文章编号:1674－2869(2015)04－0056－03

压缩编码的上下文树构造算法

付　敏[1,2],戴祖旭[1],王道蓬[2]
1 武汉工程大学理学院,湖北 武汉 430502;2 华中科技大学图像识别与人工智能研究所多谱图像信息处理国防重点实验室,湖北 武汉 430074

摘　要:上下文树是构造无算压缩算法的一种重要基础,作为信息处理过程分析随机序列统计特性的常用数据结构,随机序列中的符号来自于某个固定的符号集合. 上下文树一般是一棵 n 元树,其中 n 大于 1,但是树是一种占用计算机内存较多的数据结构,因此提出了基于压缩编码的上下文树构造算法,根据符号的一阶统计特性对符号做二进制的压缩编码,用二元树代替 $n(n>2)$元树,在相同内存的存储空间下,可以大大增加树的高度. 计算机数值实验表明基于压缩编码的上下树构造对子串做出了更大长度的相关性检测,并且提高了数据分析的精度.

关键词:上下文树;n 元树;压缩编码
中图分类号:TP309.7　　**文献标识码**:A　　**doi**:10. 3969/j. issn. 1674-2869. 2015. 04. 012

0　引　言

信息处理技术的飞速发展使得算法应用越来越为广泛,在图像处理[1],路径搜索优化[2-3]以及工程应用等领域,数学形态学越来越重要. 作为其中的一种重要的数据结构,上下文树是信息处理过程中分析随机序列统计特性的常用工具. 利用上下文树获取序列中字符串[4-5]的频率分布,可以构造高效的数据压缩算法[6-7],通过上下文树分析文本或生物 DNA 序列中符号的相关性,可以准确实现文档或 DNA 序列的自动分段[8-9]. 同时,上下文树也是研究可变长 Markov 链的主要数学工具,用于随机序列发生器设计[10-11].

随机序列中的符号来自于某个固定的符号集合,比如英文文档包括 26 个字母,DNA 序列由 A、C、G、T 组成,它们分别代表组成 DNA 的四种核苷酸一腺嘌呤、胞嘧啶、鸟嘌呤和胸腺嘧啶.因此上下文树一般是一棵 n 元树,其中 n 大于 1. 树是一种占用计算机内存较多的数据结构,系统内存容量会直接影响树的高度,而树高决定了分析过程中序列子字符串的长度,会直接影响符号串统计特性的精度.

本文提出了基于压缩编码的上下文树构造算法,根据符号的一阶统计特性对符号做二进制的压缩编码,用二元树代替 $n(n>2)$元树,在相同内存的存储空间下,可以大大增加树的高度,对子串做更大长度的相关性检测,提高分析的精度.

1　上下文树

设 $A=\{a_1,a_2,\cdots,a_n\}$ 是一个符号集合,随机序列 $x=x_{-N}x_{-N+1}\cdots x_{-1}x_0x_1\cdots x_Mx_i\in A$,$i\in I$ 从序列 x 中任取一点,比如 x_0,向左回溯,找到 $x=x_{-J}x_{-J+1}\cdots x_{-1}x_0$,其中 $0\leqslant J\leqslant N$,使得条件概率为

$$P(x|x_{-J}x_{-J+1}\cdots x_{-1}x_0)=P(x|x_{-J-1}x_{-J}x_{-J+1}\cdots x_{-1}x_0) \quad (1)$$

成立, 设 $x=x_{-j}x_{-j+1}\cdots x_{-1}x_0$ 为 x_1 的上下文环境. x_1 的上下文环境说明了影响符号 x_1 的出现的历史追溯到 x_{-J} 即可,不必再回溯到 x_{-J-1} 以及更早的符号. 要验证式(1)中的 n 个等式,需要根据序列 x 统计出字符串 $x_{-j-1}x_{-j}\cdots x_{-1}x_0x$ 和 $x_{-j}x_{-j+1}\cdots x_{-1}x_0x$ 出现的频数. 统计频数的工作由如下的上下文树算法完成.

算法 1:上下文树构造算法:

(1)初始化根节点,符号计数设置为 0.

(2) 假设根据字符串 $x^t=x_1x_2\cdots x_t$ 构造出树 T_t,当前输入符号为 x_{t+1},从根节点出发,按照字符串 $x_tx_{t-1}\cdots$的指引,访问 T_t 的结点,直到 x_t 的符号用完,或者到了叶子结点. 对每个访问到的结点,将该结点处的符号 x_{t+1} 的计数增加 1.

收稿日期:2015-01-12
基金项目:湖北省自然科学基金重点项目(2010CDA009);湖北省自然科学基金一般项目(2009CDB367);国家自然科学基金面上项目(61175013);武汉工程大学校级教研项目(X2013021).
作者简介:付　敏(1979-),女,湖北襄阳人,讲师,博士研究生.研究方向:信息处理,计算机视觉.

