
Advanced Methods for Sequence Analysis

due: Monday, November 3rd, before the lecture

Exercise 1

- a) A naive approach to constructing suffix trees would be as follows: For $0 \leq i \leq n$, let S_i denote the compact Σ^+ -tree with $words(S_i) = \{T_{k\dots j} : 1 \leq k \leq i, k \leq j \leq n\}$. Start with S_0 , the tree consisting only of a root node. To obtain S_{i+1} from S_i , try to match $T_{i+1\dots n}$ as far as possible in S_i , and then insert a new edge that is labeled with the final (unmatched) part of $T_{i+1\dots n}$. What is the complexity of this algorithm?
- b) For our suffix tree construction algorithm, we have assumed that T ends with a unique character $t_n = \$$. Show how to derive efficiently the suffix tree for $T' = t_1 t_2 \dots t_{n-1}$ (i.e., without the $\$$) from the suffix tree for T . What is the complexity of your algorithm?

Exercise 2

Dr. Dumb proposes to represent edge label by *plain* strings $T_{i\dots j}$ instead of representing them by a pair (i, j) of indices in T . Give an infinite family of example strings where the lengths of the edge-labels in Dr. Dumb's suffix tree sum to $\Omega(n^2)$.

Exercise 3

Prove formally that a compact Σ^+ -tree with n leaves contains less than $2n$ nodes.

Exercise 4

Let $T = \text{antananarivo}\$$.

- a) Give the suffix- and LCP-array for T (only the final result).
- b) From the result of (a), show step-by-step how the algorithm presented in the lecture constructs the suffix tree for T .