

第4章 贪心法

《人工智能算法》

清华大学出版社

2022年7月

提纲

- ◆ 应用背景和动机
- ◆ 贪心算法的基本思想
- ◆ 哈夫曼编码
- ◆ 总结

应用背景和动机 (1)

背景：主播带货已成为了一种新的产品推销手段。为响应国家脱贫攻坚和乡村振兴战略，边远山区的地方政府也采取主播带货的方式推广农产品。



例如：假设一批农产品要被想被大众熟知，则其影响因子需要达到1.3，现有3类主播，其中A类主播可帮助农产品提高0.4的影响因子，B类主播可帮助农产品提高0.3的影响因子，C类主播可帮助农产品提高0.1的影响因子。

问题：应该如何安排主播带货，能够在最少的主播数量下帮助政府使得该农产品被大众熟知？

直观的方案：尽量选择影响因子高的主播，即选择3名A类主播和1名C类主播，就可在最少的主播数量下让这批农产品的影响因子达到期望值。

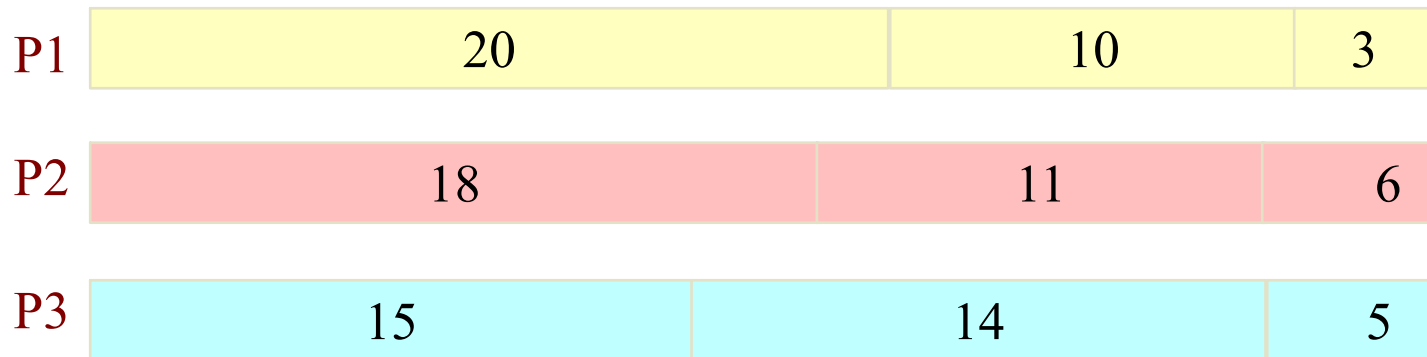
应用背景和动机 (2)

- ◆ **优化问题 (Optimization problem)**
不仅要找到答案，且要找到最佳答案
- ◆ **贪心算法 (Greedy algorithm)**
有时对优化问题能有较好的解决方案
- ◆ **贪心算法分阶段执行，每一个阶段：**
 - 当考虑做何种选择时，只考虑对当前问题最佳的选择，而不考虑其子问题的结果
 - 希望通过每一个步骤的局部最优 (Local optimum) 而得到全局最优 (Global optimum)
 - 未必能得到最优解

应用背景和动机 (3)

