Universität Konstanz

Introduction to Computation for the Social Sciences

Assignment 7

Prof. Dr. Karsten Donnay, Marius Giebenhain, Stefan Scholz
Winter Term 2019 / 2020

Please solve the exercises below and commit your solutions to our GitHub Classroom until Dec, 17th midnight. Submit all your code in one executable file (*py* / *ipynb*) and your results in one file (*csv*). You can score up to 10 points in this assignment. You will get individual feedback in your repository.

Exercise 1: Simple Web Crawler (10 Points)

Web Crawling, i.e. the automated access of web resources via software, and web scraping, i.e. the extraction of content from web resources, are essential techniques to gather data for many analysis tasks in the social sciences. We will implement a very simple web crawler that starts at a given *URL* and iteratively accesses all links, more specifically all *href*-tags in the *HTML* source code of a website.

To get an idea how to approach this problem using Python, have a look at the sample implementation of a crawler in the subchapters *Traversing a Single Domain* and *Crawling an Entire Site* in the book *Web Scraping with Python* by *Ryan Mitchell*. If you search for the book online, you will find a PDF version of it.

Implement an iterative crawler (no recursive calls) that:

- opens the front page of the *German Wikipedia*[1] and downloads the *HTML* resource
- parses the *HTML* file as a BeautifulSoup object using the package *bs4*
- collects all internal *href* tags, i.e. only links to other Wikipedia pages on the server de.wikipedia.org
- opens each of the collected links and parses the returned html resource for additional internal links

Additionally, your crawler has to meet the following requirements:

- Consider the Wikipedia front page as the root (level 0) of a tree and the Wikipedia pages linked to on the front page as level 1. Crawl no deeper than level 2!
- Visit each unique link only once, i.e. disregard links that you traversed before
- Store all links of the same level in a list, i.e. one list per level.
- Include links that you encounter more than once in the list of the lowest level (closer to the root) they appear at.

- Write the lists of links to a CSV file in the format [level ; link_URL].

Finally, submit your solutions by adding your code and csv file in your private repository within the folder *assignment07 > solution*.

[1] https://de.wikipedia.org/wiki/Wikipedia:Hauptseite