Stefan Müller
Postdoctoral Researcher
Chair of Policy Analysis
University of Zurich
https://muellerstefan.net

Seminar 615786

Quantitative Text Analysis

Draft (last update: January 18, 2019)

Latest version: https://muellerstefan.net/teaching/2019-spring-qta.pdf

---

Term: Spring term 2019
Time: Monday, 12:15–13:45
Lecture Room: **NA**
ECTS: 6.0

Lecturer: Stefan Müller
Office: AFL H 349
Office hours: Tuesday, 16:00–17:00
E-Mail: mueller@ipz.uzh.ch

Course Content

In recent times the availability of textual data has increased massively, and there are multiple opportunities for analysing these data to answer social science research questions. This course introduces students of political science to the quantitative analysis of textual data. We cover a treatment of underlying theoretical assumptions, applications of these methods in the scholarly literature, and the respective implementations in the R statistical programming language.

Each session contains practical, hands-on exercises to apply the methods to real texts. Most of these methods can be reduced to a three-step process: first, identifying texts and units of texts for analysis; second, extract quantitatively measured features from these texts and converting them to a quantitative feature matrix; third, analyse this matrix with statistical methods, such as dictionary construction and application, scaling models, and topic models, to draw inferences about the texts. Students will learn how to apply these steps to various types of texts. There will be two homeworks which cover the theoretical assumptions as well as modelling and coding of text data. Moreover, students will use their own text corpus (or one of various text corpora provided for this course) to answer a substantive question from their personal research interests for a final project.

Details

- MA/PhD seminar
- Language: English
- Grading: 2 Homeworks (20% each); Research Paper (60%)

Learning Outcomes

At the completion of this course, students will be able to:

1. Understand fundamental issues in (quantitative) text analysis such as inter-coder agreement, reliability, validation, accuracy, and precision.
2. Convert texts into quantitative matrices of features, and then analyse those features using statistical methods.
3. Use human coding and annotations of texts to train supervised classifiers.
4. Apply these methods to a custom text corpus in order to tackle a substantive research question.

Introductory Readings

General Readings

The seminar does not build on a single text book, but relies mostly on papers and chapters of books. For a general overview of quantitative text analysis, natural language processing, and computational social science, the following books are recommended.

- **jurafsky**.
- **manning08**.
- **salganik18**.

Technical Background

The following books and websites are helpful to refresh and extend the knowledge of R, RMarkdown, and the `quanteda` package. Websites such as Stack OverFlow, R bloggers, and the documentation of R packages will be useful for solving practical problems. The books below are published in print, but also legally available online.

R, RMarkdown, and `quanteda`

- **wickham17**.
- **xie18**.
- **watanabemueller**.

Data Visualisation

- **healy19**.
- **wilke19**.

Software and Packages

The applications of the course are based on the R statistical programming language. Participants should download and install the latest versions of R and RStudio. Students should also install the latest releases of the following R packages, which will be used throughout the course.

- Quantitative text analysis: `quanteda`
- Importing text data: `readtext`
- Topic models: `topicmodels` and `stm`
- Data wrangling and visualisation: `tidyverse` (esp. `dplyr`, `tidyr`, `lubridate`, and `ggplot2`)
- Creating documents and reports: `rmarkdown` and `knitr`
- Part-of-speech tagging and lemmatisation: `spacyr` (installation not mandatory)

Additionally, I strongly encourage students to get used to `git` and set up a GitHub account (recently, GitHub started to provide unlimited private repositories even in their free version). The free and open-source software GitHub Desktop allows to use git and GitHub without having to rely on the terminal. The following sites contain comprehensible introductions to `git` and GitHub:

- https://guides.github.com/activities/hello-world/
- https://help.github.com/desktop/guides/getting-started-with-github-desktop/
- https://happygitwithr.com

Syllabus Modification Rights

I reserve the right to reasonably alter the elements of the syllabus at any time by adjusting the reading list to keep pace with the course schedule. Moreover, I may change the content of specific sessions depending on the participants' prior knowledge and research interests.

Expectations and Grading

- Students are expected to read all papers or chapters assigned under **Readings**. These readings serve as the basis for in-class discussions about the advantages, disadvantages, and applicability of the various approaches to social science questions. For each session, I also assign a variety of optional readings which are not mandatory, but I strongly encourage students to (at least) skim these reading. Both the required and the optional readings consist of technical readings and at least one practical application of the respective method.
- Students also submit a **Research Paper** which counts towards 60% of the final grade. The research paper is a written analysis consisting of 5,000–5,500 words (including bibliography, captions, and footnotes). Students are required to develop a research design to answer a question with textual data. Students are free to answer questions from all subfields of political science, but must justify their choice and the relevance of the question. Students registered for an MA degree in another social science discipline are encouraged to develop a research project answering a question from their subject. Students can use existing corpora, create their own text corpus, or access textual data that may be collected in spring at the Computational Social Science Hub (part of the Digital Democracy Lab). The research papers must be submitted via OLAT as a `pdf` document before **June 14, 2019 (8:00pm CET)**. In the 10th and 11th session, each student gives a short presentation, covering the research question, relevance, text corpus,

and methodological approach. Alongside with the presentation, students will submit a 1,000 words research proposal to receive comments from peers and the lecturer. Each project will be discussed through *written* feedback by another seminar participant, and students will also receive written feedback from me. Detailed instructions on the research paper, the presentation, and the in-class discussion will be provided via OLAT.

