第33卷 第19期 计 算 机 工 程 2007年10月
Vol.33 No.19 Computer Engineering October 2007
·安全技术· 文章编号：1000—3428(2007)19—0170—03 文献标识码：A 中图分类号：TP301

水平划分数据的私密保持序贯模式挖掘

张文燕[1]，欧阳为民[2]
(1. 上海大学计算机科学与工程学院，上海 200072；2. 上海体育学院管理学院，上海 200438)

摘 要：研究了以下情况下的私密保持序贯模式挖掘：(1)多方参与；(2)每方均有自己的私有数据集；(3)要求在这多个水平划分的私有数据集的并集上多方合作挖掘序贯模式，同时各方均不向其他方泄露自己的私有数据信息。利用可交换加密技术和同态加密技术，提出一个新颖的基于安全多方计算的私密保持序贯模式挖掘算法。

关键词：可交换加密技术；同态加密；多方安全计算；私密保持序贯模式挖掘

Privacy Preserving Sequential Patterns Mining on Horizontally Partitioned Data

ZHANG Wen-yan[1], OUYANG Wei-min[2]
(1. Computer Science and Engineering College, Shanghai University, Shanghai 200072;
2. Management Department, Shanghai University of Sport, Shanghai 200438)

【Abstract】 This paper focuses on the privacy preserving sequential patterns mining in the following situation: (1)multiple parties; (2)each has a private data set; (3) wish to collaboratively discover sequential patterns on the union of the multiple private data sets without disclosing their private data to each other. It puts forward a novel approach to discover privacy-preserving sequential patterns based on secure multi-party computation by using commutative encryption and homomorphic encryption technology.

【Key words】 commutative encryption; homomorphic encryption; secure multi-party computation; privacy preserving sequential patterns mining

数据挖掘应用的一般前提是数据的开放使用，然而在现实世界里，数据库中很可能包含某些敏感信息，不宜对外泄露，因此，需要研究如何在保证不泄露敏感信息的条件下进行有效的数据挖掘[1]，私密保持数据挖掘研究应运而生。国际上对关联规则、分类和聚类等问题的私密保持挖掘研究较多，而序贯模式[2]的私密保持挖掘研究极为稀少。本文研究了水平划分数据的私密保持序贯模式挖掘，问题描述如下：设$Part_1$, $Part_2$, …, $Part_n$各自拥有一个包含敏感信息的数据集DB_1, …, DB_n，而且DB_1, …, DB_n是水平数据分布的，即为同构数据库，所有的站点都有相同的模式结构，但是每个站点包含关于不同实体的记录。各方拟在$DB_1 \cup DB_2 \cup \cdots \cup DB_n$上实施某种序贯模式挖掘算法，并且希望信息泄露受到限制，一个站点不能知道其余站点的内容，比如其余站点支持什么样的序贯模式以及支持度，除非信息是由站点本身以及全局最终结果泄露(比如一条规则的支持度是 100%，则可知每个站点对这条规则的支持度)。因此，本文提出了基于可交换加密[3]和同态加密[4]的安全多方计算[5]协议。

1 相关研究工作

1.1 安全多方计算

A.C.Yao 于 1982 年首先提出了安全两方计算[5]，O.Goldreich等人随后提出了可以计算任意函数的基于密码学安全模型的安全多方计算协议。安全多方计算是指，在分布式计算环境中一组参与者共同计算某个函数，每个参与者均提供一个输入，但是出于安全考虑，每个参与者提供的输入要求对其他参与者保密，每个参与者仅知道自己的输入和得到的输出以及由此可以推导出的信息。

安全计算协议是以组合电路的形式进行表达的，虽然在理论上具有一般性和简单性，但是其效率较低，对数据挖掘这种需要庞大输入数据的应用来说是不实用的。因此，有必要针对特别的数据挖掘任务提出简洁高效、易于实现的安全多方计算协议。