引例：多机调度

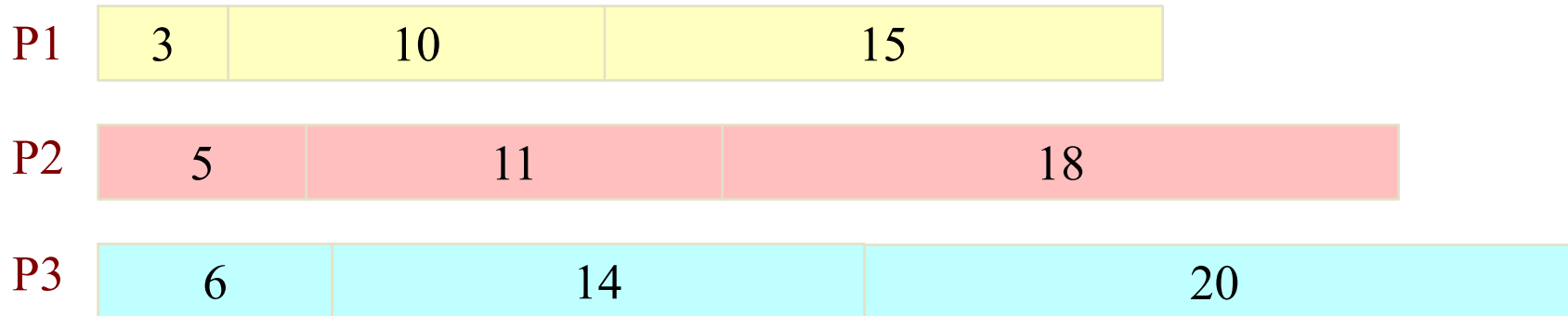
- ◆ 运行9个作业，其时间分别为3, 5, 6, 10, 11, 14, 15, 18, 20
- ◆ 共有3个处理器，可执行这些作业
- ◆ 按照**最长时间作业优先**的原则，将作业安排到空闲处理器



- ◆ 完成作业所需时间为 $18 + 11 + 6 = 35$
- ◆ 这一方案并不差，但是可能有更好的方案

应用背景和动机 (4)

若按照**最短时间作业优先**的原则，结果如何？

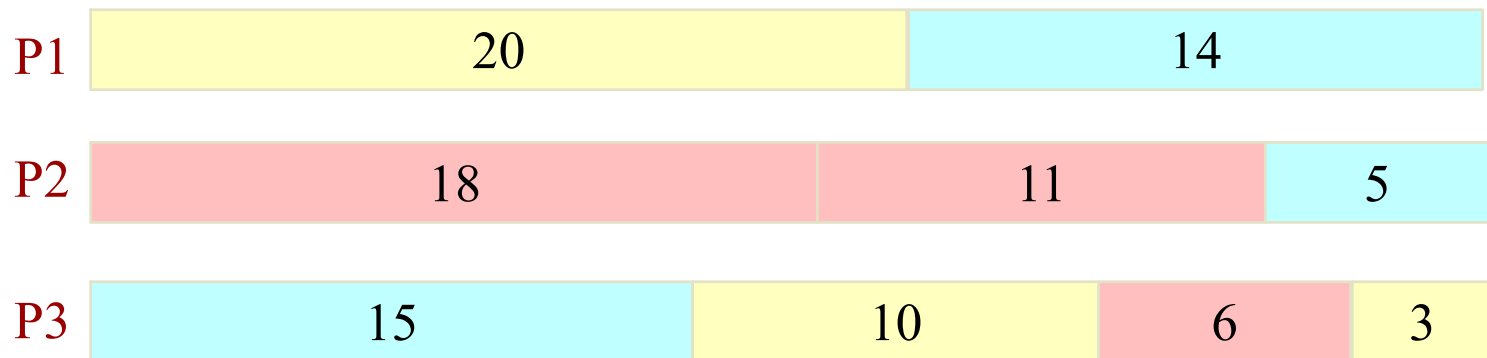


- ◆ 以上方案并不是一个好的方案，完成作业所需时间为 $6 + 14 + 20 = 40$
- ◆ 注意到：**贪心算法本身能高效地执行**

每个阶段所需：选择当前最小或最大者

应用背景和动机 (5)

更好的方案：



- ◆ 显然，该方案是最优的，仍可能有其他最优方案
- ◆ 如何得到以上最优方案？
 - 尝试所有作业安排给处理器的可能方案，选择最短完成时间
 - 需指数计算时间☹

结论：

贪心算法具有较高效率，其答案从实际应用的角度看已足够好

提纲

- ◆ 应用背景和动机
- ◆ 贪心算法的基本思想
- ◆ 哈夫曼编码
- ◆ 总结

贪心算法的基本思想

1、贪心选择性质

- 所求问题的**整体最优解**，希望通过一系列**局部最优**的选择（贪心选择）——贪心算法可行的第一个基本要素
- 通常以**自顶向下**的方式进行，以迭代的方式作出相继的贪心选择，每作一次贪心选择就将所求问题简化为规模更小的子问题
- 每一步对目前构造的部分解做一个扩展，满足：**可行（满足约束）、局部最优、不可取消**（贪心算法与动态规划算法的主要区别）

