

简书

首页

下载APP

搜索



Aa



登录

注册

写下你的评论...

评论0

赞2



后缀树 (suffix tree & array)



ak1947

关注

0.251

2018.12.19 17:42:20 字数 952 阅读 494

**定义：**后缀数组 (suffix array) 是将字符串的所有后缀进行排序放入数组中。后缀树 (suffix tree) 则是所有后缀形成的字典树 (trie) 的一种压缩表示。后缀数组相对后缀树来说，使用的存储空间更小（只用保存原始字符串和一个长度相同的整数数组）。

后缀树在字符串的很多算法（例如查找，匹配，最长公共子串等）中有广泛应用，是一种非常实用的数据结构。

**例：**对于单词banana，有如下后缀数组表示：

index	suffix	排序后index	排序后suffix
0	banana	5	a
1	anana	3	ana
2	nana	1	anana
3	ana	0	banana
4	na	4	na
5	a	2	nana

后缀数组的表示为: [5,3,1,0,4,2], 其中数字表示后缀首字母再整个字符串中的位置（即从0开始的下标）。

使用最“笨”的方法可以在  $O(n^2 \log n)$  时间复杂度内构建后缀数组，其中  $n$  是字符串长度。思路是使用  $O(n \log n)$  的排序算法对后缀字符串排序，同时保持后缀起始字符下标。因为两个字符串比较大小需要  $O(n)$  复杂度，所以整体复杂度为  $O(n^2 \log n)$ 。

本文介绍一种  $O(n \log_2^n)$  的后缀数组构建算法。简单期间，首先考虑  $O(n \log n \cdot \log n)$  复杂度的算法。出发点是充分利用所有后缀字符串都是来源于一个字符串的特点，同时借鉴radix排序思路。算法由  $k = \lceil \log_2^n \rceil$  轮排序完成，第1轮排序只对第1个字符排序，第2轮排序对前2个字符排序，第3轮排序对前4个字符排序，第k轮排序对前  $2^k$  个字符排序。注意这里有一点很重要，如果所有后缀字符串已经按照前  $2^i$  字符排序，则可以使用  $O(n \log n)$  时间按照前  $2^{i+1}$  排序，原因是**两个suffix可以在常数  $O(1)$ （而不是  $O(n)$ ）时间内比较大小**。[后缀数组](#)

**例：**构建字符串banana的后缀数组

为每个suffix分配一个序号rank，例如，对第*i*个字符串分配rank为 `str[i] - 'a'`。这样，得到下面rank表：

index	suffix	rank
0	banana	1
1	anana	0
2	nana	1
3	ana	2
4	na	3
5	a	4

评论0 赞2 ...



ak1947

关注

总资产1 (约0.11元)

活动/任务选择问题

阅读 188

现金流问题

阅读 51

推荐阅读

docker轻量级管理工具

阅读 3,577

分类树

阅读 50

生物基础知识-CDS、ORF、启动子、终止子、转录因子、UTR

阅读 8,983

数量遗传学11-选择指数和多性状选择

阅读 49

Transformers 示例 | 六

阅读 68

简书

首页

下载APP



Aa



登录

注册

评论0 赞2 ...