Overview of deadlines

Date	Time	Assignment
Friday, March 22, 2019	8:00pm CET	Homework 1 (20%)
Friday, April 26, 2019	8:00pm CET	Homework 2 (20%)
Friday, June 14, 2019	8:00pm CET	Research Paper (60%)

Course Structure

Week 1: Organisation and Introduction (February 18)

– What are quantitative text analysis and natural language processing?

– What is the structure of the course and what are the expectations?

– *Application*: installing packages and setting up a Project in RStudio

Readings

- **grimmer13**.
- **dimaggio15**.

Optional

- **lazer17**.
- **hirschberg15**.
- **gentzkow17**.

Week 2: Assumptions and Workflow (February 25)

– What are the underlying assumptions of text-as-data approaches?

– *Application*: importing textual data, creating a text corpus, and adding document-level variables

Readings

- **benoit18**.
- **wilkerson17**.

Optional

- **gilardi19**.
- **monroe15**.

Week 3: Tokenisation and Document-Feature Matrix (March 4)

– What are tokens, types, and features? What is the difference between stemming and lemmatisation?

– *Application*: tokenising texts, and creating a document-feature matrix

Readings

- **welbers17**.
- **manning08**.
- **denny18**.
- **watanabemueller**.

Week 4: Dictionaries and Sentiment Analysis (March 11)

– What are automated dictionary approaches? How can we create, test, and refine dictionaries?

– *Application*: creating multiword expressions and applying dictionaries to tokens objects and document-feature matrices

Readings

- **laver00**.
- **rooduijn11**.
- **soroka12**.
- **soroka18**.
- **proksch19**.

Optional

- **stine19**.
- **rudkowsky18**.
- **tausczik10**.
- **muddiman19**.
- **rauh18**.

Week 5: Textual Statistics, Text Similarity and Reuse (March 18)

– How do texts differ in their 'readability' and complexity? What are measures to estimate the similarity and distance between texts?

– *Application*: creating n-grams; estimating complexity and similarities/distances of texts

Readings

- **wilkerson15**.
- **cross17**.
- **bischof18**.

- **benoit19**.

Optional

- **allee16**.
- **linder18**.

Week 6: Human Coding and Document Classification (March 25)

– How can we classify documents into known and pre-defined categories? What is crowd-sourced coding?

– *Application*: typical workflow of human coding using crowdsourcing; Naïve Bayes classification

Readings

- **benoit09**.
- **mikhaylov12**.
- **benoit16**.
- **jurafsky**.

Optional

- **manning08**.
- **king17**.
- **hopkins10**.
- **watanabe18**.
- **diazlopez17**.
- **peterson18**.
- **loftis18**.

Week 7: Supervised Scaling (April 1)

– What are the assumptions, advantages, and problems of supervised scaling?

– *Application*: Worscores

Readings

- **laver03**.
- **laver14**.
- **baturo17**.

- **herzog15**.

Optional

- **lowe08**.
- **martin08b**.
- **perry17**.

Week 8: Unsupervised Scaling (April 15)

– What are differences between supervised and unsupervised scaling methods? How can we validate scaling models?

– *Application*: Wordfish and Wordshoal

Readings

- **slapin08**.
- **lowe13**.
- **schwarz17**.
- **kluever09**.

Optional

- **lauderdale16**.
- **catalinac18**.
- **greene16b**.
- **baerg20**.
- **storz18**.

Week 9: Topic Models (April 29)

– How does unsupervised document classification work? What are the assumptions, advantages, and caveats of topic models?

– *Application*: Latent Dirichlet allocation (LDA) and structural topic models (STM)

Readings

- **blei03**.
- **blei12**.
- **roberts14**.

Optional

- **grimmer10**.
- **greene17c**.
- **boussalis16**.
- **catalinac16**.
- **jacobi16**.
- **gilardi18**.

Week 10: Presentation of Projects [I] (May 6)

In this session, the first half of students will present their projects. The remaining projects will be presented in the following session. Detailed instructions on the presentations, the written outline of the research design, and how to discuss each other's proposal will be distributed through OLAT.

Week 11: Presentation of Projects [II] (May 13)

In this session, the second half of students will present their projects.

Week 12: Social Media and Multilingual Analysis (May 20)

– How can we analyse social media posts with text-as-data approaches? In what ways can we conduct multilingual analyses?

– *Application*: scraping Twitter data using an API; introducing platforms for machine translation

Readings: Social Media

- **mitts19**.
- **pfeffer18**.

Readings: Machine Translation

- **evans16**.
- **lucas15**.
- **devries18**.

Week 13: New Directions and Applications (May 27)

– What are future directions in natural language processing?

– *Application*: introducing assumptions of word2vec and deep learning approaches

Readings

- **king13**.
- **lecun15**.
- **mikolov13**.
- **mueller18b**.
- **joo**.

Optional

- **chollet18**.
- **salganik18**.