Algorithms on Strings

Problems Set 12

Aleksander Czeszejko-Sochacki

February 2019

1 Solution

Given n strings $S_1, S_2, \ldots, S_N$, firstly note, that the shortest superstring (OPT) satisfies the formula:

$$|OPT| = \sum_{i}^{N} S_i - \sum_{i>1}^{N} |overlap(S_{i-1}, S_i)| \tag{1}$$

Where strings S_i are indexed in the order of occurences in the supertring. Hence we need to maximize $\sum_{i>1}^{N} |overlap(S_{i-1}, S_i)|$. Construct a weighted graph with the following properties:

- Each node v_i represents the string S_i
- The weight of the edge $v_i v_j$ equals $overlap(S_i, S_j)$
- Add v_0 node as the start node, weight of $v_0 v_i = 0$ for each i

Then note, that the maximum value of $\sum_{i>1}^{N} |overlap(S_{i-1}, S_i)|$ is equivalent to the Hamiltonian path in this graph. We are going to solve the problem of finding Hamiltonian path using Held-Karp dynamic algorithm for TSP (without the last edge, because the last string does not overlap with the first).

Lemma 1 (Shortest path from S_i to S_j)**.** *Let $min\_path(S_i, S_j, n)$ - the cheapest path from S_i to S_j visiting exactly n nodes. Then*

$$min\_path(S_i, S_j, n) = \min(min\_path(S_i, S_k, n-1) + v_k v_j) \tag{2}$$

for $k \notin S_i, S_j$

We added v_0 to the graph, because we do not have predefined start. Start at v_0 and consecutively build the cheapest paths of length $1, 2, \ldots, N$ to get the final result. The time complexity of computing the operation in lemma is $\mathcal{O}(n)$. We have $\mathcal{O}(n * 2^n)$ subproblems, so the final time complexity is $\mathcal{O}(n^2 * 2^n)$.