tm4ss

Hands-on: a five day text mining course for humanists and social scientists in R

Gregor Wiedemann | Andreas Niekler

Natural Language Processing Group
University of Leipzig
gregor.wiedemann@uni-leipzig.de
aniekler@informatik.uni-leipzig.de

September 12, 2017

Outline

Motivation and background

Structure

Contents
- Data and resources
- Tutorials

Teaching experience

Adaptations, conclusion and future work

Overview

Motivation and background

Structure

Contents
- Data and resources
- Tutorials

Teaching experience

Adaptations, conclusion and future work

Motivation and background I

- **Large digital text collections** → primary source of data for **empiric analyses**.
- Text mining:
  - **statistical** and **computer-linguistic** methods
  - **(semi-)automatically extract semantic structures** from very large amounts of texts
  - major innovation in various disciplines (political science, economics, history...) (Lemke and Wiedemann 2016)
- **Gesis** idea 2014: **text mining course targeted to humanists and social scientists**
- **Major issue** for such a course: the famous debate of **'more hack' versus 'less yack'**
- Protagonists of DH more engagement in actual analysis by getting hands on data (Nowviskie 2014)

Motivation and background II

- **focus on the coding approach**: To fulfill DH/CSS needs + acknowledgement of ‘hack vs. yack’.
- Teaching **basics of coding** in a simple and coherent scripting environment allows scholars to create **individual solutions** tailored to their data formats and specific analysis requirements.
- Especially in **social science**, many students and **scholars already have had contact with statistical analysis** software such as SPSS, STATA or R.

Overview

Motivation and background

Structure

Contents
- Data and resources
- Tutorials

Teaching experience

Adaptations, conclusion and future work

Structure I

- The course is a **five day, full-time** workshop where students are present in class.
- Teachers (ideally): **computer science background and social science background**
- The didactic concept relies on 3 major pillars:
  1. **8 Lectures on text mining** and its applications in DH projects (30 % of course time)
  2. **8 Tutorials** on writing and discussing text mining scripts in R (50 % of course time)
  3. Presentation and discussion of **user projects** (20 % of course time)

Structure II

- Lectures contain
  1. Theoretical and methodological foundations of text mining
  2. Example studies from DH contexts
  3. Data acquisition (import, web scraping)
  4. Text preprocessing
  5. Lexicometric analysis
  6. Unsupervised machine learning
  7. Supervised machine learning and
  8. Integration with conventional text analysis methodologies.
- Tutorial sessions are the didactic core of the course.
  - E-Learning platform (ILIAS Core Team 2017),
  - Statistical programming language **R and the IDE R-Studio**

Technical Infrastructure I

- R (R Core Team 2016): programming language for statistical analysis.
- R-Studio (RStudio Team 2015): is a user-friendly (IDE) for R.
- Swirl (Kross et al. 2017): is an R package to learn R, in R.
- Packages for text analysis:
  - *tm* package (Feinerer, Hornik, and Meyer 2008).
  - *rvest* (Wickham 2016)
  - *readtext* (Benoit and Obeng 2017)
  - *openNLP* (Hornik 2016)
  - *topicmodels* (Grün and Hornik 2011)
  - *LiblineaR* (Helleputte 2017)
- Packages for visualization:
  - *wordcloud* (Fellows 2014)
  - *ggplot2* (Wickham 2009)
  - *igraph* (Csardi and Nepusz 2006)

Technical Infrastructure II

- knitr (Xie 2014)

```
38  **How many speeches do we have per president?** This can easily be counted
    with the command `table`, which can be used to create a cross table of
    different values. If we apply it to a column, e.g. *president* of our data
    frame, we get the counts of the unique *president* values.
39
40  ```{r eval=T, echo=T}
41  table(textdata[, "president"])
42  ```
```

How many speeches do we have per president? This can easily be counted with the command `table`, which can be used to create a cross table of different values. If we apply it to a column, e.g. *president* of our data frame, we get the counts of the unique *president* values.

```
table(textdata[, "president"])
```

```
##
##       Abraham Lincoln         Andrew Jackson         Andrew Johnson
##                     4                      8                      4
##          Barack Obama      Benjamin Harrison        Calvin Coolidge
##                     8                      4                      6
##     Chester A. Arthur        Donald J. Trump   Dwight D. Eisenhower
##                     4                      1                      9
## Franklin D. Roosevelt        Franklin Pierce      George H.W. Bush
##                    12                      4                      4
```

