World Academy of Science, Engineering and Technology
International Journal of Biomedical and Biological Engineering
Vol:19, No:08, 2025

Finding a Set of Long Common Substrings with Repeats from m Input Strings

Authors : Tiantian Li, Lusheng Wang, Zhaohui Zhan, Daming Zhu

Abstract : In this paper, we propose two string problems, and study algorithms and complexity of various versions for those problems. Let $S = \{s_1, s_2, \ldots, s_m\}$ be a set of m strings. A common substring of S is a substring appearing in every string in S. Given a set of m strings $S = \{s_1, s_2, \ldots, s_m\}$ and a positive integer k, we want to find a set C of k common substrings of S such that the k common substrings in C appear in the same order and have no overlap among the m input strings in S, and the total length of the k common substring in C is maximized. This problem is referred to as the longest total length of k common substrings from m input strings (LCSS(k, m) for short). The other problem we study here is called the longest total length of a set of common substrings with length more than l from m input string (LSCSS(l, m) for short). Given a set of m strings $S = \{s_1, s_2, \ldots, s_m\}$ and a positive integer l, for LSCSS(l, m), we want to find a set of common substrings of S, each is of length more than l, such that the total length of all the common substrings is maximized. We show that both problems are NP-hard when k and m are variables. We propose dynamic programming algorithms with time complexity $O(k\, n_1 n_2)$ and $O(n_1 n_2)$ to solve LCSS(k, 2) and LSCSS(l, 2), respectively, where n1 and n_2 are the lengths of the two input strings. We then design an algorithm for LSCSS(l, m) when every length > l common substring appears once in each of the $m - 1$ input strings. The running time is $O(n_1^2 m)$, where n1 is the length of the input string with no restriction on length > l common substrings. Finally, we propose a fixed parameter algorithm for LSCSS(l, m), where each length > l common substring appears $m - 1 + c$ times among the $m - 1$ input strings (other than s1). In other words, each length > l common substring may repeatedly appear at most c times among the $m - 1$ input strings $\{s_2, s_3, \ldots, s_m\}$. The running time of the proposed algorithm is $O((n12^c)^2 m)$, where n_1 is the input string with no restriction on repeats. The LSCSS(l, m) is proposed to handle whole chromosome sequence alignment for different strains of the same species, where more than 98% of letters in core regions are identical.

Keywords : dynamic programming, algorithm, common substrings, string
Conference Title : ICBE 2025 : International Conference on Biomedical Engineering
Conference Location : Moscow, Russia
Conference Dates : August 30-31, 2025