(3) 如果最后一次访问到的结点 x_{t+1} 处的计数≥2,创建一个新结点 $x_t\cdots x_{t-j+1}x_{t-j}$,新结点$x_{t+1}$ 处的计数置为 1,其他符号的计数置为 0,得到树 T_{t+1}.

从上下文树构造算法可知,树 T_t 是一棵 n 元树,其高度随着从左到右扫描序列的时刻 t 增加而增加. n 元树结点个数有如下性质:

命题 1:高度为 h 的 n 元满树第 l 层结点数为 n^l 个,全部节点数为$\frac{n^{h+1}-1}{n-1}$个.

命题 2:高度为 h 的 n 元满树的长度为 $l(l\leqslant h)$的字符串个数为 n^l 个.

2　基于压缩编码的上下文树算法

由于随机序列中符号出现的概率是非均匀分布的,利用熵压缩编码的原理可以对符号进行二进制变长压缩编码,二进制符号作为码元符号,随机序列中的自然语言符号可编码为变长的二进制串. 本文在构建上下文树的时候,树的结点不是自然语言符号,而是仅含 0,1 的二元树,每个自然语言符号都对应着二元树上的一段路径.

算法 2:基于压缩编码的上下文树算法.

(1)统计随机序列的符号的频率,构造 Huffman 编码,形成码表,用 $B(x_i)$表示自然语言符号 x 对应的二进制串.

(2)初始化根结点,符号计数置为 0,令 t 时刻自然语言符号序列为 $x_t=x_1x_2\cdots x_t$.

(3) 假设根据字符串 $x_t=x_1x_2\cdots x_t$ 构造出树 T_t,当前输入为 x_{t+1},从根结点出发,按照字符 $x_tx_{t-1}\cdots$的指引,访问 T_t 的结点,直到 x_t 的符号用完,或者到了叶子结点,比如 $B(x_t)\cdots B(x_{t-j+1})$. 对每个访问到的结点,将该结点处符号 x_{t+1} 的计数增加 1.

(4) 如果最后一次访问到的结点 x_{t+1} 处的计数≥2,创建一个新结点 $B(x_t)\cdots B(x_{t-j+1})B(x_{t-j})$,新结点处 x_{t+1} 的计数置为 1,其他符号的计数置为 0,得到树 T_{t+1}.

基于压缩编码的上下文树算法与上下文树算法不同点在于, 将 n 元树化为二元树,n 元树中的每个结点在二元树中扩展为一条路径. 该路径对应着自然语言符号的 Huffman 编码. 下面的结论表明,将 n 元树化为二元树,在表达相同数量的自然语言字符串情况下占用的内存却大大减少了.

命题 3:设 n 元树的每个节点指针域占用字节数为 d,数据域占用字节数为 m,对 n 个符号采用二进制编码,平均码长为 l_0.则高度为 h 的 n 元满树所占用字节总数

$$(nd+m)\frac{n^{h+1}-1}{n-1}>(2d+m)(2^{hl+1}-1) \qquad (2)$$

证明: 由熵编码原理可知平均码长 $l_0\leqslant\log_2^n$,因此 $n^h\geqslant 2^{l_0}$,下面证明

$$(nd+m)\frac{n^{h+1}-1}{n-1}>(2d+m)(2^{hl+1}-1)$$

表 1　3 个长度为 62 的字符串 1-62 阶条件概率计算结果范例

Table 1　Computational results of 3 stings′ conditional probability

String1:　Askacornjerkoffniggersanbigoledumbwhiteboystheyaintjussettinth

Results:

0.088 34　0.078 437 7　0.020 491 8　0.325　0.769 23　1　1　1　1　0.4　0.5　1　1　1

1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1

1　1　1　1　1　1　1　1　0.5　1　1　1　1　1　1　1　1　1　1　1　1　1

1　1　1　1　1

String2: askacornjerkoffniggersanbigoledumbwhiteboysintheworldstartgrap

Results:

0.088 34　0.078 437 7　0.020 491 8　0.325　0.769 231　1　1　1　0.4　0.5　1　1　1　1

1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1

1　1　1　1　1　1　1　0.5　1　1　1　1　1　1　1　1　1　1　1　1　1　1

1　1　1　1

String3: inceweplayedthemnebraskacornshuckerjackoffsintheorangebowlhego

Results:

0.070 715 3　0.304 453　0.026 051 1　0.348 101　0.2　0.909 091　0.2　1　1　1　1　1　1　1　1

1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1

1　1　1　1　1　1　0.5　1　1　1　1　1　1　1　1　1　1　1　1　1　1　1

1　1　1

即$(nd+m)(n^{h+1}-1)>(n-1)(2d+m)(2n^h-1)$

$dn^{h+2}+mn^{h+1}-nd-m>(2d+m)(2n^{h+1}-2n^h-n+1)$展开移项得到：

$dn^{h+2}+(m-4d-2m)n^{h+1}+(4d+2m)n^h-nd-m+4dn+mn-2d-m>0$

$dn^{h+2}+(-4d-m)n^{h+1}+(4d+2m)n^h+(m+d)(n-2)>0$

由于$(m+d)(n-2)\geqslant 0$

令 $dn^{h+2}-(4d+m)n^{h+1}+(4d+2m)n^h\geqslant 0$ (3)