Overview

Motivation and background

Structure

Contents
- Data and resources
- Tutorials

Teaching experience

Adaptations, conclusion and future work

Contents

- **Single text mining applications**
- **Combination of several applications** to complex analysis workflows
- **Same data source** for each single tutorial
- **Simple to complex** applications
- Students are **writing and running the scripts** on their **own machines***

* Only minor problems due to different OS: encoding, Java versions

Data and resources

- "State of the Union" addresses (SOTU) of the 45 presidents of the United States published between 1790 and 2017.
  - 231 documents, containing roughly 28,000 types and 1,400,000 tokens
  - The size is **large enough** for statistical analysis, but not too large.
  - Preprocessing steps or text mining applications **do not take too much time** during tutorials.
- Sentence segmentation and POS-tagging: **openNLP and publicly available pre-trained models** (Morton et al. 2005).
- Reference corpora for key-term extraction: **Leipzig Corpora Collection** (Quasthoff, Goldhahn, and Eckart 2014).

Tutorials I

- We provide **printed and digital versions** of tutorial sheets and an R project skeleton.
- During half time and at the end of each tutorial session, **parts of script are explained** by an instructor.
- For fast learners or students with R experience, each tutorial sheet provides **optional exercises**.

Tutorials II

Intro | Tutorial 1 | Tutorial 2 | Tutorial 3 | Tutorial 4 | Tutorial 5 | Tutorial 6 | Tutorial 7 | Tutorial 8

1 Text preprocessing
2 Time series
3 Grouping of sentiments
4 Heatmaps
5 Optional exercises
References

Tutorial 3: Frequency analysis

Andreas Niekler, Gregor Wiedemann

2017-09-11

This tutorial introduces frequency analysis with basic R functions. We further introduce some text preprocessing functionality provided by the R package.

1. Text preprocessing
2. Time series
3. Grouping of semantic categories
4. Heatmaps

```
options(stringsAsFactors = FALSE)
require(tm)
```

1 Text preprocessing

Like in the previous tutorial we read the CSV data file containing the State of the union addresses. This time, we add two more columns for the year and the decade. For the year we select a sub string of the four first characters from the `date` column of the data frame (e.g. extracting "1990" from "1990-02-12"). For the decade we select a sub string of the first three characters and paste a 0 to it. In later parts of the exercise we can use these columns for grouping data.

```
textdata <- read.csv("data/sotu.csv", sep = ";", encoding = "UTF-8")

# we add some more metadata columns to the data frame
textdata$year <- substr(textdata$date, 0, 4)
textdata$decade <- paste0(substr(textdata$date, 0, 3), "0")
```

Then, we create a corpus object again. For metadata we can add a DateTimeStamp to our table mapping of metadata and data.frame fields. Moreover, we apply different preprocessing steps to the corpus text. `removePunctuation` leaves only alphanumeric characters in the text. `removeNumbers` removes numeric characters. Then lowercase transformation is performed and an English set of stop-words is removed.