1.2 私密保持数据挖掘

自 2000 年R.Agrawal与R.Srikant提出基于数据随机化的私密保持数据挖掘方法以来，国际学术界又提出了包括数据干扰、数据加密和安全多方计算的技术，涉及关联规则、分类、聚类等数据挖掘问题，但尚未涉及序贯模式问题。文献[3]提出了运用可交换加密方法求关联规则的全局候选项集的方法和利用随机数干扰的方法确定全局频繁项集的方法，采用第 2 种方法，如果站点i+1 和站点i-1 同谋，则可能泄露站点i项目集的支持情况。因此，本文应用文献[5]的同态加密机制提出了其改进协议。

2 序贯模式挖掘

序贯模式挖掘是 R.Agrawal 与 R.Srikant 首先提出的。与关联规则挖掘类似，都是在交易数据库中试图挖掘先前未知的有价值的知识。所不同的是，关联规则是试图挖掘在同一交易内的顾客购物联系，如“有 75%的顾客在商场购物时买了物品 A 很可能同时还会买物品 B”；而序贯模式则是设法挖掘在不同交易间的顾客购物联系，如“有 75%的顾客在商场购物时买物品 A 后下次购物会很可能购买物品 B”。

作者简介：张文燕(1979－)，女，硕士研究生，主研方向：数据挖掘；欧阳为民，博士、教授、博士生导师

收稿日期：2006-10-30 **E-mail：**zh.wenyan@163.com

在序贯模式挖掘中，给定一个客户交易数据库 D。每笔交易记录由客户标识符、交易时间和所购物品组成。在同一交易时间，任一顾客不会产生多于一笔的交易，而且仅考虑在交易中购买哪些物品，不考虑所购物品的数量。

项目集是非空的项目集合，序列则是项目集的有序列表。顾客序列为某顾客按交易时间排序的在不同时间购买的项目集的有序列表。如果某一序列包含在某顾客序列中，则称该顾客支持该序列。一个序列的支持度为支持该序列的顾客数除以顾客总数所得的百分比值。本文称支持度不低于某个用户指定阈值 min_support 的序列为频繁序列。给定一个顾客交易数据库 DB，序贯模式挖掘就是在该数据库中挖掘最长的频繁序列。每个这样的最长频繁序列称为序贯模式。

2.1 问题定义

数据库DB在n方上($\text{Part}_1, \text{Part}_2, \cdots, \text{Part}_n$)水平划分，$\text{DB} = \text{DB}_1 \cup \cdots \cup \text{DB}_n$。$\text{DB}_i$位于$\text{Part}_i$，如果$\text{Part}_i$交易记录中包含项目集$X$，则用$X.\text{sup}_i$表示项目集$X$在$\text{Part}_i$上的支持度，称之为局部支持度。$X$的全局支持度定义为 $X.\text{sup} = \sum_{i=1}^{n} X.\text{sup}_i$，如果项目集$X$的支持度满足 $X.\text{sup} \geq s \times \left(\sum_{i=1}^{n} |\text{DB}_i|\right)$，则称$X$为全局项目集。$L_k$ 代表全局k项集，LL_k 代表Part_i支持的局部K项集。$GL_{(k)} = L_{(k)} \cap LL_k$ 代表Part_i局部频繁项目集中包含的全局k项集。分布式序贯模式挖掘的目的是当K>1 时，找到所有的$L_{(k)}$以及支持度。快速分布式关联规则挖掘算法(FDM)可以用于序贯模式挖掘。FDM概括如下：

(1)候选项集产生：在 $GL_{(k-1)}$ 的基础上产生 $CG_{i(k)}$，$GL_{(k-1)}$ 代表Part_i在(k-1)次循环中使用Apriori-generate算法产生全局候选项集。

(2)局部剪枝：对于每个 $X \in CG_{(k)}$，扫描位于Part_i的数据库DB_i，计算 $X.\text{sup}_i$。如果X在Part_i 上是局部大项集，则X肯定包含在集合 $LL_{i(k)}$ 中。很明显，如果X包含在全局项目集中，则它至少被一方支持。

(3)支持度交换：广播 $LL_{i(k)}$，得到 $\cup_i LL_{i(k)}$，各个站点计算 $\cup_i LL_{i(k)}$ 中包含元素的支持度。

