

最长严格递增子序列动态规划算法

1 问题描述

给定两个序列 $A = [a_1, a_2, \dots, a_m]$ 和 $B = [b_1, b_2, \dots, b_n]$ ，找出它们的最长严格递增子序列 (LIS)。

2 动态规划算法

2.1 状态定义

设 $dp[i][j]$ 表示序列 $A[1..i]$ 和 $B[1..j]$ 中的最长严格递增子序列的长度。

2.2 状态转移方程

- 如果 $A[i] = B[j]$ ，则最优解至少包含它们中的一个，若同时包含，则删去改 2 个元素后得到了输入为 $(i-1)(j-1)$ 以及输入元素是该问题子集的最优解，因此有递推关系；若不是同时包含，不妨设最优解中 B 不含第 j 个元素，则该元素在 j 之前已经有出现，否则与子序列相同矛盾，则将 B 中 $B[j]$ 最后一次出现的该元素与 $B[j]$ 替换，则仍是最优解，因此仍满足递推关系

$$dp[i][j] = dp[i-1][j-1] + 1$$

- 如果 $A[i] \neq B[j]$ ，最优解不同时包含这 2 个元素，则删去不含的一个，子序列长度不变，仍是缩小规模输入的最优解，因此有递推关系

$$dp[i][j] = \max(dp[i-1][j], dp[i][j-1])$$

2.3 初始条件

$$dp[i][0] = 0 \quad (\forall i)$$

$$dp[0][j] = 0 \quad (\forall j)$$

最长严格递增子序列的长度为 $dp[m][n]$ 。

Algorithm 1 填表过程

```

1: 输入: 序列  $A$  和  $B$ , 它们的长度分别为  $m$  和  $n$ , 序列内读入每个元素
2: 初始化: 二维数组  $dp$ , 大小为  $(m+1) \times (n+1)$ , 全部元素初始化为 0
3: for  $i = 1$  to  $m$  do
4:   for  $j = 1$  to  $n$  do
5:     if  $A[i] = B[j]$  then
6:        $dp[i][j] = dp[i-1][j-1] + 1$ 
7:     else
8:        $dp[i][j] = \max(dp[i-1][j], dp[i][j-1])$ 
9:     end if
10:  end for
11: end for
12: 输出:  $dp[m][n]$ , 最长严格递增子序列的长度
  
```

3 算法复杂度

该算法的时间复杂度为:

$$O(m \cdot n)$$

4 示例

令 $A = [a, b, c, b, d, a]$ 和 $B = [b, c, a, b, d]$ 。

4.1 初始化

$$dp = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

4.2 填表过程

动态规划表更新过程

初始化状态

	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$	0	0	0	0	0	0
$i = 1$	0	0	0	0	0	0
$i = 2$	0	0	0	0	0	0
$i = 3$	0	0	0	0	0	0
$i = 4$	0	0	0	0	0	0
$i = 5$	0	0	0	0	0	0
$i = 6$	0	0	0	0	0	0

更新第 1 列, 当 $i=2, j=1$ 发生增长, 对应与第一个 b 匹配)

	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$	0	0	0	0	0	0
$i = 1$	0	0	0	0	0	0
$i = 2$	0	1	0	0	0	0
$i = 3$	0	1	0	0	0	0
$i = 4$	0	1	0	0	0	0
$i = 5$	0	1	0	0	0	0
$i = 6$	0	1	0	0	0	0

更新第 2 列, $i=3, j=2$ 发生增长, 对应于 c 发生匹配

	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$	0	0	0	0	0	0
$i = 1$	0	0	0	0	0	0
$i = 2$	0	1	1	0	0	0
$i = 3$	0	1	2	0	0	0
$i = 4$	0	1	2	0	0	0
$i = 5$	0	1	2	0	0	0
$i = 6$	0	1	2	0	0	0

更新第 3 列, $i=1, j=3; i=1, j=6$ 发生增长, 对应于 a 的匹配

	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$	0	0	0	0	0	0
$i = 1$	0	0	0	1	0	0
$i = 2$	0	1	1	1	0	0
$i = 3$	0	1	2	2	0	0
$i = 4$	0	1	2	2	0	0
$i = 5$	0	1	2	2	0	0
$i = 6$	0	1	2	3	0	0

更新第 3 列, $i=2, j=4; i=1, j=6$ 发生增长, 分别对应 A 的第一个、第二个 b 和 B 中的元素 b 的匹配

	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$	0	0	0	0	0	0
$i = 1$	0	0	0	1	1	0
$i = 2$	0	1	1	1	2	0
$i = 3$	0	1	2	2	3	0
$i = 4$	0	1	2	2	3	0
$i = 5$	0	1	2	2	3	0
$i = 6$	0	1	2	3	3	0

更新第 5 列, $i=4, j=5$, 对应于 D 的匹配

	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$	0	0	0	0	0	0
$i = 1$	0	0	0	1	1	1
$i = 2$	0	1	1	1	2	2
$i = 3$	0	1	2	2	3	3
$i = 4$	0	1	2	2	3	4
$i = 5$	0	1	2	2	3	4
$i = 6$	0	1	2	3	3	4

最终结果

4.3 最终结果

最长严格递增子序列长度为:

$$dp[6][5] = 3$$