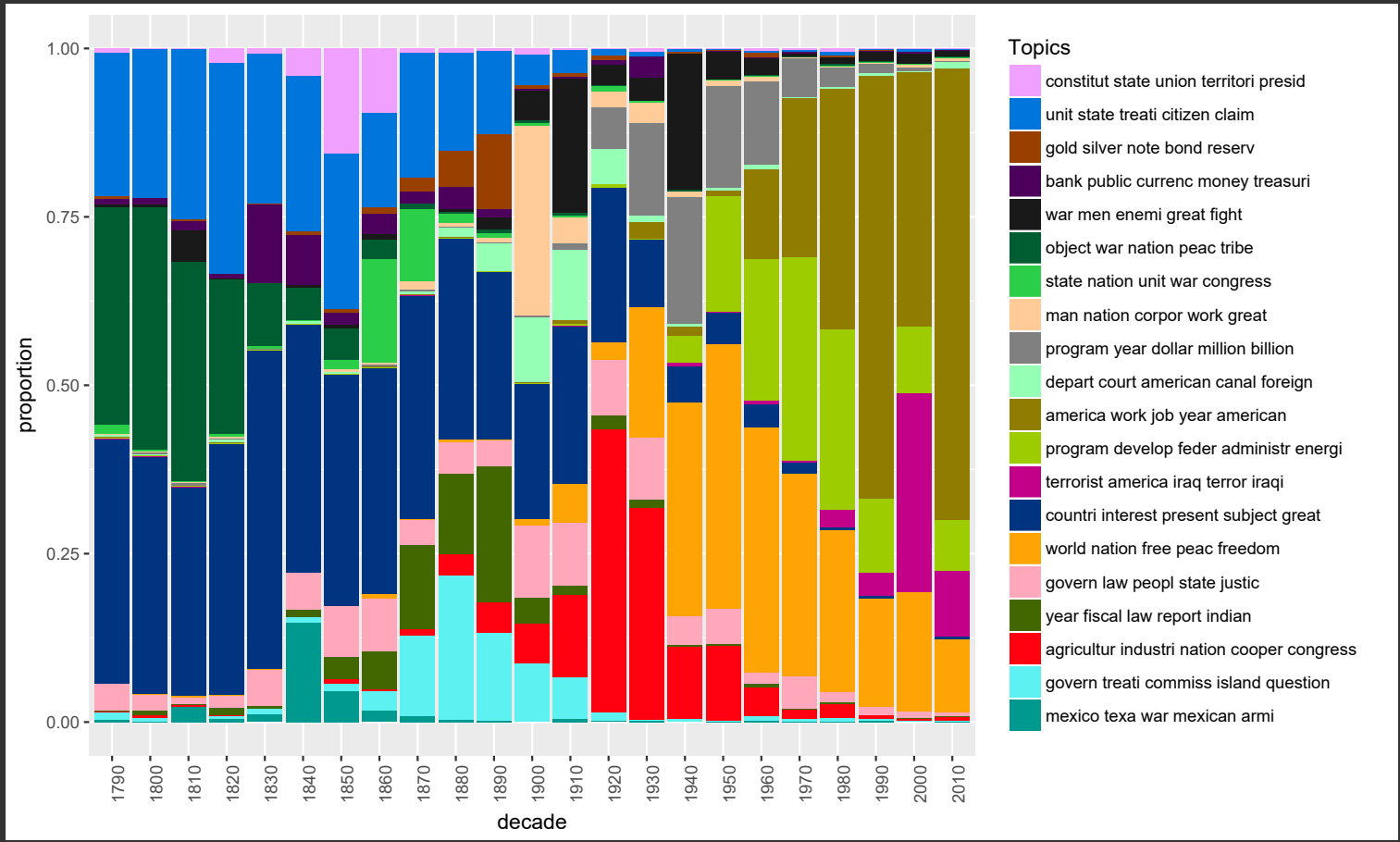
```
m <- list(ID = "id", content = "text", DateTimeStamp = "date")
myReader <- readTabular(mapping = m)
corpus <- Corpus(DataframeSource(textdata), readerControl = list(reader = myReader))
corpus <- tm_map(corpus, removePunctuation, preserve_intra_word_dashes = TRUE)
corpus <- tm_map(corpus, removeNumbers)
corpus <- tm_map(corpus, content_transformer(tolower))
corpus <- tm_map(corpus, removeWords, stopwords("en"))
corpus <- tm_map(corpus, stripWhitespace)
```

Tutorials III

We cover a wide range of text mining techniques popular throughout DH and CSS.

- Data acquisition
- Lexicometric
  - Text processing
  - Frequency analysis
  - Key term extraction
  - Co-occurrence analysis
- Machine Learning.
  - Unsupervised machine learning (Topic Models)
  - Supervised machine learning
  - Advanced preprocessing

Tutorials IV



Overview

Motivation and background

Structure

Contents
- Data and resources
- Tutorials

Teaching experience

Adaptations, conclusion and future work

Motivation and background II

- The course was **taught five times** reaching an audience up to 30 scholars per course, among others **political scientists, sociologists, economists, historians and philologists**.
- Course evaluation 2016 (N = 21)

Survey question / scale	**1**	**2**	**3**	**4**	**5**
The course is well structured.*	-	4.7	-	38.1	57.1
The knowledge transfer between theory and practice works well.*	-	4.7	9.5	28.6	57.1
I feel enabled to approach my own text mining analysis.*	4.7	19.1	33.3	23.8	19.1
The course materials were useful.*	-	-	-	23.8	76.2
I have learned a lot in the course.*	-	-	4.7	47.6	47.6
How do you assess the quantity of the course contents?**	-	-	38.1	47.6	14.3
How do you assess the amount of time for discussion?**	-	9.5	90.5	-	-
How do you assess the amount of time for practical work?**	4.7	28.6	66.7	-	-

* scale: strongly disagree (1), rather disagree (2), neither/nor (3), rather agree (4), strongly agree (5)
** scale: way too low (1), rather too low (2), just right (3), rather too much (4), way too much (5)

Overview

Motivation and background

Structure

Contents
- Data and resources
- Tutorials

Teaching experience

Adaptations, conclusion and future work

Adaptations and future work

- Highly skilled and motivated target audience consisting of scholars mostly at the Ph.D. or post-doc level.
- For **other target audiences**, course contents could be **reduced or requirement levels could be lowered**.
- R + knitr: Ideal combination for teaching in DH.
- Alternating sessions of lectures and tutorials **can be held in weekly manner** (Semester course).
  - By requesting students to hand in **papers as HTML files rendered from Rmarkdown** scripts, teachers are able to fully **reproduce the student's work**.
  - Student papers could be published to **provide alternative solutions to the class**.

