Documentation

Overview

This crawler is used to mine faculty university faculties and generate a "json" file with an entry per professor. This entry will have the following fields: faculty, url, location, email, name, top_terms and bio. The purpose of this was to feed the Expert Search app with structured text that had additional information but couldn't fit the data to work with the MeTA corpus types. It can be used to gather info on professors, including the top five terms related to their bios.

Implementation

The code borrows from MP2.1 to crawl through faculties' professor profiles. We give an array of faculty home pages and from there the script tries to infer what links to professor profiles we have in it. Then using nltk's word_tokenize we determine the words with the highest frequencies in the bios text, by counting frequency of words and normalizing by max frequency. After that we use a "heuristic" with the aid of BeautifulSoup and regex to try to gather the email, facultie and name of the professor. I couldn't implement location as I didn't setup the maps api that was used in the Expert Search's get_location.py script. At the end, it dumps a json with the faculties data in a file called "bios_json.txt" under the sample folder, alongside the main.py

Usage

We assume a machine running python 3.5 using the pip installer
The script that needs to run is main
To run this script you need to install the following packages:

- bs4: pip install beautifulsoup4
- selenium: pip install -U selenium
- nltk: pip install nltk

For selenium we assume you're testing with the firefox dirver. If not, get the driver for your browser and use it in the constructor on main.py line 181. Some browsers might get tricky.

To specify which university faculties you want to crawl, modify dir_url in main.py line 185