Task4

Nawar Saeed

Artificial Intelligence for the Web, VT21
DT506A

16 april 2021

As the table below shows, some of them are the words with the most frequency but at the same time they are not. In this task, all duplicates have been removed, and this made new words fit in the ten-words list, as it looks above. If duplicates are not removed, the ten-words list will be similar to the ten-words list in Task3.
When it comes to running the code for construction of your inverted index and the code for queries with a profile, a library called $cProfile$ has been used. By using this library, it will be possible to see how long it takes for a function to run and what algorithmic operations are performed there and how long these take.
In main, $cProfile$ is called in lines 93 and 94 in Task4.py. This checks the two functions that Task4.py has, namely the parser() and word-frequency() functions. When it comes to parser (), 27474308 function calls are performed in 10.076 seconds. The methods $append, split, replace, lower$ and len take more time to run because they are called lots of times.
What has also been observed is that a search query does not take much time, e.g. keywords query Breastival $Vestibule's$ takes 0 seconds where only 27 function calls are performed. The len method is called 10 times and is thus the most called method. The reason it takes 0 seconds is because this search query returns a single hit. Another search query such as the movieåill probably return many hits and will require 1060694 function calls in 0.802 seconds, according to $cProfile$.

The ten words with the largest number of documents									
is	a	film	by	directed	and	the	or	in	written
184282	181918	171959	164427	148047	131141	107798	69615	66397	60900