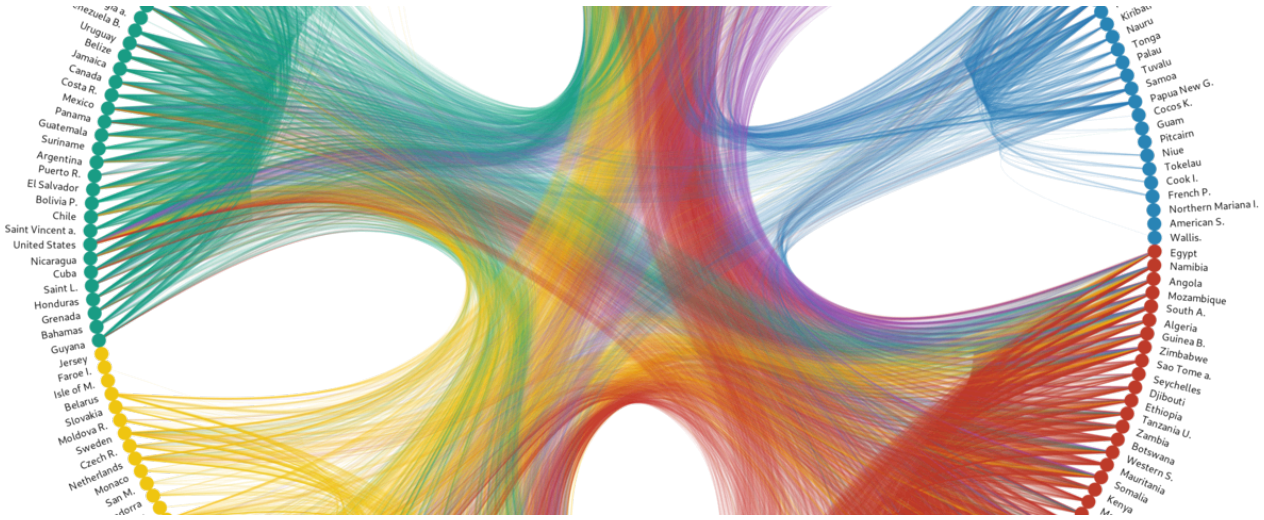
BA Seminar: The ABC of Computational Text Analysis

BA Seminar, Spring 2023, University of Lucerne

Alex Flückiger

17 January 2023

Contents



In this hands-on seminar, bachelor students of social and cultural sciences learn the basics of programming, among other essential technical skills. Building on a modern technology stack, it aims to prepare students to conduct data-driven text analysis and to make everyday life easier by fostering technological fundamentals. While learning about the importance of computation in solving problems, we also discuss the current developments in information technology. In short, the course promotes digital literacy on a practical and theoretical level.

This seminar focuses on the computational processing of digital and digitized texts using Python and the command-line. For any empirical research, the systematic preparation and aggregation of data and the swift retrieval of information are critical. These tasks require the handling of various data forms, including data that is not yet structured in a tabular format. The seminar covers the complete workflow, from gathering textual data to analyzing an entire text collection to producing interactive visualizations. Sounds cool? It certainly is.

Along the way, we deal with questions like these:

- How can texts be quantitatively exploited to complement the qualitative content analysis?
- What are regular expressions, and why are they so powerful in the context of computational text analysis?
- How to download data automatically from websites and process *en masse*?
- How can historical texts be extracted from PDFs using Optical Character Recognition (OCR)?

Go to Course Website

1 Schedule

We have 13 seminar sessions together.

The plan below is provisional. I am happy to adapt the topics, as well as the schedule, to the needs and interests of the students. Likely, we will change some topics and orderings as we go.

Date	Topic
23 February 2022	Introduction + Where is the digital revolution?
02 March 2022	Text as Data
09 March 2022	Setting up your Development Environment
16 March 2022	Introduction to the Command-line
23 March 2022	Basic NLP with Command-line
30 March 2022 (Zoom)	Learning Regular Expressions
06 April 2022 (Zoom)	Working with (your own) Data
~~13 April 2022~~	*no lecture (Osterpause)*
20 April 2022	Ethics and the Evolution of NLP
27 April 2022	Introduction to Python + VS Code
04 May 2022	Pandas und LIRI-Dataset
11 May 2022	NLP with Python
~~18 May 2022~~	*no lecture (Christi Himmelfahrt)*
25 May 2022	NLP with Python II + Working Session
01 June 2022	Mini-Project Presentations + Discussion

2 Lectures

Below you find a brief description of all the lectures. I make the slides available before the lecture starts.

2.1 Week 1: Introduction + Where is the digital revolution?

On the one hand, I present the goals and organization of the seminar. On the other hand, we look at some recent applications that give an impression of the fascinating prospects of computers in the area of artificial intelligence (AI) and digital humanities (DH).

