GESIS Fall Seminar in Computational Social Science 2022

Syllabus for week 2:
“Automated Web Data Collection with Python”

Lecturers:	Felix Soldner	Jun Sun	Leon Fröhling
Affiliation:	GESIS – Leibniz Institute for the Social Sciences	GESIS – Leibniz Institute for the Social Sciences	GESIS – Leibniz Institute for the Social Sciences
Email:	Felix.soldner@gesis.org	Jun.Sun@gesis.org	leon.froehling@gesis.org

Date: September 12-16, 2022
Time: 10:00-17:00

About the Lecturers

Felix Soldner is a postdoctoral researcher at the Computational Social Science Department at GESIS – Leibniz Institute for the Social Sciences. He uses Natural Language Processing and Machine Learning methods to analyze texts in his research. While his broad research interest is in computational social science, he also works on topics related to crime, such as fraud, deception detection, or darknet markets.

Jun Sun is a postdoctoral researcher at the Computational Social Science Department at GESIS – Leibniz Institute for the Social Sciences. He graduated from TU Dresden in computer science in 2015 and received his PhD in 2022 at the University of Stuttgart on the phenomena in growing networks and learning across networks.

Leon Fröhling is a doctoral researcher at the Computational Social Science Department at GESIS – Leibniz Institute for the Social Sciences. His research focuses on developing and implementing frameworks for the critical reflection on data collection processes and the identification and documentation of potential sources of systematic errors.

Course Description

The continuously growing importance of the internet for everyday life and the correspondingly increasing volume of digital behavioral data on the Web allows us to study human behavior from new perspectives. However, accessing or collecting such data is not always straightforward. Moreover, the heterogeneity of collected data poses the challenge of data pre-processing, ensuring that they can be effectively used in further analyses. Thus, this course aims to introduce participants to data collection from online platforms and the pre-processing necessary to make it usable for their research. Apart from these essential, technical foundations, we will also discuss basic methods to enrich raw, textual data with additional features. Lastly, we will present participants with a framework for the critical reflection on their data collection processes and documentation of their data.

This course will show and teach participants how content, comment, and interaction data can be automatically collected from social media platforms (e.g., Twitter, YouTube, Reddit) or other online platforms (e.g., eBay, Amazon). We will cover the main aspects of collecting data using the programming language Python, including APIs and custom scrapers for static and dynamic webpages. We will also show how collected data can be cleaned, pre-processed, and curated to enable further statistical analyses.

The course will include lectures on each topic, introducing the basic theoretical concepts necessary for understanding the practical implementations, which are then practiced during exercises. The exercises will be conducted in small groups and assisted by the instructors, who may help with questions and problems. In mini-projects, participants have the chance to discuss how they can apply and integrate the newly learned methods within their research.

Keywords

Automated data collection, Web scraping, APIs, Dataset curation

Course Prerequisites

- Basic knowledge of the programming language python
- Motivation to work with various web-data sources
- Willingness to engage in hands-on coding exercises to learn how to collect web-data

Target Group

Participants will find the course useful if:

- they are interested in working with web data
- they want to learn how to collect web data through APIs or webpages
- they want to learn how to pre-process and augment the collected data for further analyses (basic NLP)
- they want to learn about frameworks for the critical reflection on web-data collection processes

Course and Learning Objectives

By the end of the course, participants will:

- be able to collect online data with APIs and custom scrapers for static and dynamic websites
- be able to handle, (pre-)process and augment data for further statistical analyses
- be able to integrate the learned methods into their research
- be able to reflect and inspect automatically-collected data critically

Organizational Structure of the Course

The course will be structured around lectures in which we explain the material and methods and exercises in which participants can explore and practice what they learned. Lectures are scheduled in the morning and exercises in the afternoon, separated by a lunch break. The morning and afternoon sessions will have short coffee breaks.

Exercises will be made up of small coding assignments, prepared by the instructors in advance and designed to be solved by participants directly in Notebooks using Google Colab, and “mini-projects” in which participants can apply the newly learned methods on their own projects. Participants can work alone or in small groups during that time. Throughout the exercises, instructors will support participants in their individual or group work (e.g., conceptual or coding issues). After coding assignments, instructors will provide walk-through solutions.

Software and Hardware Requirements

Participants should bring their own laptops. The course will use Google Colab (https://colab.research.google.com/), so there is no need to have Python installed on your machine. However, you will need a Google account and an up-to-date version of Google’s Chrome web browser.

