DOI:10.13203/j.whugis2008.10.010
第33卷 第10期 武汉大学学报·信息科学版 Vol.33 No.10
2008年10月 Geomatics and Information Science of Wuhan University Oct. 2008

文章编号:1671-8860(2008)10-1026-04 文献标志码:A

非数值型数据的数据库水印算法研究

董晓梅[1] 田跃萍[1] 李晓华[1] 于 戈[1]

(1 东北大学信息科学与工程学院,沈阳市文化路三巷11号,110004)

摘 要:分析了对非数值类型数据进行水印嵌入的特殊性,提出了基于非数值型数据水印的简单替换算法及其改进方法。为了减少水印嵌入对数据统计特性的改变,提出了一个统计特征控制算法,并设计了相关的实验,实验证明了算法的可用性和鲁棒性。

关键词:关系数据库;非数值型数据;数字水印;简单替换;统计特征控制

中图法分类号:TP 309

将数字水印应用于关系数据库数据是最近几年随着数据库技术的发展而兴起的一种数据库数字水印技术,目的是为了实现数据库的版权保护。

2002年,Agrawal和Sion等人分别提出了基于关系数据库的数据库水印算法[1,2]。2003年,牛夏牧等提出了一种可以在关系数据库中嵌入具有实际意义字符串的数据库水印算法[3]。2004年Zhang等提出了一种将图像作为水印信息嵌入载体数据库的数据库水印算法[4]。国内还有其他学者在数据库水印技术的研究中取得了相应的成果[5-7]。

近年来,对关系数据库数字水印的研究主要集中在对数值型数据的研究上[8]。文献[9]提出了一种针对非数值型数据的水印算法,然而,这种算法不易进行水印信息的提取恢复,而且该方法对数据域的统计特征的影响也很不确定。本文针对关系数据库中的非数值数据进行探讨,在此基础上给出了非数值型的数据库水印算法与实现。

1 非数值型数据的水印算法

非数值型数据不能像数值型数据那样进行简单的按位操作或增/减值操作,因此,对于非数值型数据进行水印嵌入时要用同一属性域的其他值来替换原有的数据值。

本文提出了一种简单替换算法,可以针对一个属性嵌入水印信息。然后,将这个算法扩展到多个属性域,并针对出现的问题提出了改进方法。

1.1 简单替换算法

假设关系数据库模式为 $R(P,A)$,P 为主键,水印信息的编码过程中不能对 P 作任何改动,A 为 R 中用来进行水印嵌入的非数值型的属性。R 中共包含 N 个元组,r_1, r_2,…, r_N,各元组的主键分别为 $r_i.P$,元组中的属性为 $r_i.A$。e 为调节因子,控制水印嵌入的比例。K_1、K_2 为数据库拥有者的密钥。水印算法中用到的一些符号及其含义如表1所示。

表1 水印中的符号及其涵义

Tab.1 Notation and Meaning in Watermarking

符号	含意
N	R 中元组数
P	主键
v	R 中可用来嵌入水印的属性列的数量
W	代表数据库特征的水印信息
K_1, K_2, K_3	密钥|W|W 中的bit信息量
e	关系数据分组调节因子

在水印嵌入时,采用单向Hash函数来确定哪些元组可以用来进行水印嵌入。然后,对每个满足嵌入条件的元组 r_i 嵌入一位二进制水印信息,如算法1所示。将 $r_i.A$ 的值用 a_t 的值进行替换,t 由式(1)来确定:

$$t = \mathrm{cup}\ (\mathrm{Msb}(\mathrm{Hash}(r_i.P,\ K_1),\lceil \log_2 n \rceil) \cdot \mathrm{mod}(n-1),0,W[\mathrm{Msb}(\mathrm{Hash}(r_i.P,k_2),\lceil \log_2 |W| \rceil),\mathrm{mod}\ |W|]) \quad (1)$$