2.1.1 Required Reading

- Lazer, David, Alex Pentland, Lada Adamic, Sinan Aral, Albert-László Barabási, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, Tony Jebara, Gary King, Michael Macy, Deb Roy, and Marshall Van Alstyne. 2009. "Computational Social Science." *Science* 323(5915):721–23.

2.1.2 Optional Reading

- Graham, Shawn, Ian Milligan, and Scott Weingart. 2015. *Exploring Big Historical Data: The Historian's Macroscope*. Open Draft Version. Under contract with Imperial College Press. online

2.2 Week 2: Text as Data

Computational text analysis comes with many challenges that are unique due to the fuzziness of natural language. In this session, we learn about its methodological foundation, and we conduct our first computational text analysis to understand how this translates into practice.

2.3 Week 3: Setting up your Development Environment

The title says it all. We are getting ready for the practical part of the course: Programming. As the installation of Python and non-standard command-line tools may be tricky, we do this in class rather than doing it as homework. Moreover, I will also introduce some principles to organize research and jargon that guide your way in the programmer's brave new word.

2.3.1 Optional: pimp your workflow

- Healy, Kieran. 2019. "The Plain Person's Guide to Plain Text Social Science." online.

2.4 Week 4: Introduction to the Command-line

The command-line is a powerful tool at your disposal. It is the working horse for many data wrangling tasks. In this session, you learn the basics of shells and perform many operations by effectively substituting clicks on the screen with commands. Admittedly, it is not overly exciting at this stage, yet it is essential for more sophisticated automation later on.

2.4.1 Recommended Resources

- Cheatsheet for this course
- The Programming Historian
- DigitalOcean

2.5 Week 5: Basic NLP with Command-line

Counting words is the most basic method to look at texts from a computational perspective. The command-line provides tools to quickly sift through a massive text collection to describe the use of words quantitatively. In no time, you can also take a systematic look at the word usage in context. Sounds like a Swiss knife for computational text analysis in social science? It certainly is.

2.6 Week 6: Learning Regular Expressions

When working with text data, you spend a lot of time cleaning your documents and extracting some pieces of information. Doing this by hand is not only a pain but simply impossible when facing more than a few dozens of documents. Fortunately, a formal language named Regular Expressions allows writing expressive and generalizable patterns to match specific text. Using these patterns, you can systematically extract and remove any textual parts without missing a single instance.

2.6.1 Required Reading

- Ben Schmidt. 2019. Regular Expressions. online

2.6.2 Recommended Resource

Everything we have touched about text processing in greater detail.

- Nikolaj Lindberg. egrep for Linguists. online

2.6.3 Online Regular Expression Editor

- regex101 is a visual editor to check your regular expressions.

2.7 Week 7: Working with Data

To this point, you have acquired the skills to cut a document into pieces and, subsequently, to extract, replace, and count any textual elements. Unless you have interesting data, these tools are neat but of no greater use. Thus, we turn to relevant data resources for social science. Given you have plain text at hand, your tools cut through data like butter. For other formats like PDF or DOCX, we learn some remedies to convert them into plain text. Most notably, we perform optical character recognition (OCR) .

2.8 Week 8: Ethics and the Evolution of NLP

Ethics is not just an abstract topic of Philosophy. Modern NLP is more powerful than ever before and, thus, embedded in many aspects of life. Unfortunately, it also exhibits severe and not yet well-understood bias that causes harm. With the recent *data-driven deep learning turn*, NLP overcame many theoretical limitations – yet, this comes at a cost. It is our duty to better understand the working and impact of this technology.

2.9 Week 9: Introduction to Python

It may come as a surprise that we start with Python in the ninth session only. As the folks say, Python is among the coolest programming languages, relatively easy to learn, and provides excellent NLP packages so that you don't have to implement everything yourself. All true as long as you have your data ready. In this session, we begin with an introduction to the basic syntax of Python. Starting with basics is a dry matter; however, it allows you to use third-party libraries and get a handle on more sophisticated NLP analyses.

2.10 Week 10: NLP with Python

Python is the language of choice when it comes to advanced NLP. Have you ever wondered how the frequency of terms has evolved over the years? Or how the language differs between two groups whereby the groups may be formed by any metadata (people, organization, gender etc.)? In such an exploratory endeavour, using an interactive and visual mode is the most effective that complements basic statistics. In short, we finally arrived at the serious stuff in our journey. To make sure, you don't get lost in the forest of yet unknown terms you will also learn the jargon of NLP.

2.10.1 Code

Click to run the code in your browser without any installation

2.11 Week 11: NLP with Python II + Working Session

In today's session, we continue our deep dive into NLP with Python. It is the last piece of our puzzle. During this course, you have learned about the entire workflow, from assembling datasets of documents to analyze their content and visualize your findings. As soon as you have a structured text collection along with basic metadata (e.g., publication date), you can take numerous perspectives to look at your data. At this stage, it is time to kick-off the mini-projects allowing you to work with your data of interest.

