

## 算法设计与分析作业 03\_\_ 动态递归

正心 44

Prof. Hong Gao

TA: 郭环宇

发布日期: 2019/03/27

截止日期: 2019/04/03

作业提交 PDF 版本, 作业命名格式为学号 \_ 姓名 \_xx 次作业 \_version.pdf, 代码单独提交任一编程语言文件 (\*.c \*.cpp \*.h \*.cs \*.java \*....), 与作业一起打包并命名为学号 \_ 姓名 \_XX 次作业 \_version.xxx, 发送至邮箱 DBLB\_2019algorithm@163.com

### Exercise 1. (20)

(自学 ppt 最优二叉查找树) 假设我们要设计一个程序, 来实现英语文本到法语的翻译。我们可以建立一个二叉查找树, 将  $n$  个英语单词作为关键字, 对应的法语单词作为关联数据。然后我们依次遍历二叉查找树中每个节点的英文单词, 来找到合适的单词进行翻译工作。这里我们肯定是希望这个搜索的时间越少越好, 如果只考虑单纯的时间复杂度, 我们可以考虑采用红黑树或其他平衡二叉查找树, 达到每个单词搜索时间复杂度  $O(\log(n))$ 。但是对于英语单词而言, 每个单词出现的概率是不一样的, 比如 "the" 等单词。显然的让这类单词越靠近根结点越能减少搜索的时间。而对于某些很少见的单词 mycophagist, 则可以让它们远离根结点, 而且有些英文单词可能没有法语翻译, 这些单词不会出现在二叉查找树中。问题来了, 假如我们已知  $n$  个不同关键字  $k_i$  以及它们的出现概率  $p[i]$ , 我们如何来组成这样一颗二叉查找树, 使得所有搜索访问的节点数目最小呢?

给出二叉搜索树结构如下:

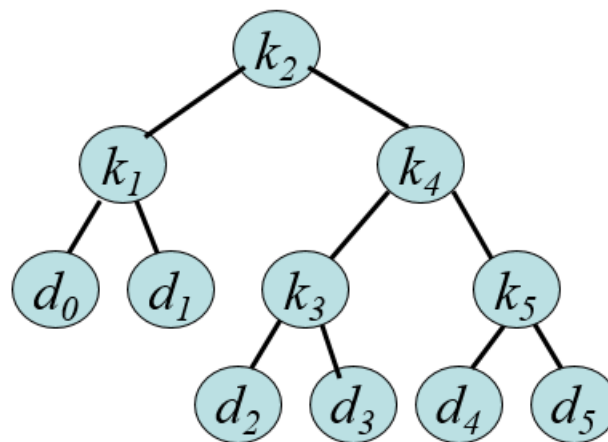
### • 二叉搜索树 $T$

#### — 结点

- $K = \{k_1, k_2, \dots, k_n\}$
- $D = \{d_0, d_1, \dots, d_n\}$
- $d_i$  对应区间  $(k_i, k_{i+1})$   
 $d_0$  对应区间  $(-\infty, k_1)$   
 $d_n$  对应区间  $(k_n, +\infty)$

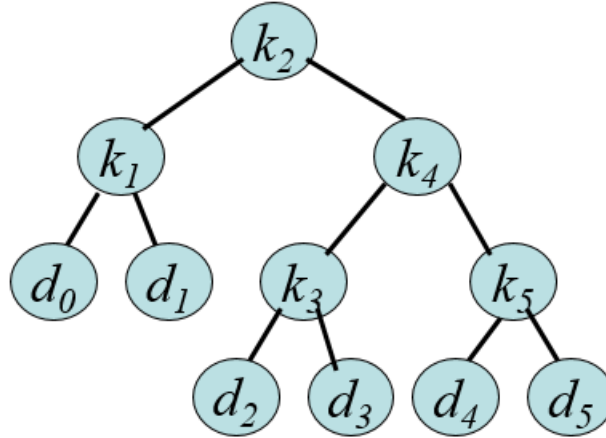
#### — 附加信息

- 搜索  $k_i$  的概率为  $p_i$
- 搜索  $d_i$  的概率为  $q_i$



$$\sum_{i=1}^n p_i + \sum_{j=0}^n q_j = 1$$

• 搜索树的期望代价



$$E(T) = \sum_{i=1}^n (DEP_T(k_i) + 1) p_i + \sum_{j=0}^n (DEP_T(d_j) + 1) q_j$$

问题定义:

