Lab 1: Information Retrieval
Reflection Report

TDDE16 – Text Mining

Anonymous ID
53604

IDA
Linköping University

November 21, 2023

RQ 1.1: *Why do we remove common stop words and lemmatize the text? Give an example of a scenario where, in addition to common stop words, there are also application-specific stop words.*

We remove common stop words, as these words rarely contributes to the context or meaning behind the text, i.e., they do not carry relevant information. These are often the most common words, for example "the", "is" and "and", but there can also be application-specific stop words. For example, in clinical or medical texts, "Dr." and "patient" might occur very often and can thus be regarded as stop words.

Lemmatization is the act of breaking down a word to its root meaning or root word. Lemmatization is also done to "increase" a words' relevance. A suffix or prefix to a word seldom adds to its relevance, therefore we want to reduce the words to their dictionary form, i.e., their lemma.

RQ 1.2: *In Problem 2, what do the dimensions of the matrix X correspond to? What information from the original data is preserved, what information is lost in this matrix?*

Each row in the matrix X corresponds to a description (there are 1614 description in the dataset), while the number of columns (in our case, 21357) corresponds to the preprocessed vocabulary (stop words removed and all words in their lemma form). More precise, the scikit learn function fit_transform() returns a tf-idf-weighted document-term matrix. Each row is a document (a description) as a tf-idf-vector representation.

Since X only stores the tf-idf-value, information of what the underlying word was is lost. More importantly, this results in the loss of structural information of the text. That is, information such as word order and grammar etc. can no longer be found in X. The indices of the words are however preserved, which makes it possible to map the tf-idf-values to the underlying words using a matrix of words.

RQ 1.3: *What does it mean that a term has a high/low idf value? Based on this, how can we use idf to automatically identify stop words? Why do you think is idf not used as a term weighting on its own, but always in connection with term frequency (tf–idf)?*

The more frequent a words' usage is across the corpus, the *lower* the idf-value will be. This means, that words with a low idf-value carry less information about the underlying text, and can thus be removed as stop words. As previously mentioned, typical English words with low idf values would be "the", "is" and "and", which can be automatically removed based on their idf-value. In other contexts, other words might be more frequently used without "carrying information", and can be removed on the same basis. On the contrary, words with high idf-values occurs less frequently, thus often carrying more information.

The term frequency (tf) is instead how many times a word appear in a document. This is used in connection with idf, since in many cases, information of how many times a specific term is mentioned in a document is highly important. For example, in a search engine, a document containing the search term multiple times, might be more relevant compared to a document containing the same term only once *(Note: relevance is not always linear in this way, which is why log-frequency weighting is often used).* The combination of the two values results in low values for terms that are common across the corpus (like "the"), but still yields high values for terms that might be mentioned often, but are still uncommon in the context of all documents.