Paper reference


Abstract

Given two biological sequences, the alignment problem seeks to find the highest-scoring match (alignment) between the two. A scoring function that rewards aligned pairs and penalizes gaps, insertions and deletions is common. The inverse sequence problem takes an alignment between sequences and a scoring function as given, and seeks to find the parameters of the scoring function that will result in the optimal score. The goal is to use this function, with those learned parameter values, to produce relevant alignments between a set of unseen sequences. Sun et. al. propose such an algorithm with $O(m \times n \times \log n)$-time for inverse global alignment without gap penalty and $O(m \times n \times \log m)$-time for inverse global alignment with gap penalty, where m is the length of the longer sequence.

Discussion

How do the results relate to the topic? The results presented by Sun et. al. precede those of Kececioglu and Kim. In particular, Sun et. al.’s approach requires that, when considering three parameters, one parameter value is effectively fixed at zero. In particular, the approach assumes a score function: $score_A = w_A - a \times x_A - b \times y_A - c \times z_A$, where $w_A$ is the number of matches, $x_A$ is the number of mismatches, $y_A$ is the number of indels and $z_A$ is the number of gaps, and $a$, $b$, and $c$ are the associated penalties (to be learned) for a given alignment, $A$. For the inverse global alignment without gap penalty, $c = 0$, and for the inverse global alignment with global alignment penalty, $a = 0$, and the algorithm searches parameter space over $0 \leq b \leq m^2$, and $0 \leq c \leq m^2$.

Is it an advance over what was previously presented? Kececioglu and Kim’s work represents an advance over the version presented by Sun et. al. Kececioglu and Kim can search and optimize over “hundreds of free parameters” and does not “directly compute a decomposition of the entire space,” as does Sun et. al. Sun et. al.’s approach, however, was not practically implemented, so making a direct comparison between this work and Kececioglu and Kim’s has some shortcomings.

Is it a different approach? The method proposed by Sun et. al. is in practice and concept distinct from the “Simple and Fast” paper. “Simple and Fast” uses a linear-programming approach, while the inverse global alignment uses a binary search approach. In particular, it splits up the b-axis (where b is the penalty for indels), and can decide to “go up or down the b-axis according to the number of indels.”

What are the main advantages/disadvantages? A key disadvantage of Sun et. al.’s approach is running time. The method proposed by Sun et. al. runs in $O(n^2 \times \log n)$ for two sequences of length n, while the one proposed by Kececioglu and Kim runs in polynomial time, as a consequence of the Separation Theorem. Further, Sun et. al.’s approach is specified for only three parameters, whereas in practice, computational biologists may stack on additional parameters that are linear in the sequence. For example, initiating a gap (parameter c) may be costly, but extending that gap may not be as costly. This sort of distinction is not possible to capture using three fixed parameters.