2、最优子结构性质

问题的最优解包含其子问题的最优解

证明贪心算法的正确性（针对最优化问题的求解）：

- 证明每一步所作的贪心选择最终导致问题的整体最优解
- 数学归纳法

提纲

- ◆ 应用背景和动机
- ◆ 贪心算法的基本思想
- ◆ 哈夫曼编码
- ◆ 总结

哈夫曼编码 (1)

◆ 背景

- 哈夫曼编码广泛地用于数据文件压缩
- 压缩率通常在20%~90%之间（变长码）
- 哈夫曼编码算法用字符在文件中出现的频率表来建立一个用0, 1串表示各字符的最优表示方式
- 给出现频率高的字符较短的编码，出现频率较低的字符以较长的编码，可以大大缩短总码长

◆ 前缀码

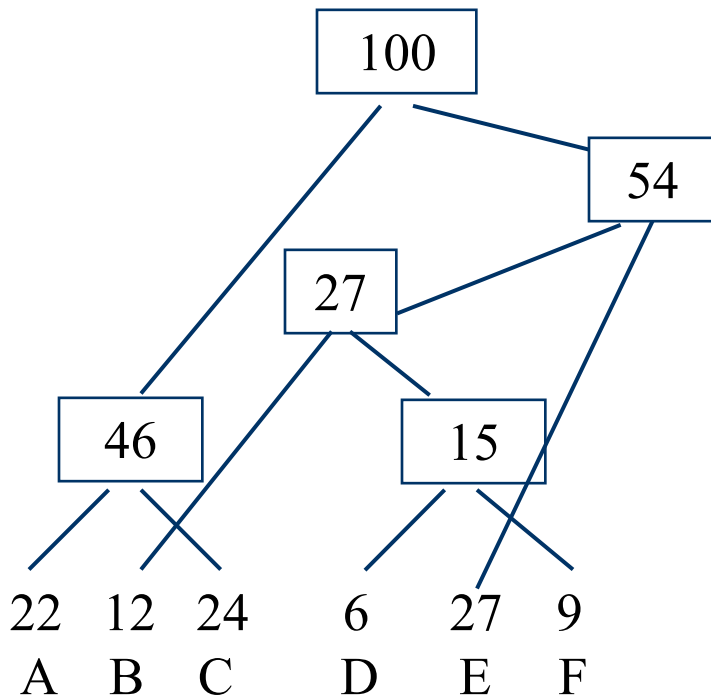
- 尽可能多地压缩文件、源文件很容易被重建
- 任一字符的代码(0,1序列)都不是其他字符代码的前缀——完全二叉树 T
- 满足前缀约束，则编码无二义性

- 平均码长：
$$B(T) = \sum_{c \in C} f(c) d_T(c)$$

字符 c 出现的频率为 $f(c)$ ，
在 T 中的深度为 $d_T(c)$

哈夫曼编码 (2)

- ◆ 哈夫曼编码（ Huffman encoding ）算法为贪心法
- ◆ 选择并合并最小出现频度的两个数字



编码结果

A=00
B=100
C=01
D=1010
E=11
F=1011

◆ 平均编码长度

$$0.22*2 + 0.12*3 + 0.24*2 + 0.06*4 + 0.27*2 + 0.09*4 = 2.42$$

哈夫曼编码能得到最优编码☺

哈夫曼编码 (3)

哈夫曼编码算法

输入—— C : n 个元素及其权值的集合

输出—— T : 哈夫曼树

步骤:

$Q \leftarrow \text{PriorityQueue}(C)$ //对集合 C 按照元素权值初始化最小堆优先队列 Q

$T.weight \leftarrow 0, T.left \leftarrow \emptyset, T.right \leftarrow \emptyset$ //初始化哈夫曼树

