CRC 算法的研究①

杨梅娟 尹德春
(西华师范大学计算机学院 南充 637002)

摘 要:CRC 是目前网络中普遍采用的一种检测码,介绍了 CRC 的理论基础、冗余位的产生方法、实现方式。在简要分析的基础上,给出了软件实现算法的 Delphi 程序代码。

关键词: 循环冗余校验(CRC) 生成多项式 Delphi
中图分类号:TP301.6

Research on CRC's Calculating Method

Yang Meijuan Yin Dechun
(College of Computer Science, China West Normal University, Nanchong 637002)

Abstract: CRC is an error-detection code widely used in the computer network. This article illustrates its theoretical foundation, production of redundant codes as well as means of implementation. After giving briefly discussion on the CRC, a program with Delphi is made out.

Key words: cyclic redundancy check, generator polynomial, Delphi
Class number: TP301.6

数据通信中,为确保高效而无差错地传送数据,必须对数据进行校验,在接收方为了检测所接收的数据是否有误码,可以采用多种检测方法,其中循环冗余校验 CRC(Cyclic Redundancy Check)算法是一种对传送数据块进行校验的高效的差错控制方法。CRC 在计算机通信、网络协议、文件压缩程序中得到了广泛的应用,而 CRC 码的生成可以通过硬件实现,也可以通过软件实现。

1 CRC 校验算法的主要思想

发送端计算机运用 CRC 算法计算出待发送数据的 CRC 校验码,并附加在待发送数据的末尾,即在发送数据的同时增加 CRC 码(编码过程)。发送后,接收端计算机检测数据和 CRC 码之间的数学关系是否正确(译码过程),若不正确则说明数据信息在传输过程中有误码,通常通过发送端的重传来校正错误,直至传送正确为止。

2 CRC 校验的数学原理

假设在一个长度为 n 的码组中,有 k 个信息位和 n - k 个校验位,其中所传输的数据系列可看为高次(k 次)多项式,也称为位序列多项式,用 D(x)表示在传输前,将 D(x)用预先规定的生成多项式 P(x)去除,再将其余式 R(x),即余数码(Block Check Code,简称 BCC 码)附加在所传输数据的尾部一并传送。在接收方,用同样的生成多项式 P(x)去除,若除得的余式值为零,可判断出所接收的数据是正确的,若余式非零,说明发生了错误。

CRC 校验的数学原理[1],[2]及编码规则步骤如下:

(1)信息位的多项式为 $D(x) = d_{k-1}x^{k-1} + d_{k-2}x^{k-2} + \cdots\cdots + d_2x^2 + d_1x^1 + d_0x^0$ 这里 $d_i = 0, 1 (i = 0, 1, \cdots, k-1)$。

(2)将欲传送的序列 D(x)乘以 x^{n-k},其中 n - k 为余数码 BCC 的位数,也即将被校验的序列数据左移 n - k 位,在信息位后面附加 n - k 个"0",这样做的目的是空出 n - k 位,以便拼装将来求得的 n - k 位余数。

(3)将 $x^{n-k}D(x)$ 的积用生成多项式 P(x)去除(注意:是进行模 2 除法),忽略其商 Q(x),将其余

① 收到本文时间:2005 年 5 月 8 日

数 R(x) (BCC) 取出,即 $x^{n-k}D(x)/P(x) = Q(x) + R(x)/P(x)$ 式中 R(x) ——余式(n - k 位); Q(x) ——倍式。

需要指出的是:在模运算中,模 2 加减法的结果是相同的,都可以用逻辑 XOR 运算完成,模 2 乘法运算可以用逻辑 AND 运算完成,模 2 除法只在除数为 1 时有效,并且结果维持不变,也可以看作是 XOR 运算。在进行长除法时要采用模 2 除法,模 2 除法就是计算过程中采用模 2 减法(XOR 运算)的除法,减法过程没有进位和借位,因此 BCC 的位数一定总是比 P(x) 少一位。

(4) 输出码多项式为 $x^{n-k}D(x) + R(x)$。

3　生成多项式的种类

并不是任何一个多项式都可以作为生成多项式的,从检错与纠错的要求出发,生成多项式必须满足:任一位发生错误都应使余数不为 0;不同位发生错误应当使余数不同;应满足余数循环规律。

常用的生成多项式举例如下(其中 x 值等于 2):

(1) CRC - 9 ,$P(x) = x^9 + x^6 + x^5 + x^4 + x^3 + 1$, BCC 由 9 位组成。

(2) CRC - 12 ,$p(x) = x^{12} + x^{11} + x^3 + x^2 + 1$,BCC 由 12 位组成。

(3) CRC - 16 ,$p(x) = x^{16} + x^{15} + x^2 + 1$, BCC 由 16 位组成。

(4) CRC - CCITT ,$p(x) = x^{16} + x^{12} + x^5 + 1$, BCC 由 16 位组成。

(5) CRC - 32 ,$p(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$, BCC 由 32 位组成。

