Romanian Information Retrieval System

Iordache Adrian-Razvan

November 23, 2021

Project Description

For this project we will implement an Information Retrieval System for Romanian language.

This project will be composed of two main stages:

- Indexing Stage (will take a set of documents and will create an inverted index which will be later used for different queries)
- Searching Stage (based on the inverted index previously created we will be able find information from an unstructured collection of documents efficiently)

As far as the implementation goes we will create two classes that can run independently (Indexer and Searcher) and a third one (Retrieval System) which will be used as a wrapper over the other two classes.

Based on the previous affirmation, we will be able to run this project in two ways, using all of it as a unitary framework with the Retrieval System class or each class separately with specific arguments for each one.

Now we can dive further in the explanation of each class:

Indexer Class

This class will have two major roles, one for data cleaning and the other one for document indexing.

In this project we will handle text cleaning with the following methods:

1. Removing diacritics
2. Removing stop words
3. Text stemming

The second and third preprocessing step will be automatically done by the Romanian Analyzer from Lucene API.

We need to take into consideration the fact that Romanian Analyzer handles only the stop words with diacritics (like 'și', 'că', 'să'), but we want our system to remove stop words without diacritics also.

For that we will take the default list of Romanian stop words from the Lucene API (placed in utils folder) and before the creation of the Romanian Analyzer we will update that list with the stop words without diacritics and pass it to the constructor of Romanian Analyzer.

In the indexing stage:

- we will take each document from a specified folder (.pdf, .txt, .doc or .docx files)
- extract the content from it, implementation based on the Tika Library
- removing the diacritics from the content of the document
- store it inside the inverted index with the help of the IndexWriter object from Lucene API

Searcher Class

Based on the searcher class we will be able to query the inverted index to find the documents that contain the information that we want.

The searcher will take two arguments:

- the path to a query file (which will contain the query for the searcher)
- the path to the inverted index (already created by the indexer)

Retrieval System Class

This class will be used as a wrapper over the previous two classes, this will allow us to use this project as single framework.

For more information about dependencies and how to run this project

Please check the README.txt file