Assignment 1: Web crawler + Data store

CE6155 Web Intelligence and Message Understanding (2024 Spring)

ncu-wimu-2024@googlegroups.com

NCU WIDM Lab

Outline

- Description
- Example
- Grading Policy
- Schedule

Description

- The final goal of this semester is to complete a website navigation chatbot.
- In the first assignment, we must select the website we want to navigate, crawl the data on the website, and store it in the database.
- You can choose the website of any NCU department or the other university website. (Don't be the same as the sample code)
- Import the data to Elasticsearch to provide search function

crawler.py - Crawl Web information

- Crawling on [NCU Chiness]
- Design a crawler to crawl titles, URLs, anchors, text content, and all information that can be helpful for website navigation.
- [Crawling tutorial - LINK]
- Sample code：[LINK]

crawler.py - Crawl Web information

- The sample code uses the BFS algorithm to crawl down all the links on the NCU Chiness website.
- We use visited to avoid crawling to duplicate websites.
- In fact, the website structure is usually not tree-like. You can think about whether there is a better way.

```python
def bfs_crawl(root_url, max_depth):
    visited = set()
    queue = deque([(root_url, 0)])
    results = []

    while queue:
        current_url, depth = queue.popleft()

        # 停止條件：  (parameter) max_depth: Any
        if depth > max_depth:
            break

        # 如果該 URL 已經訪問過，跳過
        if current_url in visited:
            continue

        # 獲取該 URL 的所有超連結
        title, content, links = ncu_crawl(current_url)

        if title and content and links:
            print("Depth:", depth, "URL:", current_url)

            # 將所有未訪問過的超連結加入隊列，並更新深度
            for text, link in links:
                if link not in visited:
                    queue.append((link, depth + 1))

            # 標記當前 URL 為已訪問
            visited.add(current_url)
            results.append({'url': current_url, 'title': title, 'depth':depth, 'content'
    return results
```

crawler.py - Crawl Web information

- Save the crawled result to a JSON file
- In the sample code, we save the crawled web pages in json format as follows.

```
'''
data = {
        'url': text,
        'title': text,
        'depth': integer,
        'content': text,
        'links': [tuple(text,text)]
    }
'''
```


```
{
    "url": "https://www.chinese.ncu.edu.tw/?page_no=2&category%5B%5D=5c417a3b1d41c83169000197&tags%5B%5D=all",
    "title": "國立中央大學中國文學系",
    "depth": 2,
    "content": "國立中央大學中國文學系\nOpen login\nClose login\nclose\n登入 國立中央大學中國文學系\nUsername\nPassword\n登入\n忘記密碼
    "links": [

        [
            "回首頁|",
            "https://www.chinese.ncu.edu.tw/"
        ],
        [
            "中央大學首頁|",
            "https://www.ncu.edu.tw/"
        ],
        [
            "招生訊息|",
            "https://www.chinese.ncu.edu.tw/zh_tw/Curriculum2/Admission2/Undergraduate2"
        ],
        [
            "規章辦法|",
            "https://www.chinese.ncu.edu.tw/zh_tw/About2/Details_Regulations/Departmental_Rules2"
        ],
        [
            "網路導覽",
            "https://www.chinese.ncu.edu.tw/zh_tw/sitemap"
        ],
```

