Assignment #3: Additional Information

Understanding the data dump:

In Assignment-2, crawlers of all the groups collectively crawled 37,497 URLs. We collected these URLS and are providing them to you as 'webpages_clean.zip' file. This zip file contains the following:

1. bookkeeping.json
2. bookkeeping.tsv
3. Folders 0 to 74

Folders:
The 37,497 URLs are organized into 75 folders, each folder having 500 files. Every file has the extracted HTML source code of a particular URL.

Bookeeping files:
bookkeeping.json and bookkeeping.tsv are two different formats of the same file. These files maintain a list of all the URLs that have been crawled. Every URL has an identifier associated with it. This identifier helps locate the HTML code of the URL. The identifier is of the format: "folder_number/file_number"

For example, consider the entry on line 13 of bookkeeping.json:

"0/108": "vision.ics.uci.edu/papers/RamananBK_ICCV_2007"

This means that the HTML code extracted for the link "vision.ics.uci.edu/papers/Ramanan BK_ICCV_2007" is located at folder 0, file number 108.

Understanding the content of the files:

We have extracted the HTML source code of these URLs and cleaned them so that parsing the content of the URLs is easier for students. Hence, instead of giving you the full HTML code of a website, we have given you the code of only the required tags.

While cleaning the HTML files, we have kept only selected HTML tags and removed the rest of them. The list of HTML tags that are retained in the pages given to you are:

"<body>", "</body>",
"<title>", "</title>",
"<h1>", "</h1>",
"<h2>", "</h2>",
"<h3>", "</h3>",
"", "",
"", ""

Assignment #3: Additional Information

Please note that a given URL source code does not necessarily contain all the tags mentioned above. In fact, very few of the URLs contain the tags mentioned above. To extract the content from these tags, you will be using an HTML parser. There are many libraries available to achieve this task and we encourage you to compare the available options before selecting a library to perform HTML parsing for you (Suggestions: Beautifulsoup, HTMLParser)

Note: The content of the page will not contain the <title>, </title> tags. If plain text exists before the start of a <body> tag, this text is the title of the page and has been extracted from the <title> tags. For example, consider the content of the page:

UCI Machine Learning Repository <body>
Center for Machine Learning and Intelligent Systems
About
Citation Policy
Donate a Data Set
Contact </body>

Here, the title of the page is 'UCI Machine Learning Repository'.

Broken HTML:

The HTML source code of the URLs may not be well formed. This means that the code may not necessarily have a pair of opening and closing tags. For example, there might be an open tag but the associated closing tag might be missing. The HTML parsers that you will use to parse the documents should be able to handle broken HTML. Hence, as mentioned above, while selecting the parser for your project, please ensure that it can handle broken HTML.

Use of libraries:

It is strictly not allowed to use libraries that perform the entire task of index creation or ranking for you. Hence, libraries such as Lucene or Elastic Search are not allowed.

You may use libraries that help you achieve specific tasks. For example, you can use a tokenizer such as NLTK to tokenize your content.