收稿日期:2008-08-28。
项目来源:国家自然科学基金资助项目(60673139)。

函数 cup(a, b, c)计算将 a 的第 b 位用 c 替换后的结果。Msb(a, m)表示数值 a 的 m 位最重要位,Lsb(a)表示数值 a 的最不重要位,这里我们只选择最末位。& 表示按位"与"运算。元组的主键 $r_i \cdot P$ 和 K_1 生成 $0 \sim 2^{\lceil \log_2 g \rceil}$ 范围内的秘密数值,其最不重要位将被由 $r_i \cdot P$ 和 K_2 生成的秘密数值确定的水印信息替换。

算法 1　WMEncoding(R, K_3, K_1, K_2, e, W)

1) W=WMGen (R, K_3) //形成水印信息

2) for each tuple $r_i \in R$ do

3) if (Hash($r_i \cdot P$, A, K_1) mod e = 0)

4) a=Msb(Hash ($r_i \cdot P$, A, K_1), $\lceil \log_2 n \rceil$) mod(n−1)

5) b=Msb(Hash($r_i \cdot P$, A, K_2), $\lceil \log_2 |W| \rceil$) mod $|W|$

6) t=cup(a, pos(Lsb (a)), W[b])

7) $r_i \cdot A = a_t$

8) next tuple r_i。

在水印提取时(实现过程如算法 2 所示),首先进行水印信息二进制位串的恢复,用相同的方法确定可能嵌有水印信息的元组 r_i,根据 $r_i \cdot A = a_t$ 求得相应的水印信息为 t&1:w'[Msb(Hash ($r_i \cdot P$, K_2), $\lceil \log_2 |W| \rceil$)mod $|W|$] = t&1

算法 2　WMDecoding (R', K_3, K_1, K_2, e)

1) W=WMGen(R', K_3) //形成水印信息

2) for each tuple r_i(R' do

3) if(Hash($r_i \cdot P$, A, K_1) mod e=0)

4) 求得 t , 使得 $r_i \cdot A = a_t$

5) b=Msb(Hash($r_i \cdot P$, K_2), $\lceil \log_2 |W| \rceil$) mod $|W|$

6) w'[b] = t & 1

7) $W' = E(w')$ //处理提取的水印信息

在简单替换算法中,只是针对单个非数值型属性域 A 进行了水印嵌入编码。在该算法中,关系数据库有 N 个元组,即使对所有元组进行水印嵌入的情况下,大约最多只可嵌入 $\log_2 N$ 位水印信息。为了扩展水印带宽,使能够嵌入合理长度的水印信息,可以将水印信息嵌入在不同的属性域中。针对不同的属性域使用相应的域名和逐位取得的水印信息来确定进行替换的数据值:

$$t = \text{cup}(\text{Msb}(\text{Hash}(r_i \cdot P, A\text{Name}, K_1), \lceil \log_2 n \rceil) \bmod (n-1), 0, W) \quad (2)$$

1.2　统计特征控制算法

在算法 1 中,嵌入水印时没有直接将水印信息与原有关系数据属性值进行编码,而是用水印信息参与关系数据值的选取,保证了原有关系数据信息的可用性。但这种简单替换显然会破坏原有关系数据的统计分布特性。

本文首先分析原有属性域的统计特征,例如属性值 a_j 对应的频度为 c_j,函数 $f(a_j) = c_j/n$ 计算对应频率。然后利用部分水印信息对频率值作小波变换,用一个二维数组对最小频率和最大频率进行记录,直到数据库拥有者的最大限制约束,即将属性域的数据统计特征控制在拥有者许可的范围内。若水印嵌入后的统计特征在此范围内则嵌入,否则进行回滚操作。

2　实验结果

为检验算法的有效性及相关性能,针对部分 MIT cup98 数据进行了实验。

2.1　简单替换算法实验

对 1 200 个元组的关系数据分别进行一个属性域的 16 位、32 位、128 位二进制水印信息进行嵌入实验,结果如表 2 所示。可以看出随着水印信息信息量的增加,简单替换算法对原始数据的统计特征的影响在增大。