2.11.1 Explore interactively: 1 August Speeches by Swiss Federal Councilors

As a matter of tradition, Swiss Federal Councilors give an official speech on the Swiss National Day. Simon Schmid (journalist at Republik), with the collaboration of Prof. Andreas Kley (Faculty of Law, UZH), collected many of these speeches and kindly shared the resulting dataset with me. The collection comprises 166 speeches, which is a multiple of the publicly available here.

The interactive visualization linked below shows how the language differs between speakers of *Social Democratic Party of Switzerland* (SP) and speakers of other parties. The top right corner shows terms that have been frequently used by all parties. In contrast, the top left and the lower right corner reveal words that have been used primarily by the members of the SP and correspondingly by the centre-right parties.

You can search for the terms of your interest. Moreover, you may click on the points in the plot to show the context of the corresponding words within speeches. These functions allow for a quick investigation of the corpus along the dimensions of Swiss parties.

2.12 Week 12: Mini-Project Presentations + Discussion

In this session, it is your turn. Going beyond mere toy examples, you present what you have worked on and show off your first harvest of computational text analysis.

The seminar is coming to an end, yet it doesn't have to be a dead-end. You may have gotten more proficient in cursing your computer but also fighting your way through the jungle of technology. Keep going, cheers!

3 Assignments

You have to submit three assignments to complete the seminar successfully. The purpose of the assignments is not making the course hard to pass but rather to foster your engagement with the covered topics. As you like, you may prefer to work in teams to discuss different approaches. Nonetheless, each student has to come up with their own solution and submit it before the deadline.

#	Topic	Published	Deadline (by midnight)	Solution
1	Data Wrangling	24 March 2022	31 March 2022	Example
2	Regex NLP	08 April 2022	15 April 2022	Example
3	Python NLP	13 May 2022	20 May 2022	Example

3.1 Formal Instructions

Your submission is a single script, meaning that is readily executable, and is named as follows:

- Bash scripts: `SURNAME_KED2022_NR.sh`
- Python scripts: `SURNAME_KED2022_NR.py`

The script follows the order of the tasks in the assignment. In addition to the commands you have used in your solution, you also provide a short, yet concise explanation to your solution. You should include these comments directly in the script by starting the line with `#`.

Please use the following examples as template:

3.1.1 Bash

```
#!/bin/bash

#####################################################
### assignment 1
### Seminar: The ABC of Computational Text Analysis
### University of Lucerne
#####################################################

### task 1a)
echo "this is a test"
# solution: echo prints out the provided text in the command-line
```

```
### task 1b)
echo -n "test" | wc
# solution: wc counts the lines, words and characters.
# The argument -n is necessary to omit the trailing new-line symbol.
# "test" has 4 characters.

...
```

3.1.2 Python

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

##################################################
### assignment 1
### Seminar: The ABC of Computational Text Analysis
### University of Lucerne
##################################################

### task 1a)
print("Hello, World")
# outputs the provided string to the prompt

...
```

4 Mini-Project

You conduct a small computational text analysis and present the results in the final session. To give as many options as possible, you are free to choose your research question as well as computational methods and data you are going to use. It is certainly more fun when you work with data from your area of interest.

Again, the aim of this project is not to overwhelm students with too ambitious requirements. It should be the other way around. You will have as much freedom as you need to engage with your data creatively. I will be glad when you realize that your knowledge is already good enough to perform powerful analyses.

The only requirement is to complement your claims with some quantitative facts about the data, which you can freely choose. You may work in teams of two.

4.1 Inspiring Student Projects

- Gender differences in 1 August speeches [Code], Dario Haab, Valentina Meyer, Nils Brun, 2022
- Analysis of Bulletin Board Systeme (BBS), Josias Bruderer, 2021

5 Optional Seminar Paper

You are welcomed to write an optional seminar paper (Hauptseminararbeit) for which you get additional credit points. As I am in the position of a guest lecturer, I will accept seminar papers in cooperation with Prof. Sophie Mützel.

Due to the practical foundation of this seminar, you are well-prepared to subsequently apply computational text methods in a personal project. Although this is not a requirement, you may want to turn your mini-project into a seminar paper by deepening your empirical inquiry.

Students planning to write a seminar paper should send me an email with a short outline of their research idea until **15 May 2022** at the latest. When you would like to discuss your idea in person, feel free to do so any time after the seminar.

Requirements for the seminar paper (Hauptseminararbeit):

- Write your thesis in German or English.
- Use any computational methods to analyze your data.
- Your paper has a theoretical question guiding your methodical approach. In other words, methods are a means, not an end in themselves.
- Formal: 15 pages (A4), 12 pt Times New Roman, 3cm margin, 1.5 line spacing.
- Deadline for submitting the final paper: **31 August 2022**.