zur »Hochschullehrerin des J
7 Der persönliche Eindruck zählt: Studienberater aus vier Kontinenten informi
8 Spintronik: Physikerteam gelingt Nachweis eines nano-mechanischen Torsionse
9 „Neue malerische Wendungen“: University Club der Jacobs University zeigt ab
10 RWE startet CO-Konversions-Pilotanlage auf Basis einer von der Jacobs Unive
# ... with 628 more rows
```

2. Remove stop words and perform stemming.

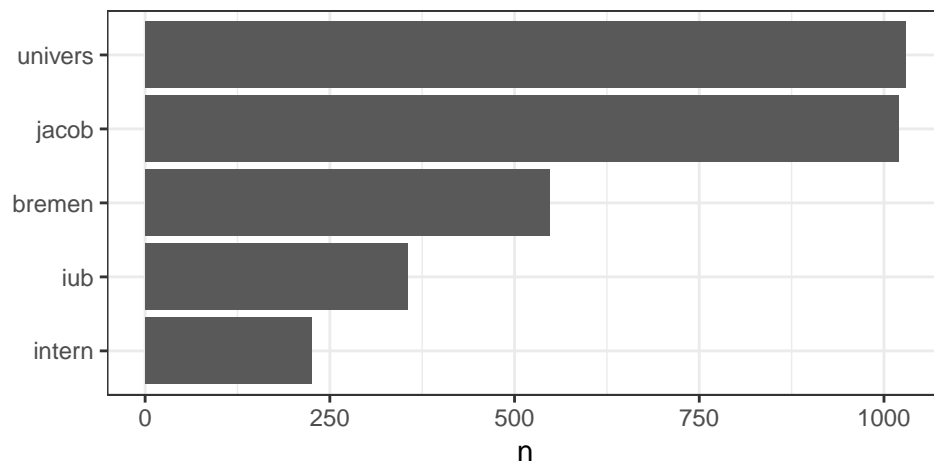
```
t <- text_df %>%
  unnest_tokens(word, text) %>%
  anti_join(tibble(word = c(stopwords("de"), stopwords("en")))) %>%
  mutate(stemmed_word = wordStem(word))
```

Joining, by = "word"

id	year	word	stemmed__word
1	2008	wissenschaft	wissenschaft
1	2008	jenseits	jenseit
1	2008	science	scienc
1	2008	fiction	fiction
1	2008	jacobs	jacob
1	2008	university	univers

3. Perform a frequency analysis to compute the term-document (TD) matrix. What are the most common terms?

```
top_5_words <- t %>%
  group_by(stemmed_word) %>%
  count(sort = TRUE) %>%
  ungroup() %>%
  slice(1:5)
```



```
word_counts <- t %>%
  group_by(id, stemmed_word) %>%
  count() %>%
  arrange(id, -n) %>%
  ungroup()

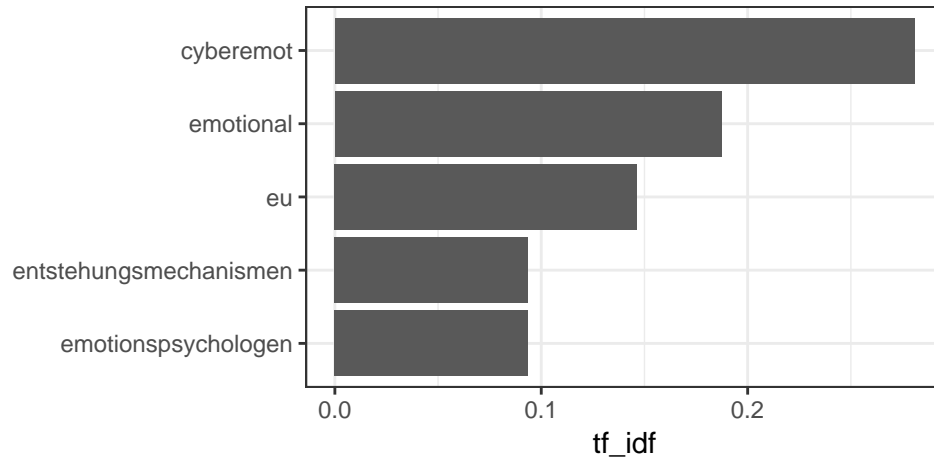
td <- word_counts %>% spread(stemmed_word, n, fill = 0) %>%
  select(-id) %>%
  as.matrix()
```

univers	jacob	bremen	iub	intern
3	3	0	0	0
2	2	5	0	0
1	2	2	0	0
1	2	0	0	1
2	2	2	0	0
1	1	1	0	0
2	3	1	0	0
2	1	1	0	0
4	3	1	0	0
3	3	1	0	0

4. Compute inverse-document frequency (IDF) and term importance (TI). What are now the most common terms?