(4)广播挖掘结果：各个站点广播 $\cup_i LL_{i(k)}$ 中元素的支持度，由此，每个参与方就可以计算$L_{(k)}$。

2.2 可交换加密

在隐私保护协议中可交换加密是一个重要的工具，对于任意可行的加密密钥 $K_1, \cdots, K_n \in K$，被加密信息 M 和任意置换 i, j，如加密算法满足以下 2 个特性，就是可交换加密算法。

$$E_{k_{i1}}(\cdots E_{k_{in}}(M)\cdots) = E_{k_{j1}}(\cdots E_{k_{jn}}(M)\cdots) \tag{1}$$

其中，$\forall M_1, M_2 \in M$，只要 $M_1 \neq M_2$ 且对于给定的 $k, \varepsilon < \frac{1}{2^k}$，

$$Pr(E_{k_{i1}}(\cdots E_{k_{in}}(M_1)\cdots) = E_{k_{i1}}(\cdots(E_{k_{in}}(M_2)\cdots))) < \varepsilon \tag{2}$$

3 序贯模式挖掘的安全多方计算协议

将 FDM 算法应用于本文的水平划分数据的私密序贯模式挖掘算法，用本文提出的协议 1 代替 FDM 算法中广播局部频繁大项集 $LL_{i(k)}$ 以及其支持度这一步。本文给出在不暴露特定项目集来源的前提下，求局部频繁项目集并集的方法。

在 FDM 算法中(见 2.1 节)，第 3 步暴露了项目集的来源，即哪个站点支持哪些大项集，泄露了数据持有方的隐私。为了到达隐私保护的目的，对于一方传出去的用于与其他方交流的信息进行加密，使得其他方无法从相互交流的信息中得到此方的隐私信息，并且采用了信息在挖掘参与方进行加密时置换加密序号，使信息来源不明显的方法，在信息交换完毕后再进行解密，这样就在不泄漏私有信息的情况下，进行了安全的多方计算，得到了全局结果，在本算法中涉及到了对于数据项集的加密解密。根据式(1)可知可交换加密算法是安全的。

协议 1 全局候选项集的收集算法

其主要思想是根据可交换加密原理，每个站点对于它自己的局部大项进行加密，而且要向局部大项集合中掺入一些干扰项集来隐藏实际被支持的局部大项集的数量。然后每个站点对于其他站点传过来的项集进行加密。

这里给出本节用到一些符号。F 代表那些能被用来作为干扰项集的数据，$LL_{e_{i(k)}}$ 表示在站点 i 上被加密了的局部频繁 k-项集，$E_{k1}(i)$ 是站点 i 上的加密函数，$D_{k2}(i)$ 是站点 i 上的解密函数。

算法步骤如下：

(1)每个站点得到 $LL_{e_{i(k)}}$，集合 $\cup LL_{i(k)}\left(i = 1, \cdots, n\right)$ 的大小是全局候选项集 $CG_{(k)}$ 的大小，而 $CG_{(k)}$ 是每个站点都知道的。一个站点能够用一个标准随机数生成器来干扰集合 $LL_{e_{i(k)}}$。

(2)站点 0 从其他的偶数站点得到完整的加密过的项集集合，而站点 1 从其他的奇数站点得到完整的加密过的项集集合。分开收集是为了加强算法的安全性，使得站点 0 从其他站点得到的隐私信息更少。

(3)站点 1 将收集到的项集集合传到站点 0，然后站点 0 将 2 个集合进行合并。

(4)依据式(2)，不管项目集在各个站点上的加密次序如何，总可以按照相同的次序解密。将合并后的项集集合依次在每个站点上进行解密，将干扰项去掉，得到合并后的解密项集，即全局候选项集。最后将结果广播出去。

4 全局候选项集的支持度阈值判定协议

在全局候选项集的支持度阈值判定协议中，本文采用同态加密[4]的特性。同态加密是允许直接对密文进行操作的加密变换。同态加密技术由算法的同态性保证了用户可以对敏感数据进行操作但又不泄露数据信息。其基本思想如下：