除以 n^h 得到：

$$dn^2-(4d+m)n+(4d+2m)\geqslant 0 \quad (4)$$

$$\Delta=(4d+m)^2-4d(4d+2m)=m^2>0$$

故关于 n 的不等式(4)有解

$$n\leqslant 2 \text{ 或 } n\geqslant 2+\frac{m}{d}$$

也即 $n\geqslant 2$ 时结论成立.

3 实验结果

利用基于压缩编码的上下文树构造算法对英文长篇小说《Forrest Gramp》(Winston Groom, 1986)开展了统计工作,程序运行环境为 32 位微软 XP service pack3 操作系统,Pentium Dual-Core E6700 CPU,主频 3.20 GHz,内存 2 GByte 分析字符串长度达到 62 个自然语言符号时,运行时间约 34 分钟, 统计出共 281 705 条长度为 62 的字符串,每个字符串分别计算出从 1 阶到 62 阶条件概率以供后续分析使用. 表 1 给出了其中的 3 个计算结果.

从数据统计分析可以看出,符号的一阶统计特性可以在数值实验中得到,对文本信息的符号做二进制的压缩编码,在结构处理上采用二元树代替(>2)元树,在相同内存的存储空间下,理论上证明可以增加树的高度,实验也进一步验证结论. 基于此,可以对长度更大的子串进行相关性检测,并且提高分析的精度.

参考文献：

[1] 洪汉玉,章秀华,程莉.道路病害形态特征的图像分析[J].武汉工程大学学报,2014,36(4):70-76.
HONG Han-yu,ZHANG Xiu-hua,CHENG Li. Image analysis method for road disease morphology characteristic[J]. Journal of Wuhan Institute of Technology, 2014,36(4):70-76.(in Chinese)

[2] 孙玉昕, 章瑾.利用堆排序优化路径搜索效率的分析[J].武汉工程大学学报,2013,35(10):50-55.
SUN Yu-xin, ZHANG Jin.Practical analysis of improving path searching efficiency by heap sort[J].Journal of Wuhan Institute of Technology, 2013,35(10): 50-55.(in Chinese)

[3] 王学华,刘莉君,马凡杰,等.数控激光加工路径链表快速搜索优化[J].武汉工程大学学报 ,2014,36(10): 52-57.
WANG Xue-hua,LIU Li-jun,MA Fan-jie et al. Rapid routine searching of numerical control laser processing based on linked list structure[J]. Journal of Wuhan Institute of Technology,2014,36(10):52-57.(in Chinese)

[4] 徐超,周一民,沈磊.一种面向隐含主题的上下文树核[J].电子与信息学报,2010, 32(11):2695-2700.
XU Chao,ZHOU Yi-min, SHEN Lei.A context tree kernel based on latent semantic topic[J]. Journal of Electronics & Information Technology,2010,32(11): 2695-2606.(in Chinese)

[5] RISSANEN J. A universal data compression system[J]. IEEE Transactions on information theory, 1983, 29(5): 656-664.

[6] 陈亮,孟庆愿, 董彦磊,等.CTW 无损压缩算法在管道无损检测中的应用[J].实验技术与管理, 2012,29(6):42-47.
CHEN Liang,MENG Qingyuan,DONG Yanlei,et al. Using CTW lossless compression algorithm in pipelines nondestructive testing[J]. Experimental Technology and Management,2012,29(6):42-47.(in Chinese)

[7] DUMONT Thierry.Context tree estimation in variable length hidden[J]. IEEE Transactions on information theory, 2014, 60(6): 3196-3208.

[8] GWADERA R, GIONIS A, MANNILA H. Optimal segmentation using tree models [J]. Knowledge and Information Systems, 2008, 15(3): 259-283.

[9] Martins D A, Neves A J R, Pinho A J. Variable Order Finite-Context Models in DNA Sequence Coding[C]// Pattern Recognition and Image Analysis. Springer Berlin Heidelberg, 2009: 457-464.

[10] BÜHLMANN P. Model selection for variable length Markov chains and tuning the context algorithm[J]. Annals of the Institute of Statistical Mathematics, 2000, 52(2): 287-315.

[11] CÉNAC P, CHAUVIN B, PACCAUT F, et al. Context trees, variable length Markov chains and dynamical sources[C]//Séminaire de Probabilités XLIV. Springer Berlin Heidelberg, 2012: 1-39.

(下转第 64 页)