Data Compression学习笔记一：Golomb编码

mylq
悬想何益？但不忘栽培之功，怕没有枝叶花实？

前置知识：一元编码（Unary Coding）

一元编码(Unary coding)是一种简单的只能对非负整数进行编码的方法，对于任意非负整数num，它的一元编码就是num个1后面紧跟着一个0，或者num个0后面紧跟着一个1，具体哪种情况需要协议的约定，如无特殊约定，一般默认使用第一种。

num	unary coding
0	0
1	10
2	110
3	1110
4	11110
5	111110

Golomb编码基本原理

Golomb编码是一种基于游程编码（run-length encoding,RLE）的无损的数据编码方式，当待压缩的数据符合几何分布（Geometric Distribution）时，Golomb编码取得最优效果。

举个游程编码的例子，如以下的待编码的二进制串，

> 00000100110001010000001110100010000010001001000110100001001

该串含有18个游程，5, 2, 0, 3, 1, 6, 0, 0, 1, 3, 5, 3, 2, 3, 0, 1, 4, 和2。其平均值是(5+2+0+3+1+6+0+0+1+3+5+3+2+3+0+1+4+2)/18 ≈ 2.28；对游程长度进行排序，获得其中位数2。

> 0, 0, 0, 0, 1, 1, 1, 2, 2, 2, 3, 3, 3, 3, 4, 5, 5, 6

我们用p来表示二进制串中0出现的概率，那么1出现的概率就是1-p；显然，较大的p表明有更高的概率出现长游程，较小的p暗示长游程数量较少。

对于待编码的非负整数n，Golomb码会选择一个参数m来对整数n进行编码。参数m的选择取决于概率p和游程长度的中位数。

为了获得非负整数n的Golomb编码，我们需要根据选择的m计算三个数据：商q（quotient），余数r（remainder），和c。

$$q=\left\lfloor \frac{n}{m} \right\rfloor,\quad r=n-qm,\quad c=\lceil \log_2 m \rceil$$

Golomb编码由两部分构成，第一部分是使用一元编码（Unary）表示的商q；第二部分是使用特殊方式编码的二进制值余数r。余数r的取值可能是

> 0, 1, 2, 3,……,m-2,m-1

对于待编码的非负整数n而言，其拥有的余数个数不大于m；我们注意到 $2^c-m \le 2^{c-1}$ 显然成立，所以对于其前 2^c-m 个余数，它们以无符号整数的方式进行编码，并可以存储在c-1个比特位中。剩下的余数以无符号整数的方式编码，并存储在c个比特位中（以最大的c比特整数结束）。当m等于2的整数次幂（ $m=2^c$ ）时比较特殊，此时不需要任何(c-1)位的编码(由Robert F.Rice发展，因此也被称为Rice码)。

我们知道 $n=r+qm$；所以对于任何已知参数m的Golomb编码，我们可以使用商q和余数r轻易地重构出n。

Golomb编码示例

选择 $m=3$,可得 $c=2$,和余数0，1，2。我们计算出 $2^c-3=1$,所以第一个余数0被编码在 $c-1=1$ 个比特位中。剩余的余数被编码在2个比特位中，分别为 10_2 和 11_2 。

选择 $m=5$,可得 $c=3$,和余数0，1，2，3，4。我们计算出 $2^c-5=3$,所以前三个余数0，1，2被编码在 $c-1=2$ 个比特位中，分别为 00_2 ， 01_2 ， 10_2 。剩余的余数被编码在3个比特位中，分别为 110_2 和 111_2 。

下表显示了 m、c、2^c-m 的一些示例。

m	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
c	1	2	2	3	3	3	4	4	4	4	4	4	4	4	4
2^c-m	0	1	0	3	2	1	0	7	6	5	4	3	2	1	0

下表显示了一些Golomb编码的示例。

m/n	0	1	2	3	4	5	6	7	8	9	10	11	12
3	0|0	0|10	0|11	10|0	10|10	10|11	110|0	110|10	110|11	1110|0	1110|10	1110|11	11110|0
5	0|00	0|01	0|10	0|110	0|111	10|00	10|01	10|10	10|110	10|111	110|00	110|01	110|10

Golomb编码适用范围

显然，当m较小时，Golomb编码码长开始很短，但随着n增长，码长迅速增加；这适用于0出现的概率p较小的游程编码，即只有很少的长游程。当m较大时，Golomb编码的初始码长较长（对于n=1，2...），但是随着n增长，其码长增加速度较慢；这适用于0出现的概率p较大的游程编码，即存在较多的长游程。

解码

以m=16为例，解码时，先依次读取第一个0前面的1，并对其进行计数得到A，则编码长度为A+c+1,(对于m=16，即A+5位)。若我们将该段编码最右端5位用R来表示，那么这段编码的值就是16A+R。

当m不是2的整数次幂时，解码器需要做更多的工作。首先移除A个值为1的比特位和紧随其后的一位值为0的比特位，接着我们将其后的c-1位比特表示为R。如果 $R<2^c-m$,那么编码长度为A+1+(c-1)，并且其值是m*A+R。如果 $R\ge 2^c-m$ ，那么编码长度为A+1+c，并且其值是 $m*A+R_l-(2^c-m)$,其中 R_l 是由R及其后一位比特组成的c比特整数。

m的选择

最佳的m取决于p，当平均编码长度最短，可以证明其是最接近 $-\frac{1}{\log_2 p}$ 的整数，即其值满足以下条件：

$$p^m \approx \frac{1}{2}$$

更进一步，可以获得最佳的m的值为

$$m=\left\lceil -\frac{\log_2(1+p)}{\log_2 p} \right\rceil.$$

当p未知时，可以采用根据当前输入调整m的自适应算法Goladap，这里不再赘述。

编辑于 2020-08-12 07:28

编码 数据压缩 视频编码

写下你的评论...

2 条评论
默认
最新

威仔
哈夫曼编码啥时候出来
2020-08-11
回复 喜欢

mylq
作者

关注一下，你就知道了
2020-08-11
回复 喜欢

文章被以下专栏收录

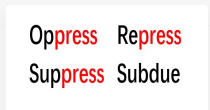
学习笔记
计算机视觉、视频编解码、图像处理等相关领域学习笔记

推荐阅读

如何在FPGA中实现高效的compressor加法树
moon 发表于AI加速

Compressed suffix array-[Succinct data structure]
王素贞 发表于Succi...

搞不懂后期制作的COMPRESSOR压缩效果器...
混音师阿辉 发表于第一次混音...



oppress、repress、suppress和subdue的区别
小菜狗

×
登录即可查看 超5亿 专业优质内容
超 5 千万创作者的优质提问、专业回答、深度文章和精彩视频尽在知乎。
立即登录/注册

赞同 12 2 条评论 分享 喜欢 收藏 申请转载 ...