```
tf_idf <- word_counts %>%
  bind_tf_idf(term = stemmed_word, document = id, n = n)
```

id	stemmed_word	n	tf	idf	tf_idf
1	cyberemot	3	0.0434783	6.4583383	0.2807973
1	eu	3	0.0434783	3.3672958	0.1464042
1	jacob	3	0.0434783	0.3490907	0.0151779
1	univers	3	0.0434783	0.1022306	0.0044448
1	emotional	2	0.0289855	6.4583383	0.1871982
1	projekt	2	0.0289855	2.4693542	0.0715755



5. Compute pairwise cosine and Euclidean distance between all documents.

```
cos_dist <- dist2(td, method = 'cosine')
euc_dist <- dist2(td, method = 'euclidean')
```

```
cos_dist[1:3, 1:3] %>% kable()
```

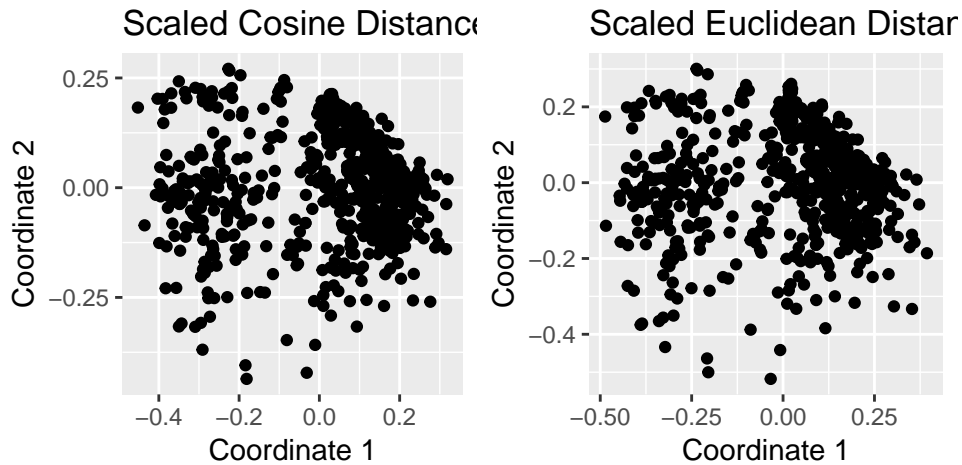
1	2	3
0.0000000	0.8578515	0.8455115
0.8578515	0.0000000	0.7659177
0.8455115	0.7659177	0.0000000

```
euc_dist[1:3, 1:3] %>% kable()
```

0.000000	1.309848	1.300393
1.309848	0.000000	1.237673
1.300393	1.237673	0.000000

6. Apply a multi-dimensional scaling approach to the distance matrix and render a 2D scatterplot. Compare the two distance metrics.

```
cos_dist_fit <- cmdscale(cos_dist, k = 2)
euc_dist_fit <- cmdscale(euc_dist, k = 2)
```



7. Capture the year of release during parsing and color code the scatterplot by time. Produce a Word Cloud for each year.

```
create_wordcloud <- function(year) {
  d <- t %>%
    filter(year == year) %>%
    group_by(stemmed_word) %>%
    count() %>%
    ungroup()

  wordcloud(words = d$stemmed_word, freq = d$n,
    max.words = 50, colors = brewer.pal(8, "Dark2"))
  text(x = 0.5, y = 1, cex = 0.5, as.character(year))
}

for (i in 1999:2015) {
  create_wordcloud(i)
}
```