表 2　简单替换算法对属性域统计特征的影响

Tab.2　Change of Attribute Static Information under Simple Replace

属性值	嵌入前频率	16 位水印信息		32 位水印信息		128 位水印信息	
		嵌入后频率	改变比例/%	嵌入后频率	改变比例/%	嵌入后频率	改变比例/%
WI	0.057 5	0.026 7	1.45	0.060 0	4.35	0.071 7	24.6
GA	0.078 3	0.077 5	1.06	0.075 83	3.19	0.070 0	11.0
IL	0.080 0	0.079 2	1.04	0.078 33	2.08	0.076 7	4.17
MI	0.086 6	0.087 5	5.77	0.086 7	0	0.087 5	0.96
FL	0.097 5	0.097 5	0	0.983 3	0.85	0.098 3	0.85
NC	0.117 5	0.117 0	0.71	0.116 7	0.71	0.115 0	2.13

改进算法将水印信息分散在不同的属性域中。为此对 1 200 个元组的关系数据分别进行两个属性域的 16 位、32 位、128 位二进制水印信息嵌入实验。对同一属性域的平均频率改变比例和

最大改变比例情况与简单替换算法结果如表3所示，不难看出进行多属性的水印嵌入分散了因数据值的改变造成的属性域统计特征的影响。同时对不同的属性域进行水印编码时使用了不同的密钥，提高了对子集采样攻击的鲁棒性。

表3 将水印分散到不同属性域内频率改变比例

Tab.3 Frequency Change Ratio in Watermarking Different Attributes

	16位水印信息/%		32位水印信息/%		128位水印信息/%	
	平均值	最大值	平均值	最大值	平均值	最大值
单属性嵌入操作	1.82	4.17	3.96	6.42	10.0	24.6
两属性嵌入操作	1.45	5.77	3.32	4.35	4.32	11.3

2.2 统计特征控制算法实验

为控制嵌入水印信息操作对数据域统计特征的影响，假设允许各属性域的改变比例在$[-\xi,\xi]$内，即若原有数据值的频率为δ，水印嵌入后其频率应控制在$\delta\times[1-\xi,1+\xi]$范围内。选取$\xi=5\%$，分别对不同数据量(1 200, 2 400, 3 200, 5 600, 9 600)的数据库进行48 bits长度的水印信息嵌入操作，并与前两个算法的结果进行了比较，如表4所示。

2.3 水印信息提取与攻击实验

实现了针对统计特征控制算法的水印提取及攻击程序，针对部分MIT cup98数据进行了实验。

1) 水印提取

针对不同数据量(1 200, 2 400, 3 200, 5 600, 9 600)的数据库进行不同长度的水印信息嵌入操作，提取结果如表5。

表4 统计特征控制下的水印嵌入情况

Tab.4 Watermarking Insertion under Controlling Attribute Static Information

元组数目	平均改变比例/%			最高改变比例/%		
	简单算法	改进算法	统计特征控制算法	简单算法	改进算法	统计特征控制算法
1 200	4.35	3.04	0.76	8.23	7.35	1.21
2 400	3.23	1.52	0.61	9.42	9.04	0.68
3 200	3.45	3.07	0.13	7.96	6.83	4.19
5 600	2.26	1.91	1.19	8.93	8.19	6.82
9 600	2.21	1.84	0.09	5.22	3.51	4.81

表5 水印信息提取结果

Tab.5 Result of Watermarking Recovery

元组数量	1 200	2 400	3 200	5 600	9 600
平均提取比例/%	81.2	87.5	91.7	89.6	89.6

2) 攻击实验

对5 600个元组的数据库进行不同类型的攻击实验，结果如表6所示。可以看出，算法对于子集删除攻击、子集增加或更新攻击时的鲁棒性都比较好。

由于水印添加位置是根据主键、属性名称和密钥K_1的单向哈希函数值确定的，位置与属性值的行或列顺序无关，无论数据库的行列顺序怎样变化，都不会影响水印。因而，对水印数据库的索引操作和排序操作，均不会对水印构成威胁。

表6 攻击下的水印恢复

Tab.6 Watermarking Recovery under Data Attack

子集采样-删除	删除比例/%	<5	5～10	10～15	15～20	20～25
	平均提取比例/%	89.1	87.5	85.4	84.4	75.0
子集增加	平均提取比例/%	84.4	83.1	78.3	78.1	67.5
子集更新	平均提取比例/%	85.8	83.2	80.2	78.2	65.6

