一种基于隐私保护的分布式聚类算法

姚 瑶 吉根林
(南京师范大学计算机系 南京 210097)

摘 要 针对水平划分的分布式数据库提出了一种基于隐私保护的分布式聚类算法PPDK-Means,该算法基于K-Means的思想实现分布式聚类,并且聚类过程中引入半可信第三方,应用安全多方技术保护本站点真实数据不被传送到其他站点,从而达到隐私保护的目的。理论分析和实验结果表明PPDK-Means算法是有效的。

关键词 分布式聚类,隐私保护,安全多方计算

Distributed Clustering Algorithm Based on Privacy Protection

YAO Yao JI Gen-lin
(Department of Computer, Nanjing Normal University, Nanjing 210097, China)

Abstract This paper proposed algorithm PPDK-Means for privacy-preserving K-Means clustering over horizontal partitioned database. Using semi-trusted third party and secure multi-party technology, PPDK-Means does not transfer real data to other sites in clustering procedure. Theoretical analysis and experimental results show that algorithm PPDK-Means is effective and privacy preserving.

Keywords Distributed clustering, Privacy preserving, Secure multi-party computation

人们提出了一些分布式聚类算法,如K-DMeans[1],DK-Means[2],D-Combing[3],DBDC[4],SDBDC[5]等,这些算法不具有隐私保护功能,它们在聚类过程中将本站点有关真实数据传送给其他站点,从而导致信息泄露。在实际分布式聚类应用中,有时候需要保护本站点的真实信息不被传送给其他站点,即需要进行隐私保护,为此,需要研究基于隐私保护的分布式聚类算法。

聚类过程中的隐私保护方法可大致分为数据扰乱和安全多方计算两种。基于数据扰乱的隐私保护聚类思想是通过转换数据使得真实的敏感数据不为人知,然后再进行聚类分析。而基于安全多方计算的隐私保护聚类主要通过构造安全多方协议,使得一组站点在仅仅拥有自己私有信息的情况下能最终获知全局聚类信息。后者主要应用于分布式聚类分析。

针对垂直划分的分布式数据库,文献[6,7]提出基于隐私保护的分布式聚类算法,而本文针对水平划分的分布式数据库提出了基于隐私保护的分布式聚类算法PPDK-Means(Privacy Preserving clustering with Distributed K-Means)。该算法基于K-Means的思想实现分布式聚类,并且聚类过程中引入半可信第三方,应用安全多方技术保护本站点真实数据不被传送到其他站点,从而达到隐私保护的目的。理论分析和实验结果表明PPDK-Means算法是有效的。

1 相关概念

(1)安全多方计算

安全多方计算的基本思想是[8]:在整个计算过程中,如果每一方除自己的输入和结果以外不知道另外任何一方的私有信息,则此计算视为安全的。分布式计算的要求是各个站点之间进行通信共同参与计算。安全多方计算的关键是如何保证站点在共同参与计算的情况下,它们之间的通信数据没有揭示出任何私有信息。

(2)半可信第三方

安全多方计算的一种方法是,假设有一个完全可信第三方,每个站点把自己的信息发送给可信方,可信方进行计算后得出结果发送给各个站点。但是现实世界中并不存在完全可信第三方,所以引入半可信第三方(Semi-Trusted Third Party),简称STTP。所谓半可信第三方,简单说就是参与方遵守协议的要求,但是它可能会试图根据中间计算结果,尽量获得额外信息。

半可信第三方STTP基于这样的假设[9]:①第三方是不可信的。因此,不能从参与的各方获得私有信息,以及从计算结果得到私有信息。②第三方不可以和任何一方有勾结。③第三方严格遵守安全多方协议。本文利用半可信第三方产生随机向量,伪装站点间的通信信息,从而达到隐私保护目的。

2 聚类过程中的隐私保护

设分布式系统中有 p 个站点 $\{S_1, S_2, \cdots, S_p\}$,各站点相应的 m 维局部数据集分别为 $\{DB_1, DB_2, \cdots, DB_p\}$,每个局部数据集的大小分别为 $n_1, n_2, \cdots, n_p$,$DB=\bigcup_{i=1}^{p} DB_i$ 称为全局数据集。全局数据集可划分为 k 个聚簇 $\omega_1, \omega_2, \cdots, \omega_k$,每个簇中的数据点个数分别为 $t_1, t_2, \cdots, t_k$,全局聚簇中心点分别为 $c_1, c_2, \cdots, c_k$,聚簇 $\omega_i (i=1,2,\cdots,k)$ 所对应的各站点的局部聚类

到稿日期:2008-06-30 本文受国家自然科学基金项目(40771163)资助。

姚 瑶(1984—),女,硕士研究生,主要研究方向为分布式聚类算法;吉根林(1964—),男,博士,教授,博导,主要研究方向为数据挖掘技术及其应用,E-mail:glji@njun.edu.cn。

中心为$\{c_{1i},c_{2i},\cdots,c_{pi}\}$，相应的数据点个数为$\{n_{1i},n_{2i},\cdots,n_{pi}\}$。聚类的目标函数$E=\sum_{i=1}^{k}\sum_{j=1}^{t_i}d_{ij}(x_j,c_i)$，其中$d_{ij}(x_j,c_i)$是数据点$x_j$和中心点$c_i$之间的距离。不失一般性，设$S_p$为主站点，$S_1,\cdots,S_{p-1}$为从站点。