4　CRC 的实现方式

实现 CRC 的方式常用的有硬件实现和软件实现,本文重点介绍软件的实现。(1) 硬件实现。发送或接收的信息码元都经过 CRC 电路工作,最后所有码元校验完毕,校验电路里的数就是整个信息码元的冗余值。(2) 软件实现。用软件编程来模拟 CRC 原理,把发送和接收的码元都按位运算,最后确定其校验值,这种方法的缺点是校验时间长,速度慢,但优点是占用空间少。生成多项式以 CRC - 32 为例,实现代码如下:

```
function CalCRC32(Data,CRC,GenPoly:DWORD):DWORD;
Var
  i:Integer;
begin
  CRC:=CRC xor Data;//Data 为待处理的输入数据
  for i:=0 to 7 do
  if(CRC and $01)<>0 then //只测试最低位
    CRC:=(CRC shr 1) xor GenPoly;//最低位为 1,移位和异或处理
  else CRC:=CRC shr 1;//否则只移位(除 2)
Result:=CRC;
end
```

通常为了避免上述算法的缺点,也可以使用高效快速的查表算法,预先生成 CRC 码表(256 项),用于查表法的实现,在程序初始化的时候就先调用,预先生成 CRC32TABLE[256]查表数据。参数表的构造方法如下:

```
pocedure InitCRC32Tab(GenPoly:DWORD);
Var
  i:Integer;
begin
  for i:=0 to 255 do
    CRC32TABLE[i]:=CalCRC32(i,0,GenPoly);
end;
```

程序初始化生成参数表的代码:

```
procedure TForm1.FormCreate(Sender:TObject);
var
  GenPoly32:DWord;
  {//CRC - 32 = X32 + X26 + X23 + X22 + X16 + X12 + X11 + X10
 + X8 + X7 + X5 + X4 + X2 + X1 + 1
// 00000100 11000001 00011101 10110111($04C11DB7) 低位先行($EDB88320)}
begin
  GenPoly32:=$EDB88320;
  InitCRC32Tab(GenPoly32);
end;
```

构造好参数表后,可通过查表快速计算,优点是速度快,但缺点是占用空间多,要预先生成 CRC32TABLE[256]查表数据。

详细的实现过程可采用如下的程序段:

```
procedure QuickCRC32(FileName:string;var CRC32:DWORD);
var
  F:file;
  BytesRead:DWORD;
  Buffer:array[1..65521] of Byte;
  i:Word;
begin
  FileMode:=0;
  CRC32:=$ffffffff;
  {$I-}
```

(下转第 69 页)

寻求,保持和预防客户流失等领域。

(3)分类分析

分类分析就是通过分析示例数据库中的数据,为每个类别做出准确的描述或建立分析模型或挖掘出分类规则,然后用这个分类规则对其它数据库中的记录进行分类。在客户关系管理中,可以对客户群体进行分类,根据消费群体或个体的消费属性,把大量的客户分为不同的类。对每一类消费群体采取不同的服务方式,可以提高客户对产品的满意度,进而获取最大的利润。

(4)聚类分析

聚类分析是把一组数据按照相似性和差异性分为几个类别,其目的是使得属于同一类别的数据间的相似性尽可能的大,不同类别数据间的相似性尽可能的小。它可以应用到客户群体的分类,客户背景分析,客户效益分类分析和预测,市场的细分和客户的细分等。

4 结束语

在信息时代,要充分利用企业的信息资源,从以产品为中心的管理模式转变为以客户为中心的管理模式上来,利用数据挖掘技术,分析客户的特征,探索企业和所对应市场的运营规律性,不断提高企业的经济效益是企业发展的必由之路。

参 考 文 献

[1] FayyadUM, Piatetsky - Shapirog, Smythp. Advance in knowledge discover and data mining [M]. California: AAAIPress, TheMITPress, 1966

[2] Cooleyr. Grouping Webpage references into transactions for mining world wide webb rowsing patterns[C]. Proceedings of KDEX, 97. New portBeach, USA, 1997

[3] JiaweiHan, Miche line Kamber. 数据挖掘概念与技术[M]. 北京:机械工业出版社, 2001

[4] 柳炳祥,徐远纯. 数据挖掘在企业危机管理中的应用[J]. 科学与科学技术管理, 2002, (6):78～80

[5] Philip Kotler. Marketing Management: Analysis, Planning, Implementation and Control[M]. Prentice - Hall, Inc. 1997

[6] Alex Berson, Stephen Smith. 构建面向 CRM 的数据挖掘应用[M]. 北京:人民邮电出版社, 2001

(上接第 31 页)

```
    AssignFile(F, FileName);
    Reset(F, 1);
    if IOResult = 0 then
begin
    repeat
    BlockRead(F, Buffer, SizeOf(Buffer), BytesRead);
    for i := 1 to BytesRead do
      CRC32 := (CRC32 shr 8) xor CRC32Table[Buffer[i] xor
(CRC32 and $000000FF)];
    until BytesRead = 0;
    end;
    CloseFile(F);
    { $I+ }
    CRC32 := not CRC32;
end;
```

5 结束语

本文简要介绍了 CRC 的基础理论及其软件实现过程,当需要对数据进行校验时,只须运行相应的 CRC 程序,即可完成校验过程。

参 考 文 献

[1] 瞿中等. CRC 算法在计算机网络通信中的应用[J]. 微机发展, 2002, (2) 12～14

[2] 顾文达等. 快速循环冗余校验算法及其程序实现[J]. 南京理工大学学报, 1995, (4) 113～116

[3] 李清慈. 关于循环冗余校验的分析[J]. 计算技术, 1989, (1)

[4] 顾慰文. 纠错码及其在计算机系统中的应用[M]. 人民邮电出版社, 1980

[5] 马秀莲,李廷芳. 数字通信差错控制技术[M]. 北京:铁道出版社, 1991

[6] 尹晓勇. 计算机网络基础[M]. 北京:电子工业出版社, 1996