假设 E_{k1} 和 D_{k2} 分别代表加密、解密函数，$K1$ 为加密密钥，$K2$ 为解密密钥。明文数据空间中的元素是有限集合 $\{M_1, \cdots, M_n\}$，α 和 β 代表运算，若

$$\alpha(E_{k1}(M_1), \cdots, E_{k1}(M_n)) = E_{k1}(\beta(M_1, \cdots, M_n))$$

成立，则 $(E_{k1}, D_{k2}, \alpha, \beta)$ 为同态加密函数族。根据上述特性，可以得到

$$E_{k1}(M_1) \times \cdots \times E_{k1}(M_n) = E_{k1}(M_1 + \cdots + M_n)$$

4.1 全局候选项集的支持度阈值判定协议

由协议 1 可以求出全局候选项集即 $\cup LL_{i(k)}\left(i = 1, \cdots, n\right)$，还需要确定这些候选项目集是否满足用户指定的支持度阈值。FDM算法中的(4)使得站点泄露了 $LL_{(k)}$ 中元素的支持度。对于项目集 $X \in LL_{(k)}$，需要确定 $X.\text{sup} \geq s \times |D|$？，根据 $X.\text{sup} \geq s \times |D| = s \times \left(\sum_{i=1}^{n} |DB_i|\right)$，可以将之归结为判定 $\sum_{i=1}^{n}\left(X.\text{sup}_i - s \times |DB_i|\right) \geq 0$ 是否满足。不必真正交换支持数，只要能够得到最终的比较结果即可。在不暴露 $X.\text{sup}_i$ 或 $|DB_i|$ 的前提下，协议 2 给出了解决方法。先随机选择某一方为密钥生成者，不失一般性，本文假定Part_1为密钥生成者。

协议 2 全局候选项集的支持度阈值判定协议

(1)Part_1生成加密函数 E_{k1} 和解密函数 D_{k2}，其中，$K1$ 为加密密钥；$K2$ 为解密密钥，并置候选序列支持计数器

CS.Count=0。$Part_i$ 计算自己站点的 $(X.\sup_i - s \times |DB_i|)$，记作 M_i，对其进行加密，即有 $E_{k1}(M_i)$。

(2)$Part_i$发送 $E_{k1}(M_i)$ 到$Part_n$。

(3)$Part_n$计算 $t = E_{k1}(M1) \times \cdots \times E_{k1}(M_n)$，并将计算结果$t$发送到$Part_1$。

(4)$Part_1$解密$Part_n$的计算结果t，即

$$D_{k2}(t) = D_{k2}\left(E_{k1}(M_1) \times \cdots \times E_{k1}(M_n)\right)$$

如果 $D_{k2}(t) \geq 0$，则候选项目集为全局频繁项目集；否则，将之从候选项目集中去除。

4.2 正确性分析

因为：

$$E_{k1}(M_1) \times \cdots \times E_{k1}(M_n) = E_{k1}(M_1 + \cdots + M_n)$$

所以：

$$D_{k2}(t) = D_{k2}\left(E_{k1}(M_1) \times \cdots \times E_{k1}(M_n)\right) = M_1 + \cdots + M_n$$

如果$Part_1$的解密结果值≥0，则候选项目集的支持度满足用户设定的阈值，为全局频繁项目集；否则有 $M_1 + \cdots M_n < 0$，候选项目集从所在的集合中删除。因此，上述全局候选项集的支持度阈值判定协议能够正确地得出候选项目集是否为全局频繁项目集。

4.3 私密性分析

按上述协议，$Part_n$从$Part_i$ $(i = 1, \cdots, n)$ 获得的是 $E_{k1}(M_i)$，由于没有解密密钥 $K2$，因此$Part_n$并不知道候选项目集在$Part_i$上的支持度。同样，$Part_1$ 从 $Part_n$ 获得的是 $E_{k1}(M1) \times \cdots \times E_{k1}(M_n)$，借助其拥有的解密密钥 $K2$，可以获知 $(M_1 + \cdots + M_n)$ 的大小，即CS.Count与用户指定的阈值的比较值，但并不知道$Part_j$ $(j = 1, \cdots, n)$ 对候选项目集的支持度。至于$Part_2, \cdots, Part_{n-1}$，由于它们只是将本方数据加密后发送$Part_n$，除了最后得到候选序列支持数与用户指定的阈值的比较值外没有得到任何别的信息。由此可知，协议各方的数据私密性均得到了保护。

