1. Data Acquisition - Web Crawler/Scraper

Task

- Select a web source of your own choice for the non-trivial web crawling task.
  - The resource should contain hundreds/thousands of unique pages to crawl.
  - Each page should contain at least:
    - Title - e.g. an article title, a product title, …
    - Main content/text - a main text of the article, a description of the product, …
    - Additional features describing the page - information about author, date of publishing, product item parameters, …
- Identify possible issues with crawling:
  - Explore the robot exclusion protocol, availability of the sitemaps description, …
  - Identify issues with extraction of relevant information
    - Extraction using machine readable annotations, own set of rules/selectors, automatic detection of the content, …
- Properly design and implement the extraction task
  - Inputs and outputs of the task
  - Dealing with policies
  - Selection of the language/tools
- Configure the crawler
  - focus on crawling of just one single host (domain)
  - set the crawl interval! Otherwise you can be banned!
  - set the crawl depth
  - user-agent string
  - seed URLs
  - and other settings you consider important.

Instructions for submitting

In your private namespace on EDUX provide the following information:

- Describe the web resource
  - e.g. main URL, extracted information
- Describe possible issues with crawling
  - e.g. policies, …
- Describe the design of the extraction task
  - Inputs and outputs of the task
- Implement the crawler/scraper
  - You can use any language - recommended is the scrapy in Python
- Store data in a structured format
  - e.g. simple JSON format
  - optional: Store data to a database of your choice - e.g. mongo, solr, …

- Provide the link to your implementation
  - You can use: https://gitlab.fit.cvut.cz [https://gitlab.fit.cvut.cz] or https://github.com/ [https://github.com/]
- Provide the link to extracted data
- Comment on
  - issues during the design/extraction
  - ideas for extensions/improvements/future work

Ideas/Motivating Examples

- Crawling articles from specific domain
  - e.g. news articles
- Crawling blog posts
- Crawling tweets
- Crawling e-shop articles
- Crawling discussion/comments
- Extraction of data from social networks
- …

/mnt/www/courses/MI-DDW.16/data/pages/hw/01/start.txt · Poslední úprava: 2017/02/19 18:08 autor: kuchajar