CS207 Design and Analysis of Algorithms

Sajith Gopalan

Indian Institute of Technology Guwahati

sajith@iitg.ac.in

February 14, 2022

Dynamic Programming

Dynamic Programming

- Particularly useful for optimization problems. Solution space. Constraints. Many feasible solutions, each of which satisfies the constraints, and has a value. We wish to find one feasible solution that minimises (maximizes) value.
- Steps:
  - Formulate the structure of an optimal solution
  - Recursively define the value of an optimal solution
  - Compute the value of an optimal solution, typically in a bottom-up fashion
  - Construct an optimal solution from computed information
- Works particularly when the following properties hold:
  - Optimal Substructure: an optimal solution to the problem contains within it optimal solutions to subproblems
  - Overlapping subproblems: recursive algorithm for the problem solves the same subproblems repeatedly

The Longest Common Subsequence Problem

- Consider sequences "ABDCBCD" and "AEFGBFHDBCDCHEBHEJCKCBD"
- The former is a subsequence of the latter
- Consider "AEFGBFHDBCDCHEBHEJCKCBD" and "AJKBAACCDCMMDBABCHHJDL"
- "ABDCBCD" is a subsequence is both; so it is a common subsequence
- There is no longer sequence that is a subsequence of both
- So, it is the longest common subsequence (LCS)
- That's our problem. Given two sequences, find their LCS.

The Longest Common Subsequence Problem

- Suppose X is a sequence of length m and Y is a sequence of length n
- Let Z of length k be an LCS of X and Y
- Let X_i denote the prefix of length i of X. That is, $X_i = X[1 \ldots i]$
- If $X[m] = Y[n]$ then $Z[k] = X[m] = Y[n]$ and Z_{k-1} is an LCS of X_{m-1} and Y_{n-1}

The Longest Common Subsequence Problem

- If $Z[k] \neq X[m] = Y[n]$, then add $X[m] = Y[n]$ to Z gets us a longer LCS. But Z is the LCS. Contradiction.
- So, $Z[k] = X[m] = Y[n]$
- Now suppose Z_{k-1} is not an LCS of X_{m-1} and Y_{n-1}, but W of greater length is
- Then $W.Z[k]$ is an LCS of X and Y and its length is more than k. Contradiction.
- So, Z_{k-1} is an LCS of X_{m-1} and Y_{n-1}

The Longest Common Subsequence Problem

- If $X[m] \neq Y[n]$ and $Z[k] \neq X[m]$, then Z is an LCS of X_{m-1} and Y
- Suppose, $X[m] \neq Y[n]$ and $Z[k] \neq X[m]$ and Z is not an LCS of X_{m-1} and Y
- But W of greater length is
- Then W is an LCS of X and Y as well. Contradiction.
-
- Analogously, if $X[m] \neq Y[n]$ and $Z[k] \neq Y[n]$, then Z is an LCS of Y_{n-1} and X

Length of LCS: Recursive formulation

- Let $l[i,j]$ denote the length of the LCS of X_i and Y_j
- Then

$$l[i,j] = \left\{ \begin{array}{ll} 0 & \text{if } i = 0 \vee j = 0 \\ l[i-1,j-1]+1 & \text{if } i,j > 0 \wedge X[i] = Y[j] \\ \max\{l[i,j-1], l[i-1,j]\} & \text{if } i,j > 0 \wedge X[i] \neq Y[j] \end{array} \right\}$$

- Once $l[i-1,j-1]$, $l[i-1,j]$ and $l[i,j-1]$ are known, $l[i,j]$ can be computed in $O(1)$ time
- So the l array can be computed in $\Theta(mn)$ time
- If we remember, at each (i,j), which of the three options applies, then the LCS can be printed out in linear time
- Note that there might exist multiple LCSes
- Our algorithm will find out one of them

The Longest Common Subsequence Problem: The Memoized Solution

- Exercise: Write the memoized solution