BA Seminar: The ABC of Computational Text Analysis

Alex Flückiger | University of Lucerne

23 Februar 2021

Contents

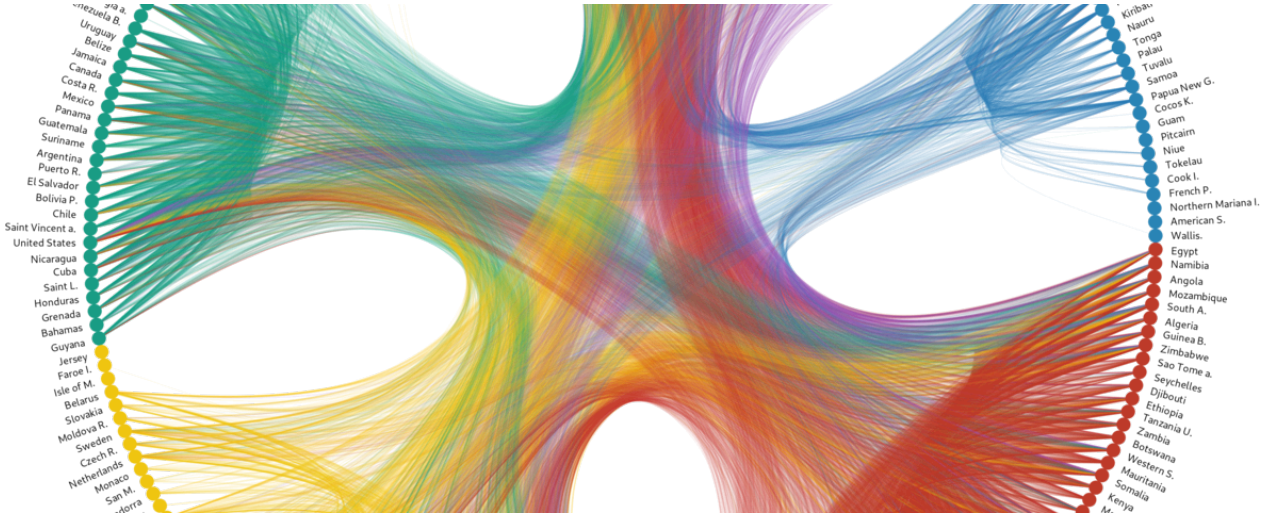

Figure 1: Country mentions in Speeches at the United Nation's General Debate

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 as well as to make everyday life easier by fostering technological fundamentals. While learning about the role of computation in solving problems, we also discuss the current developments in information technology. In short, the course promotes practical and theoretical digital literacy.

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 as well as the swift retrieval of relevant pieces of information. These tasks require handling a wide variety of data forms, including data that is not yet structured in a tabular format. The seminar covers the full 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 UniLu website

1 Schedule

We have 12 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
25 February 2021	Introduction + Where is the digital revolution?
04 March 2021	Text as Data
11 March 2021	Setting up your Development Environment
18 March 2021	Introduction to the Command-line
25 March 2021	Basic NLP with Command-line
01 April 2021	Learning Regular Expressions
08 April 2021	*no lecture (Osterpause)*
15 April 2021	Advanced RegEx + Data Sources
22 April 2021	Creating new Data Sets + Ethics
29 April 2021	Introduction to Python
06 May 2021	NLP with Python
13 May 2021	*no lecture (Christi Himmelfahrt)*
20 May 2021	NLP with Python + Working Session
27 May 2021	Mini-Project Presentations + Discussion
03 June 2021	*no lecture (Fronleichnam)*

2 Lectures

Below you find a brief description of all the lectures. I make the slides available before the lecture starts. The slides are provided in the following formats and may be opened by clicking the icon:

- HTML to open in a browser
- PDF document
- Markdown source

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 help you to find 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

- 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 to clean your documents and extract some pieces of information. Doing this by hand is not only a pain but simply impossible when facing many documents. Fortunately, there is a formal language named Regular Expressions that allows writing expressive and generalizable patterns. Using these patterns, you can extract and remove any textual parts systematically 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

A visual editor to check your regular expressions.

- Rubular

2.7 Week 7: RegEx + Data Sources

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. Besides some further practicing with RegEx, we turn to relevant data resources in social science. Given you have plain text at hand, your tools cut through this data like butter. For other formats, we learn about some remedies in the next session.

2.8 Week 8: Creating new Data Sets + Ethics

The world we live in is not made for machines but people – for better or for worse. While perfectly readable, documents often require a subsequent conversion to allow machine processing. Firstly, digital documents are shipped in various formats and need a conversion to plain text. Secondly, historical documents require an additional step called optical character recognition (OCR) to extract the text from the scanned original. Converting thousands of documents is easy when using the shell.

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 this dry subject is necessary as it allows you to modify the more sophisticated NLP analyses to your needs.

2.10 Week 10: NLP with Python

Python is the language of choice when it comes to serious NLP. Have you ever wondered how the frequency of terms evolves over the years? Or how the language differs between two groups of documents whereby the groups may be formed by any metadata (person, organization, gender etc.)? Exploring is most effective in an interactive and visual mode - so be it. Among some basic statistics, this is the serious stuff where we finally arrive in our journey. Moreover, you will learn the jargon of NLP to don't get lost in the forest of yet unknown terms.

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 in 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 meta data (e.g., publication date), you can take numerous perspectives to look at your data. At this stage, it is time for the kick-off of the mini-projects allowing you to work with your data of interest.

2.12 Week 12: Mini-Project Presentations + Discussion

In this session, it is entirely your turn. Going beyond mere toy examples, you present what you have worked on in groups and showing 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 making your way through the jungle of technology. Continue the journey, 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
1	Data Wrangling	18 March 2021	25 March 2021 (by midnight)
2	Regex NLP	01 April 2021	08 April 2021 (by midnight)
3	Python NLP	06 May 2021	13 May 2021 (by midnight)

3.1 Formal Instructions

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

- Bash scripts: `SURNAME_KED2021_NR.sh`
- Python scripts: `SURNAME_KED2021_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

```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.

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 **10 May 2021** 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 2021**.