Day-to-day Schedule and Literature

Day 1: Introduction to APIs

In the morning, we will start with a short discussion about the expectations of the course and how participants envision using web-scraping in their work. We then follow up with an introduction about web-scraping and basic concepts, such as APIs and custom scrapers.

After the lunch break, we will give an interactive introduction of basic commands of the Reddit API and how to work with JSON files (common outputs received from APIs). Participants will have time to practice using the Reddit API, including how to save and work with the Reddit data. Lastly, we will guide participants through the process of obtaining their individual YouTube API keys, needed for the following sessions.

Literature:

- Li, F., Zhou, Y., & Cai, T. (2021). Trails of data: Three cases for collecting web information for social science research. *Social Science Computer Review*, *39*(5), 922-942.
- Nyhuis, D. (2021). Application programming interfaces and web data for social research. In *Handbook of Computational Social Science*, Volume 2. Routledge.

Day 2: Working with the Reddit and Youtube APIs
Building on the content discussed on Day 1, we will deepen participants' understanding of APIs, discussing common APIs for data sharing, as well as social media APIs. Using the Reddit Pushshift API as an example, in the first two lecture sessions in the morning, we will show how to use Pushshift and the wrapper package PSAW to query Reddit data. Using the YouTube Data API as an example we will also show how to use APIs that require credentials in the lecture session in the afternoon. All lecture sessions are followed by exercises.

Literature (for reference):

- *Pushshift.io: Learn about big data and social media ingest and analysis*
- *PSAW: Python Pushshift.io API Wrapper (for comment/submission search)*
- *YouTube Data API Documentation*

Day 3: Introduction to Web-Scraping
On day 3, we will introduce how to web scrape static websites. We will cover general knowledge of HTTP requests, HTML and CSS in the first lecture in the morning. In the exercise that follows, we will then cover how to systematically extract web data from html pages with beautifulsoup. In the afternoon lecture session, we will cover how use requests, regex and selectorlib to scrape static web pages. In the exercise that follows, we will then cover how to systematically extract web data from html pages with requests, regex and selectorlib.

Literature (suggested reading):

- *Lab for Data Mining with Pandas on Wikipedia data* by Brian Keegan, Department of Information Science, CU Boulder
- *PyCon 2015 Pandas tutorial* by Brandon Rhodes
- *The dataquest blog* by Vik Paruchuri

Day 4: Dynamic Web-Scraping
In the morning, we will discuss what dynamic web pages are and how to obtain information from them using the python package Selenium. Participants will learn how to navigate (click, scroll, etc.) through webpages and retrieve information automatically.
In the afternoon, participants will have the time to practice the presented skills with prepared exercises. We will finish with a short overview of best practices for web scraping, common scraping problems, and how to overcome them.

Literature:

- Selenium with python (documentation)
- Luscombe, A., Dick, K., & Walby, K. (2021). Algorithmic thinking in the public interest: navigating technical, legal, and ethical hurdles to web scraping in the social sciences. *Quality & Quantity*, online first, 1-22.

Day 5: Web-Data Processing and Documentation

On the last day of our course on the automated collection of web-data with Python, we will first learn about the different steps necessary to prepare the raw, mostly textual data into suitable formats for subsequent analysis, and finally discuss the importance of properly documenting datasets collected from the Web.

In the first lecture of the day, we will introduce the basics of Natural Language Processing (NLP) and thereby understand why textual data needs to be preprocessed, before looking into the most popular preprocessing methods. In the exercise, we will try out available Python libraries (e.g., spaCy, NLTK, huggingface) to prepare some of the data that we collected during the previous days for further analysis.

In the second lecture of the day, we will demonstrate the importance of comprehensively documenting web-data datasets, especially if used to do research on human behaviour and interactions. We will introduce different approaches for the documentation of datasets and for the critical reflection on potential sources of bias and error in the data collection process. In the exercise, we will use these frameworks to examine the data that we collected during the previous days for systematic errors, and document one of the collection processes as well as the resulting dataset.

Literature (suggested reading):

- Bender, E. M., & Friedman, B. (2018). Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6, 587-604.
- Sen, I., Flöck, F., Weller, K., Weiß, B., & Wagner, C. (2021). A total error framework for digital traces of human behavior on online platforms. *Public Opinion Quarterly*, 85(S1), 399-422.