5 结论

本文主要研究如何在保持水平划分数据各方数据私密性的前提下多方合作挖掘序贯模式的问题。为此，提出基于可交换加密机制和同态加密机制的私密序贯模式挖掘的安全多方计算协议，运用了可交换加密机制求全局候选项目集，将同态加密技术运用于全局候选项集支持度收集的步骤中，不必将各自的数据全部发送到可信任的第三方，很大程度上保证了各方数据的私密性。

下一步的工作是提出私密保持度量方法，定量分析安全多方计算协议所能达到的私密保持的水平，运用本文思想来处理安全多方分类、聚类等问题。

参考文献

1 Agrawal R, Srikant R. Privacy-preserving Data Mining[C]//Proc. of ACM SIGMOD'02. 2000: 439-450.

2 Agrawal R, Srikant R. Mining Sequential Patterns[C]//Proc. of the 11th Int'l Conference on Data Engineering. 1995.

3 Kantarcioglu M, Clifton C. Privacy-preserving Distributed Mining of Association Rules on Horizontally Partitioned Data[J]. IEEE Transactions on Knowledge and Data Engineering, 2004, 16(9).

4 Rivest R L, Adleman L, Detroulos M L. On Data Banks and Privacy Homomorphism[C]//Proc. of Foundations of Secure Computatin. New York: Academic Press, 1978: 169-179.

5 Yao A C. Protocols for Secure Computations[C]//Proc. of the 23rd Annual IEEE Symposium on Foundations of Computer Science. 1982.

（上接第 169 页）

数据查询 SELECT 的巴科斯范式语法结构如下：

```
select_command::="select"["all"|"distinct"]
("*"|( displayed_column{","displayed_column}))
"from"( selected_table{","selected_table})
["where"condition]{connect_clause}{group_clause}
{set_clause}{order_clause}{update_clause}
```

数据对分离出的 displayed_column 或“ * ”中的加密字段和非加密字段，标记加密字段。以便从数据库中取出后对加密字段解密，以及重新组装恢复 SELECT 语句。

数据操纵语句 INSERT,UPDATE,DELETE 的巴科斯范式语法结构如下:

```
insert_command::="insert""into"([schema_name"."]
(table_ name))"("column_list")"
(("("value_list")")|query)
update_command::="update"[schema_name"."]table_name
"set"column"="( expression|subquery)
{","column"="(expression|subquery)}
"where"condition
delete_command::="delete""from"( schema_name"."]
table_name["where"condition]
```

在数据操纵于语句 INSERT,UPDATE,DELETE 中，DELETE 中不涉及对单个字段的操作，因此不用对它分离字段处理。INSERT 和 UPDATE 中进行分离加密字段和非加密字段，对加密字段用对称加密体制加密数据，再组装成 SQL 数据操纵语句，写入数据库。所有的 SQL 语句解析过程中只对处理的字段进行加/解密处理，对条件和其他部分因为加密字段不能实现对数据制约因素的定义和密文数据的排序、分组和分类等以及 SQL 语言中的内部函数将对加密数据也失去作用，因此可以用原语句结构追加回组装 SQL 语句中。

5 结论

本文重点论述了远程数据访问的安全性要求及所设计的方案，一定程度上满足了对敏感数据访问的安全要求，并进一步确定了实现该模型的客户身份认证模块、数据安全传输模块、SQL 语句解析模块和数据安全事务模块的实现机制，该模型具有信息保密性、信息完整性、身份认证等功能，但在具体实现时还需用到其它一些安全技术，如访问控制技术来确保整个系统的安全要求。

参考文献

1 任　江, 袁宏春. 对 SSL 协议及其安全性分析[J]. 电子科技大学学报, 1998, 37(4): 24-27.

2 冯登国, 卿斯汉. 信息安全: 核心理论与实践[M]. 北京: 国防工业出版社, 2000.

3 尚　杰, 戴一奇, 李向阳. 密文数据库及其密钥管理[J]. 计算机应用研究, 1996, 24(3): 34-36.