ElasticSearch – Run Elasticsearch

ElasticSearch tutorial - LINK

1. Download Elasticsearch

2. Open cmd and change directory and run bin\elasticsearch.bat

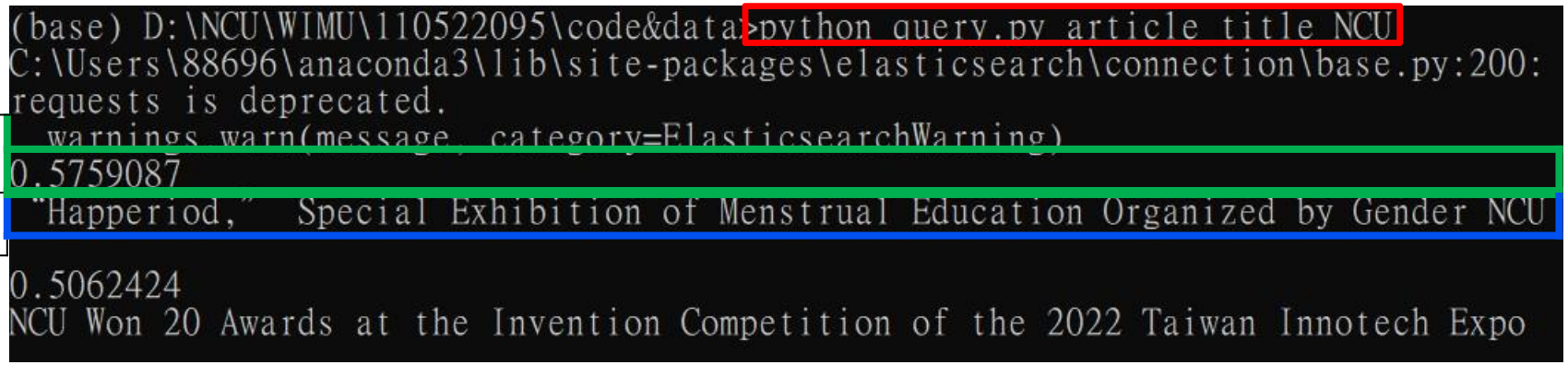
Change directory to Elasticsearch directory
Running bin\elasticsearch

```
D:\NCU\WIMU\elasticsearch-7.11.2>bin\elasticsearch.bat
[2023-02-25T10:57:11,264][INFO ][o.e.n.Node               ] [LAPTOP-PEGS45MB] version[7.11.2], pid[9600], build[default/
zip/3e5a16cfec50876d20ea77b075070932c6464c7d/2021-03-06T05:54:38.141101Z], OS[Windows 10/10.0/amd64], JVM[AdoptOpenJDK/O
penJDK 64-Bit Server VM/15.0.1/15.0.1+9]
[2023-02-25T10:57:11,322][INFO ][o.e.n.Node               ] [LAPTOP-PEGS45MB] JVM home [D:\NCU\WIMU\elasticsearch-7.11.2
\jdk], using bundled JDK [true]
[2023-02-25T10:57:11,324][INFO ][o.e.n.Node               ] [LAPTOP-PEGS45MB] JVM arguments [-Des.networkaddress.cache.t
tl=60, -Des.networkaddress.cache.negative.ttl=10, -XX:+AlwaysPreTouch, -Xss1m, -Djava.awt.headless=true, -Dfile.encoding
=UTF-8, -Djna.nosys=true, -XX:-OmitStackTraceInFastThrow, -XX:+ShowCodeDetailsInExceptionMessages, -Dio.netty.noUnsafe=t
rue, -Dio.netty.noKeySetOptimization=true, -Dio.netty.recycler.maxCapacityPerThread=0, -Dio.netty.allocator.numDirectAre
nas=0, -Dlog4j.shutdownHookEnabled=false, -Dlog4j2.disable.jmx=true, -Djava.locale.providers=SPI,COMPAT, -XX:+UseG1GC, -
Djava.io.tmpdir=C:\Users\88696\AppData\Local\Temp\elasticsearch, -XX:+HeapDumpOnOutOfMemoryError, -XX:HeapDumpPath=data,
 -XX:ErrorFile=logs/hs_err_pid%p.log, -Xlog:gc*,gc+age=trace,safepoint:file=logs/gc.log:utctime,pid,tags:filecount=32,fi
lesize=64m, -Xms3936m, -Xmx3936m, -XX:MaxDirectMemorySize=2063597568, -XX:G1HeapRegionSize=4m, -XX:InitiatingHeapOccupan
cyPercent=30, -XX:G1ReservePercent=15, -Delasticsearch, -Des.path.home=D:\NCU\WIMU\elasticsearch-7.11.2, -Des.path.conf=
D:\NCU\WIMU\elasticsearch-7.11.2\config, -Des.distribution.flavor=default, -Des.distribution.type=zip, -Des.bundled_jdk=
true]
```