3 性能分析

1) 复杂度分析。在简单替换算法及其改进算法中，只需对N个元组的关系数据库进行一次遍历、更新操作，其复杂度为$O\left(N+\frac{N}{e}\right)\approx O(N)$。

为使水印信息嵌入的同时能够控制数据库的统计特征变换，需要在水印嵌入前对属性域的统计特征进行分析，故需要进行两次遍历操作和一次更新操作。对有$n(n<<N)$个不同数据值的属性统计特征进行小波变换，其复杂度为：

$$O(N+N+n\lg n+N)\approx O(N)$$

2) 盲检测分析。本文提出的算法中，由于水印嵌入位置是根据主关键字、属性名称和密钥K_1的单向哈希函数值确定的，水印信息未参与水印嵌入位置的计算，实现了数字水印的盲检测。

参 考 文 献

[1] Agrawal R, Hass J P, Kiernan J. Watermarking Relational Databases[C]. The 28th VLDB Conference, Hong Kong, 2002

[2] Sion R, Atllah M, Prabhakar S. Rights Protection for Relational Data[C]. The 2003 ACM SIGMOD International Conference on Management of Data,

San Diego, 2003

[3] 牛夏牧, 赵亮, 黄文军, 等. 利用数字水印技术实现数据库的版权保护[J]. 电子学报, 2003, 31(12A): 2 050-2 053

[4] Zhang Zhihao, Jin Xiaoming, Wang Jianmin, et al. Watermarking Relational Database Using Image [C]. The Third International Conference on Machine Learning and Cybernetics, Shanghai, 2004

[5] 黄敏, 张浩, 曹加恒. 一种基于关系数据库的水印技术[J]. 计算机工程与应用,2005(10): 153-155

[6] 肖湘蓉,孙星明. 基于水印的数据库安全控制研究[J]. 计算机工程与应用,2005(6): 175-178

[7] Zhang Yong, Zhao Dongning, Li Deyi. Watermarking Relational Databases [J]. Journal of PLA University of Science and Technology (Natural Science Edition), 2003, 4(5): 1-4

[8] Sion R, Atallah M. Prabhakar S. Resilient Information Hiding for Abstract Semi-Structures [C]. The Workshop on Digital Watermarking, Seoul, 2003

[9] Solanas A, Domingo-Ferrer J. Watermarking Non-numerical Databases [EB/OL]. http://www.springerlink.com/content/c0vn1423pp165667/fulltext.pdf, 2007

第一作者简介:董晓梅,副教授,博士,主要研究方向为信息安全。
E-mail:dongxiaomei@ise.neu.edu.cn

Study of Watermarking Nonnumeric Data in Relational Databases

DONG Xiaomei[1]　*TIAN Yueping*[1]　*LI Xiaohua*[1]　*YU Ge*[1]

(1 School of Information Science and Engineering, Northeastern University, 11 Lane 3 Wenhua Road, Shenyang 110004, China)

Abstract: Most of current digital watermarking algorithms for relational databases can only embed digital watermark into numeric data. They seldom deal with nonnumeric data. Therefore, research on digital watermarking algorithms that can embed watermark into non-numeric data is very important, which will expand the application of digital watermarking. The particularity of embedding digital watermark into non-numeric data is analyzed. A simple substitution algorithm is firstly proposed, which can embed digital watermark into single attribute. Then an improving approach is proposed to extend the previous algorithm to multiple attributes. On the purpose of making less change to the statistics of the data in watermarking, a statistic control algorithm is then proposed to improve the previous two algorithms. Finally, some relative experiments are designed to test the proposed algorithms. The experimental results show that the algorithms are effective and robust.

Key words: relational database; nonnumeric data; digital watermark; simple substitution; statistic control

About the first author: DONG Xiaomei, associate professor, Ph.D, majors in the information security.
E-mail: dongxiaomei@ise.neu.edu.cn