Laboratorium 5. Pozyskiwanie danych

Za wieloma systemami lub modelami językowymi stoi ogromny zasób danych tekstowych. Istnieje wiele repozytoriów z danymi, które pozwalają na budowanie modeli. Skąd oraz w jaki sposób te dane są pozyskiwane? W obecnych czasach serwisy internetowe stanowią pokaźne źródło treści zapisanych w języku naturalnym. Laboratorium to pozwoli poznać techniki na pobieranie treści z wybranych stron internetowych. Rozwiązaniem tego laboratorium będzie przydatne narzędzie do scrapowania oraz crawlowania danych tekstowych z internetu.

Zadanie 5.1. Po pierwsze: API

Niejednokrotnie przed przystąpieniem do skrapowania danych warto jest sprawdzić czy serwis nie oferuje API. Taką możliwość daje serwis wolnelektury.pl, które jest pokaźnym zbiorem utworów literackich w języku polskim. Wykorzystaj framework `Requests` wraz z narzędziem `Beautiful Soup`

1. Zapoznaj się z API serwisu wolnelektury.pl
2. Napisz skrypt pobierający plik z informacjami o wszystkich dostępnych utworach.
3. Za pomocą danych z poprzedniego podpunktu pobierz 20 losowych utworów.
4. Wypisz tytuł, treść, autora, gatunek oraz epokę dla każdego utworu.
5. W skrypcie pod zadaniem napisz komentarz z propozycją innego serwisu jaki zawiera API.

Dokumentacja:
- `Beautiful Soup`[1]
- `Requests`[2]

Zadanie 5.2. Skrobanie treści

Następne zadanie to implementacja programu, który scrapuje wskazane dane z lokalnego serwisu informacyjnego Gazeta Wrocławska - Wiadomości[3]. Wykorzystaj do tego zadania bibliotekę `Scrapy` lub framework `Requests` wraz z narzędziem `Beautiful Soup`

1. Zaimplementuj program, który pobiera tytuł dowolnego artykułu z serwisu "Gazeta Wrocławska - Wiadomości"
2. Użyj funkcję `Inspect` w przeglądarce, aby zidentyfikować element z tytułem na stronie html
3. Zapoznaj się z plikiem `robots.txt` serwisu "Gazeta Wrocławska - Wiadomości"
4. W skrypcie pod zadaniem napisz komentarz wyjaśniający znaczenie oraz zawartość tego pliku

Dokumentacja:
- `Scrapy introduction`[4]
- `XPath selector`[5]
- `Inspect`[6]

Zadanie 5.3. Pełzanie po stronach

Kolejne zadnie to implementacja programu, który pobiera dane z więcej niż jednej strony automatycznie szukając nowych hiperłączy. Wykorzystaj do tego zadania bibliotekę `Scrapy` lub framework `Requests` wraz z narzędziem `Beautiful Soup`

1. Napisz program typu Crawler:
   - Program powinien odnajdywać na stronie głównej hiperłącza do artykułów
   - Następnie dla każdego znalezionego hiperłącza, które należy do serwisu, pobierz tytuł artykułu oraz pierwszy akapit treści
   - Ustaw opóźnienie na co najmniej 2 sekundy
   - Rozwiązanie może być rozszerzeniem skryptu z poprzedniego zadania
2. Pobrane tytuły i treść wypisz na terminalu
3. W skrypcie pod zadaniem napisz komentarz wyjaśniający w jakim celu ustawia się opóźnienie

Dokumentacja:
- `Crawl Spider`[7]
- 'Find all'[8]

[1] <https://www.crummy.com/software/BeautifulSoup/bs4/doc/index.html>
[2] <https://requests.readthedocs.io/en/latest/>
[3] <https://gazetawroclawska.pl/wiadomosci/>
[4] <https://docs.scrapy.org/en/latest/intro/tutorial.html>
[5] <https://docs.scrapy.org/en/latest/topics/selectors.html#working-with-xpaths>
[6] <https://blog.hubspot.com/website/how-to-inspect>
[7] <https://docs.scrapy.org/en/latest/topics/spiders.html?highlight=rule#crawlspider>
[8] <https://www.crummy.com/software/BeautifulSoup/bs4/doc/#find-all>