Suffix Tree Construction

Jacob Bieker 201601099, Jonas Tranberg 20053514

April 2017

1 Overview

In this project, we implemented the McCreight suffix tree algorithm in Python. Our project does work. One problem we had was our algorithm had an off-by-one error when determining the locations of the string, which we overcame by adding one when returning the locations.

From running the tests with the given files and searches, the average number of bytes used per character is 1 byte overall. It changes from roughly 5 bytes with the "banana.txt" to 1 with "ancient-mariner.txt" and "walrus-and-carpenter.txt". Per suffix node, all texts had the same amount of bytes per node at 96 bytes on average.

For the construction time, the timing ranged from 0.83 seconds for the largest one, the Ancient Mariner, to 0.0000004 seconds for both the Banana and Mississippi texts.

Search time for all texts is 0.0000004 seconds.