Recuperação de Informação / Information Retrieval

2020/2021 MEI/MIECT, DETI, UA

Assignment 1

Submission deadline: **29 October 2020**

For this assignment you will create a simple document indexer, consisting of a corpus reader / document processor, tokenizer, and indexer.

The corpus for this assignment is available here: https://bit.ly/2Rg7gbX

Note: the dataset was obtained from ai2-semanticscholar-cord-19/historical_releases.html

1. Create a corpus reader that iterates over the collection (corpus) of document and returns, in turn, the contents of each document.
   For this assignment consider only the title and abstract fields and ignore documents with an empty abstract.

2. Create two tokenizers:
   - i. A simple tokenizer that replaces all non-alphabetic characters by a space, lowercases tokens, splits on whitespace, and ignores all tokens with less than 3 characters.
   - ii. An improved tokenizer that incorporates your own tokenization decisions (e.g. how to deal with digits and characters such as ’, -, @, etc).
     Integrate the Porter stemmer (http://snowball.tartarus.org/download.html) and a stopword filter. Use this list as default: https://bit.ly/2kKBCqt

3. Create an indexing pipeline. Use a suitable data structure for the index, defined by you.

4. Index the corpus using each tokenizer above and answer the following questions:
   - a) What was the total indexing time and how much memory (roughly) is required to index this collection?
   - b) What is your vocabulary size?
   - c) List the ten first terms (in alphabetic order) that appear in only one document (document frequency = 1).
   - d) List the ten terms with highest document frequency.

Instructions:

- Use Python or Java (in this case, manage your project with Maven)
- **Modelling**, code **structure**, **organization** and **readability** will be considered when grading your project
- **Comment** your code; and make sure you include your name and student number
- Write **modular** code
- Favour **efficient** data structures
- Use **parameters**, preferably through the command line
- Make sure all your programs compile and run correctly
- Submit your assignment by the due date using Moodle