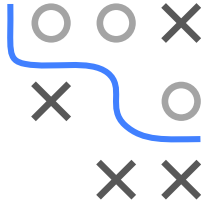
Algorithms and Data Structures

Matrix Approximation
Topic Extraction

Learning goals

- Topic extraction

APPLICATION: TOPIC EXTRACTION



Often online articles refer to articles with similar content, e.g.

Building AI that Can Build AI

"related"

In the Future, Warehouse Robots Will Learn on Their Own

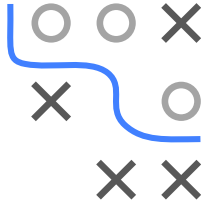
"related"

As Amazon Pushes Forward With Robots, Workers Find New Roles

The first two articles should definitely have one topic in common, just like the last two articles. We want to extract these two topics using a NMF.

APPLICATION: TOPIC EXTRACTION / 2

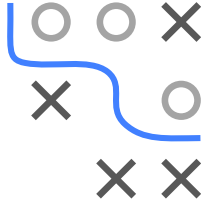
We set up the corresponding document-term matrix.



```
##                doc1 doc2 doc3
## accelerate        1    0    0
## accelerating      1    0    0
## accurately        1    0    0
## across            1    1    2
## address           1    1    0
## adjust            1    0    0
```

APPLICATION: TOPIC EXTRACTION / 3

We "search" two topics linking the articles.

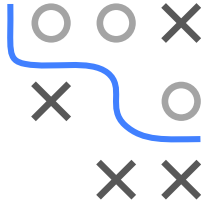


```
set.seed(1)
res = nmf(tdm, 2, "Frobenius")
```

```
##                topic1   topic2
## accelerate     0.0016   3.8e-11
## accelerating   0.0016   3.8e-11
## accurately     0.0016   3.8e-11
## across         0.0023   4.1e-03
## address        0.0026   6.3e-05
## adjust         0.0016   3.8e-11
```

TOPIC EXTRACTION

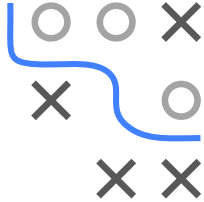
For both topics, we print the 30 words with the largest values in the columns of matrix **W**. The size of the word in the wordcloud is determined by the value of w_{ij} (placement of the word is completely random).



(a) Topic 1

(b) Topic 2

TOPIC EXTRACTION: COEFFICIENT MATRIX H



```
H
```

```
##          topic 1   topic 2
## doc1     4.5e+02   1.8e-09
## doc2     2.9e+02   5.8e+01
## doc3     1.6e-09   4.9e+02
```

The coefficient matrix shows: The first article clearly refers to the first extracted topic, article 3 clearly to the last. Article 2 addresses both topics.

Implementation in R: `https://rpubs.com/JanpuHou/300168`