UIBK

Exam

Date Analysis II: Information Retrieval and full-text search

Student: Nikulina Kseniia

Innsbruck, 2022

1 Question 1

What are the parts of an information retrieval system? How would you characterize them?
Information retrieval is the activity of finding material (usually documents) of an unstructured nature (usually text) from a large collection of documents, that satisfies an information need of the user. There are four main parts of information retrieval system:

- Internal representation of the documents
  The documents are mapped from an external representation to an internal representation called Indexing, which is then stored in a system. Document indexing is the identification of specific attributes of a document to simplify and expedite accurate retrieval of a document. This is accomplished with an index, a system used to make finding information easier with descriptive data.

- A set of valid queries
  User defines a set of valid queries, so we can know which documents are valid for our document search. Basically, user defines filters for documents on this step.

- A set of operations that creates a subset of the documents from a query
  During this step retrieval process is happening. Retrieval is a process of matching the documents with the user's query and returning back the desired result.

- A set of operations that rank documents according to a query This process is based on a retrieval function which returns a list of documents ordered as per their relevance with respect to the query.

2 Question 2

Which of the following statements are correct? You are allowed to tick either "true", "false" or "it depends". If you choose "false" or "it depends", please justify your claim.

- "In scenarios where a user wants the results sorted by e.g. filename or date, you do not need to calculate the TF-IDF."
  True

- "The TF-IDF score of a document depends only on how often a term shows up in a document."
  False TF-IDF weight is composed by two terms: the first computes the normalized Term Frequency (TF) - the number of times a word appears in a document, divided by the total number of words in that document; the second term is the Inverse Document Frequency (IDF), computed as the logarithm of the number of the documents in the corpus divided by the number of documents where the specific term appears.

- “You cannot combine stemming technology with lower-casing function. These technologies will get in each other’s way, and the result of the preprocessing would be useless.”
  False Because stemming is basically removing the suffix from a word and reduce it to its root word. (flying -> fly) and lower-casing functions returns a character string that is the same length with each uppercase letter replaced by the corresponding lowercase letter. So, if we want to use this two technologies together it won’t be a problem because the outputs of these functions do not affect each other. If we want first implement lower-case and then stemming technology: FLyIng -> flying -> fly. Other case first stemming technology and then lower-case function: FLYIng -> FLY -> fly

- “You cannot combine stemming technology with lemmatizing technology. These technologies will get in each other’s way; thus the result of the preprocessing would be useless.”
  It depends Doing both stemming and lemmatization or only one will result in really slight differences and choosing one technology over another depends on result that we want to get. For lemmatization algorithm context is important, in the mean time stemming algorithm just chops the end of the word. If we use both of these algorithms together for one word then the result will be different sometimes, so before using one of these we first need to figure out the task and then choose one technology over the other. Stemming usually refers to a crude heuristic process that chops off the ends of words in the hope of achieving this goal correctly most of the time. Lemmatization usually refers to doing things properly with the use of a vocabulary and morphological analysis of words.
  Stemming technology is much faster as it’s a fairly simple case-based algorithm. Lemmatizing technology is much more expensive. Use stemming technology when the vocab space is small and the documents are large. Conversely, go with lemmatizing technology when the vocabulary space is large but the documents are small. If speed of searching is important then stemming should be used.

3 Question 3

See question3.py

4 Question 4

- function index: I would not change here anything

- function get terms from document: Here we need first perform function "split content into words"and only then preprocess words with function "pre process"

- function search: We need to add function "pre process"because for example customer will search "Pillow but in our dictionary we have only "pillow". It is the same, only difference is lower case