

Algorytmy Tekstowe

Trie i drzewa sufiksów - raport

Jakub Pinowski

Do testów użyte zostało pięć przykładowych tekstów:

- bbb\$
- aabbabd
- ababcd
- abcbccd
- tekst z pliku "1997_714.txt"

1 Porównanie czasów działania

1.1 bbb\$

- Trie: 0.0596ms
- Simple suffix tree building: 0.0648ms
- McCreight: 0.0741ms

1.2 aabbabd

- Trie: 0.078ms
- Simple suffix tree building: 0.108ms
- McCreight: 0.147ms

1.3 ababcd

- Trie: 0.0236ms
- Simple suffix tree building: 0.03ms
- McCreight: 0.0265ms

1.4 abcbccd

- Trie: 0.4418ms
- Simple suffix tree building: 0.0760ms
- McCreight: 0.0679ms

1.5 Pierwsze 2000 znaków tekstu z pliku "1997_714.txt"

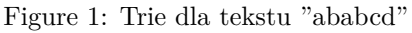
- Trie: 5.3250s
- Simple suffix tree building: 0.0198s
- McCreight: 0.007s

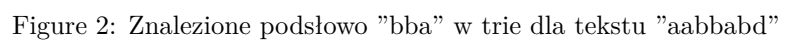
1.6 Tekst z pliku "1997_714.txt"

- Simple suffix tree building: 3.9253s
- McCreight: 2.1463s

Niestety stworzenie struktury trie dla całego pliku 1997_714.txt wymagałoby zbyt dużo pamięci jak na możliwości przeciętnego, domowego komputera, więc porównanie ograniczyło się tylko do 2000 pierwszych znaków (co i tak wymagało kilku gigabajtów miejsca) i dobitnie pokazało jak słabo wypada proste budowanie trie, nawet w porównaniu z prostym budowaniem drzewa suffiksów.

2 Przykładowo zbudowane struktury





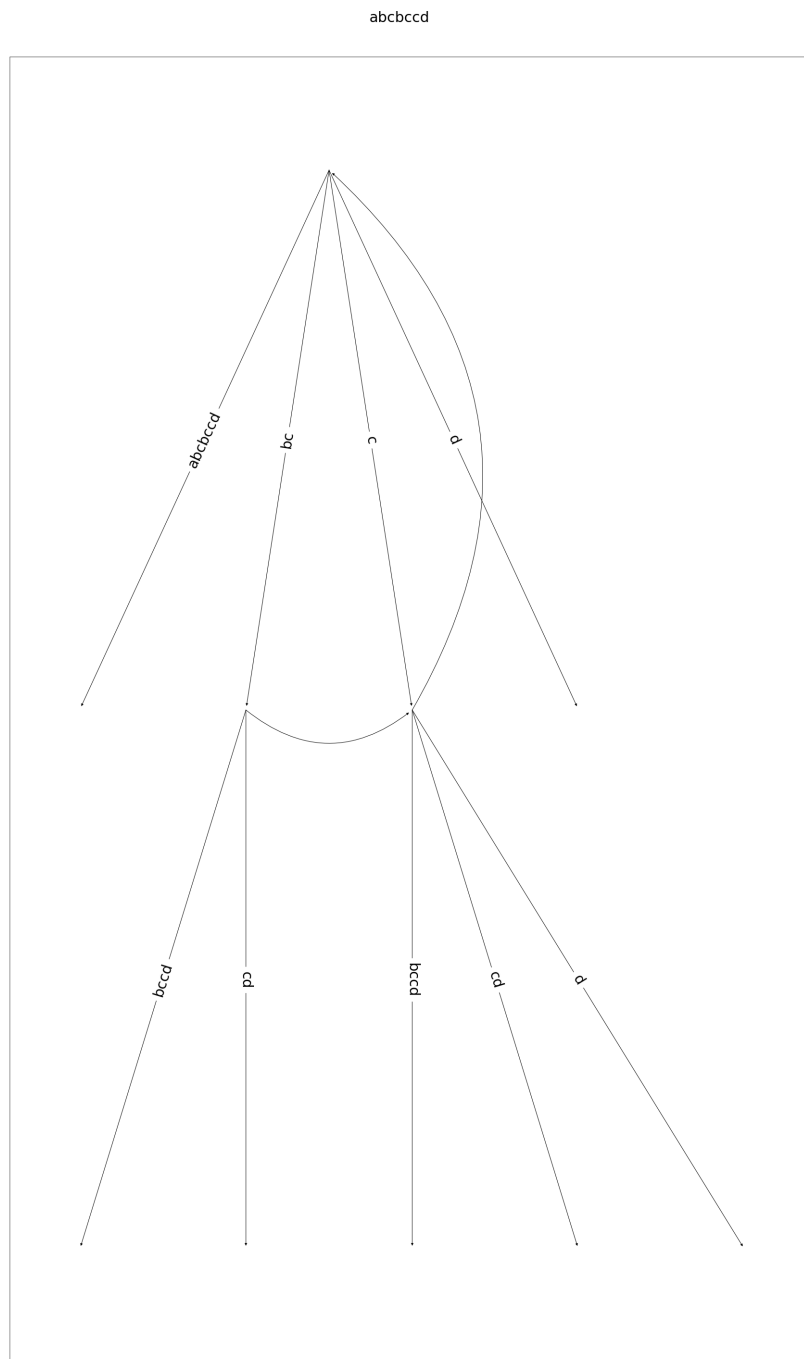


Figure 3: Drzewo sufiksów dla tekstu "abcbccd" zbudowane przy pomocy algorytmu McCreighta (krawędzie w kształcie łuków pokazują linki stworzone w trakcie działania algorytmu)