Conclusion

- Published under GPLv3: `https://tm4ss.github.io`
- **Open source textbook** for self-learners with an **extended theoretical introduction** to the course **is planned**.
- Conclusion:
  - R programming language as a **flexible and easy to learn environment** for many complex text analysis tasks.
  - R + knitr to **create tutorial sheets for gaining practical experience**
  - better more than less time for hands-on sessions
  - **public course material** for self-learners and alternative teaching formats

References

Benoit, Kenneth and Adam Obeng (2017). *readtext: Import and Handling for Plain and Formatted Text Files*. URL: https://CRAN.R-project.org/package=readtext.

Csardi, Gabor and Tamas Nepusz (2006). "The igraph software package for complex network research". In: *InterJournal* Complex Systems, p. 1695. URL: http://igraph.org.

Feinerer, Ingo, Kurt Hornik, and David Meyer (2008). "Text mining infrastructure in R". In: *Journal of Statistical Software* 25.5, pp. 1–54. URL: http://www.jstatsoft.org/v25/i05.

Fellows, Ian (2014). *wordcloud: Word Clouds*. URL: https://CRAN.R-project.org/package=wordcloud.

Grün, Bettina and Kurt Hornik (2011). "Topicmodels: an R package for fitting topic models". In: *Journal of Statistical Software* 40.13, pp. 1–30. URL: http://www.jstatsoft.org/v40/i13/.

Helleputte, Thibault (2017). *LiblineaR: Linear Predictive Models Based on the LIBLINEAR C/C++ Library*.

Hornik, Kurt (2016). *openNLP: Apache OpenNLP Tools Interface*. URL: https://CRAN.R-project.org/package=openNLP.

ILIAS Core Team (2017). *ILIAS: Open Source e-Learning*. Köln. URL: https://www.ilias.de.

Kross, Sean et al. (2017). *swirl: Learn R, in R*. R package version 2.4.3. URL: https://CRAN.R-project.org/package=swirl.

Lemke, Matthias and Gregor Wiedemann, eds. (2016). *Text Mining in den Sozialwissenschaften: Grundlagen und Anwendungen zwischen qualitativer und quantitativer Diskursanalyse*. Wiesbaden: Springer VS.

Morton, Thomas et al. (2005). *OpenNLP: A Java-based NLP Toolkit*. URL: http://opennlp.sourceforge.net.

Nowviskie, Bethany (2014). "On the Origin of "Hack" and "Yack"". In: *Journal of Digital Humanities* 3.2. URL: http://journalofdigitalhumanities.org/3-2/on-the-origin-of-hack-and-yack-by-bethany-nowviskie/.

Quasthoff, Uwe, Dirk Goldhahn, and Thomas Eckart (2014). "Building Large Resources for Text Mining: The Leipzig Corpora Collection". In: *Text Mining: From Ontology Learning to Automated Text Processing Applications*. Ed. by Chris Biemann and Alexander Mehler. DOI: 10.1007/978-3-319-12655-5_1. Cham: Springer International Publishing, pp. 3–24. ISBN: 978-3-319-12655-5. URL: http://dx.doi.org/10.1007/978-3-319-12655-5_1.

R Core Team (2016). *R: A Language and Environment for Statistical Computing*. Vienna, Austria. URL: https://www.R-project.org/.

RStudio Team (2015). *RStudio: Integrated Development Environment for R*. Boston, MA. URL: http://www.rstudio.com/.

Wickham, Hadley (2009). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. ISBN: 978-0-387-98140-6. URL: http://ggplot2.org.

— (2016). *rvest: Easily Harvest (Scrape) Web Pages*. URL: https://CRAN.R-project.org/package=rvest.

Xie, Yihui (2014). "knitr: A Comprehensive Tool for Reproducible Research in R". In: *Implementing reproducible research*. Ed. by Victoria Stodden, Friedrich Leisch, and Roger D. Peng. Boca Raton: Taylor and Francis. ISBN: 978-1466561595.