Import Data and Query

- **import2_elasticsearch.py** – Import crawled data into ElasticSearch
- **query.py** – Use ElasticSerach to input the query and find the result
  - Please design your requirements to query the result (ex. Find article title)

Print query score
Print article title



import2_elasticsearch

Elasticsearch	Index	Type	Mapping	Document	Field
	⬇	⬇	⬇	⬇	⬇
Relational Database	Database	Table	Schema	Row	Column

- Define the elasticsearch index name, type name(The new version of elasticsearch does not require), mapping structure

```python
def load2_elasticsearch():
    index_name = f'ncu_chinese'
    type = 'one_to_one'
    es = Elasticsearch()
```

```python
hp_mapping = {
    "properties": {
        "url": {
            "type": "text"
        },
        "title": {
            "type": "text"
        },
        "depth": {
            "type": "integer"
        },
        "content": {
            "type": "text"
        },
        "links": {
            "type": "text"
        }
    }
}
```

Query

- In the sample code, use the most basic match query.

```
# Query DSL
search_params = {
    "query": {
        "match": {item: query}
    }
}
```

- Please refer to Query DSL to improve the accuracy of your query.
- Tip: In the field of NLP, vector query is often used, you can refer to Script score query.

Report

1. (15%) Please record the performance statistics of the web crawler, for example: how long does it take to crawl 100 web pages, how long does it take to crawl 500 web pages? Then, how do you speed up the process, and what is the resulting performance after speeding up?

2. (15%) Based on the URLs you have crawled, analyze the structure and layout of the website, and draw an approximate sitemap of the website.

3. (20%) Please compare the differences in query results using different query methods. Please attempt to use embedding models for vector search.

Report: LINK

Overview

- Please write a **crawler** to get structured data and use the **Elasticsearch** to query data you want.
- Please pack the file into a .zip file and upload it to the ee-class system.
  - Pack the file into std_ID.zip (ex. 110522095.zip) and need to include these files
    - **crawler.py –** to crawl the structured data you want
    - **query.py –** use elasticserach to input the query and find the result
    - **import2_elasticsearch.py –** import crawled data to elasticsearch
    - **result.json –** the crawl result
    - **report.pdf –** Please output the PDF file in A4 size

    5 points will be deducted if the payment format and file name is not as required.
- Sample code：LINK
  - Allow students to make modifications based on the sample code

Grading Policy

- **Program Demo (50%)**
  - (15%) Execute your code on site to confirm it works.
  - (15%) Use kibana to display your stored content.
  - (20%) Query specifies pagination, tell me how you search (you cannot use url or any primary key as query)
    - For example: how to find relevant information about previous department chairs?
    - You must use your query methods to find the page as follows.

歷屆系主任

名字	任期（學年度）	任期（年月）
王力堅	110-2、102、103、104-1	102年2月～105年1月
劉錦五	58、59、60	58年8月～61年7月
于大成	67、68、69	67年8月～70年7月
胡自逸	61、62、63、64、65、66	61年8月～67年7月
胡自逸	70	70年8月～71年7月
曾昭旭	75、76	75年8月～77年7月
蔡德發	71、72、73、74	71年8月～75年7月
蔡德發	77、78	77年8月～79年7月
張夢機	77、78	77年8月～79年7月
林平和	81、82、83、84、85、86	81年8月～87年7月

- **Report (50%)**

Schedule

- Schedule
  - Assignment 1 release: 2024/03/07
  - Assignment 1 submission: 2024/03/21 23:59
  - Late submission: score = score x (1 - (late days x 0.1))
  - Demo: 2024/03/25 ~ 2023/03/27 Please go to this LINK to fill in the demo time