For $i = 1$ To $n - 1$ Do

$T.left \leftarrow$ 中权值最小的元素 X

$Q.DeleteMin(X)$ //从优先队列 Q 中删除权值最小的元素 X

$T.right \leftarrow$ Q 中权值最小的元素 Y

$Q.DeleteMin(Y)$ //从优先队列 Q 中删除权值最小的元素 Y

$T.weight \leftarrow X.weight + Y.weight$

$Q.Add(T)$ //将 T 插入到优先队列 Q 中, 权值为 $T.weight$

End For

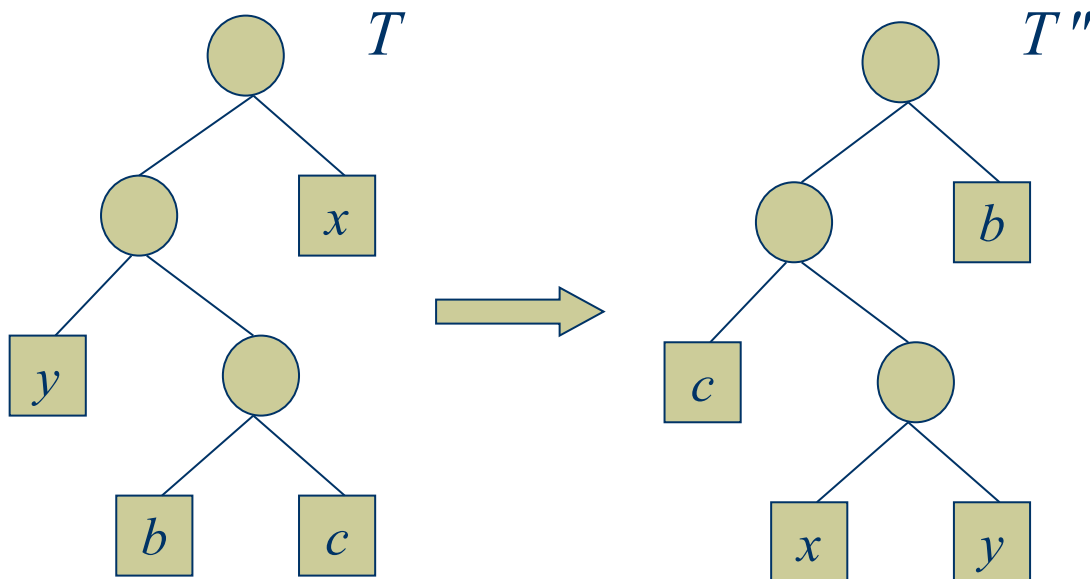
Return T

- ◆ 初始构造优先队列的执行时间为 $O(n)$
- ◆ 优先队列 Q 的插入和删除操作的计算时间为 $O(\log n)$
- ◆ 二叉树的 $(n - 1)$ 次迭代构造时间复杂度为 $O(n \log n)$
算法时间复杂度为 $O(n \log n)$

哈夫曼编码 (4)

◆ 哈夫曼编码算法正确性证明的思路

对最优前缀码二叉树 T 作修改得 T'' ， T'' 表示对 C 做出贪心选择得到的最优前缀码， x 和 y 是 T'' 中最深叶子且为兄弟（树 T'' 与 T 具有相等的平均码长）。



• T 中:

- $f(b) \leq f(c)$

$f(x) \leq f(y)$

- x 和 y 是具有最小频率的两个字符

$f(x) \leq f(b)$

$f(y) \leq f(c)$

• T'' 是对 C 做出贪心选择的前缀编码树

提纲

- ◆ 应用背景和动机
- ◆ 贪心算法的基本思想
- ◆ 哈夫曼编码
- ◆ 总结

总结

- ◆ 应用背景和动机
- ◆ 贪心算法的基本思想和关键
 - 贪心选择性质
 - 最优子结构性质
 - 贪心选择标准
- ◆ 贪心算法的重要实例：哈夫曼编码
- ◆ 贪心算法的正确性证明思路：数学归纳法



结语

谢谢！