输入:

$K = \{k_1, k_2, \dots, k_n\}, k_1 < k_2 < \dots < k_n$

$P = \{p_1, p_2, \dots, p_n\}, p_i$  为搜索  $k_i$  的概率

$Q = \{q_0, q_1, \dots, q_n\}, q_i$  为搜索值  $d_i$  的概率

输出: 构造  $K$  的二叉搜索树  $T$ , 最小化在  $T$  内一次搜索的期望代价

$$E(T) = \sum_{i=1}^n (DEP_T(k_i) + 1) p_i + \sum_{j=0}^n (DEP_T(d_j) + 1) q_j$$

(a) 请描述一颗最优二叉查找树的结构;(6)

(b) 建立优化代价的递归方程;(6)

(c) 分析一颗最优二叉查找树的期望搜索代价;(8)

(c.1) 给出最优二叉查找树算法伪代码 Optimal-BST(p, q, n)(4)

(c.2) 分析给出算法的复杂度 (4)

**Exercise 2.** (9)

字符串的编辑距离, 又称为 Levenshtein 距离, 由俄罗斯的数学家 Vladimir Levenshtein 在 1965 年提出。是指利用字符操作, 把字符串  $A$  转换成字符串  $B$  所需要的最少操作数。

例如对于字符串 "if" 和 "iff", 可以通过插入一个 'f' 或者删除一个 'f' 来达到目的。

问题描述:

给定两个单词 word1 和 word2, 计算出将 word1 转换成 word2 所使用的最少操作数。

你可以对一个单词进行如下三种操作:

插入一个字符  
删除一个字符  
替换一个字符

example1:

输入: horse ros

输出: 3

解释:

horse  $\rightarrow$  rorse (将'h' 替换为'r')

rorse  $\rightarrow$  rose (删除'r')

rose  $\rightarrow$  ros (删除'e')

example2:

输入: intention execution

输出: 5

解释:

intention  $\rightarrow$  inention (删除't')

inention  $\rightarrow$  enention (将'i' 替换为'e')

enention  $\rightarrow$  exention (将'n' 替换为'x')

exention  $\rightarrow$  exection (将'n' 替换为'c')

exection  $\rightarrow$  execution (插入'u')

(a) 请描述最小编辑距离的最优子结构;(6)

(b) 建立优化代价的递归方程;(6)

(c) 输入输出如上, 编程实现最小编辑距离算法;(8)

(d) 分析给出算法的复杂度;(2)

### Exercise 3. (6)

0/1 Knapsack Problem

假设我们有  $n$  件物品, 分别编号为  $1, 2 \dots n$ 。其中编号为  $i$  的物品价值为  $v_i$ , 它的重量为  $w_i$ 。为了简化问题, 假定价值和重量都是整数值。现在, 假设我们有一个背包, 它能够承载的重量是  $W$ 。现在, 我们希望往包里装这些物品, 使得包里装的物品价值最大化, 那么我们该如何来选择装的东西呢?(优化解的结构分析与课题 ppt 相同)

建立与课题 ppt 不同的优化代价的递归方程;(6)

### Exercise 4. (8)

(编程实现) 格雷编码

格雷编码是一个二进制数字系统, 在该系统中, 两个连续的数值仅有一个二进制的差异。

给定一个非负整数  $n$ , 表示该代码中所有二进制的总数, 请找出其格雷编码顺序。一个格雷编码顺序必须以 0 开始, 并覆盖所有的  $2^n$  个整数。

PS: 对于给定的  $n$ , 其格雷编码顺序并不唯一。

当  $n = 2$  时, 根据上面的定义,  $[0, 1, 3, 2]$  和  $[0, 2, 3, 1]$  都是有效的格雷编码顺序。

Sample Input:

1

Sample Output:

$[0, 1]$

Sample Input:

2

Sample Output:

[0, 1, 3, 2]

PS: 0 - 00

1 - 01

3 - 11

2 - 10