2.1 基于 K-Means 的分布式聚类算法 DK-Means

文献[2]提出了分布式聚类算法 DK-Means，它是 K-Means 聚类算法的扩展，具体步骤描述如下：

输入：局部数据集$\{DB_1,DB_2,\cdots,DB_p\}$，聚簇的个数 k。

输出：k 个聚簇。

步骤：

```
master site Sp:broadcast ({c1,c2…,ck}); /*主站点随机产生 k 个初始聚簇中心并广播*/
while E is not stable do
{ for each slave site Sj(1≤j≤p−1) do
  { receive ({c1,c2,…,ck}); /*接收聚簇中心*/
    for each data object d∈DBj do
    partition(d,{c1,…,ck}); /*计算 d 与所有全局聚簇中心的距离*/
    for i=1 to k do /*根据最近原则确定 d 所属聚簇*/
    computing(cji,nji); /*计算 k 个局部聚簇信息*/
    send ({(cj1,nj1),…,(cjk,njk)}) to master site; /*向主站点传送局部聚簇信息*/
  }
  master site Sp:
  { for each data object d∈DBp do
    partition(d,{c1,c2,…,ck}); /*计算 d 与所有全局聚簇中心的距离*/
    for i=1 to k do /*根据最近原则确定 d 所属聚簇*/
    computing(cpi,npi); /*计算 k 个局部聚簇信息*/
    for j=1 to p−1 do
    receive ({(cj1,nj1),…,(cjk,njk)}); /*主站点接收从站点 i 的聚簇信息*/
    for i=1 to k do
    ci=(c1i×n1i+c2i×n2i+…+cpi×npi)/(n1i+n2i+…+npi); /*主站点计算 k 个全局聚簇中心*/
    broadcast ({c1,c2,…,ck}); /*向从站点广播全局聚簇中心*/
    computing(E); /*计算全局目标函数 E*/
    }
}
```

2.2 分布式聚类中的隐私保护思想

DK-Means 算法能够实现分布式聚类，但它不具有隐私保护功能，为此我们对它进行改进，提出 PPDK-Means 算法，利用安全多方计算技术，保护从站点向主站点发送的局部聚类信息不被泄露。

为了达到隐私保护的目的，需要一个半可信第三方 STTP，可以假设除主站点以外的任意一个站点为 STTP。不失一般性，假设S_1为 STTP。S_1需产生两个 p 维随机向量 V 和 V'，分别扰乱局部聚类信息 $c_{ji}\times n_{ji}$ 和 n_{ji} 的值，其中 $V=\{v_1,v_2,\cdots,v_p\}$，$V'=\{v_1',v_2',\cdots,v_p'\}$，且向量中元素需满足$\sum_{j=1}^{p}v_j=0$，$\sum_{j=1}^{p}v_j'=0$。随机向量产生完成后，$S_1$发送扰乱值$(v_j,v_j')$给相应站点$S_j$。

S_j接收到一对扰乱值(v_j,v_j')后，开始对局部聚类信息加密。首先，扩展v_j为m维的扰乱向量V_j，即$V_j=\{v_j,v_j,\cdots,v_j\}_{1\times m}$。然后，计算在站点$S_j$中聚簇$\omega_i$所对应的局部聚类中心与相应数据点个数的乘积$c_{ji}\times n_{ji}$，令$d_{ji}=c_{ji}\times n_{ji}$。则在站点$S_j$中$(d_{ji},n_{ji})$为需要保护的一对局部聚类信息。最后，$S_j$分别利用$V_j,v_j'$加密$(d_{ji},n_{ji})$为$(d_{ji}',n_{ji})'$，其中$d_{ji}'=d_{ji}+V_j$，$n_{ji}'=n_{ji}+v_j'$。依次类推，对$S_j$中每个聚簇的局部聚类信息做相同操作的加密。加密完成后发送加密后的局部聚类信息$\{(d_{j1}',n_{j1}'),(d_{j2}',n_{j2}'),\cdots(d_{jk}',n_{jk}')\}$给主站点进行全局聚类中心点计算。

主站点收到各从站点的所有局部聚类信息后，根据式(1)进行全局中心点计算，然后广播全局中心点给各从站点，重新进行聚类。直到目标函数 E 稳定，算法结束。

$$c_i=\frac{d_{1i}'+d_{2i}'+\cdots+d_{pi}'}{n_{1i}'+n_{2i}'+\cdots+n_{pi}'}\quad(i=1,2,\cdots,k)\tag{1}$$

在 PPDK-Means 聚类算法中，依靠 STTP 产生随机向量 V 和 V'扰乱局部聚类信息，从而保证了局部聚类信息不被泄露。同时因为随机向量具有$\sum_{j=1}^{p}v_j=0$，$\sum_{j=1}^{p}v_j'=0$的特征，使全局中心点信息$\{c_1,c_2,\cdots,c_k\}$没有因局部信息扰乱而改变，最终保证聚类结果不会因局部聚类信息加隐私保护操作而有改变。

2.3 PPDK-Means 算法描述

上面论述了算法 PPDK-Means 的隐私保护思想，下面描述 PPDK-Means 的具体步骤：

算法 PPDK-Means

输入：局部数据集$\{DB_1,DB_2,\cdots,DB_p\}$，聚簇的个数 k。

输出：k 个聚簇.

步骤：

```
master site Sp:broadcast ({c1,c2,…,ck}); /*主站点随机产生 k 个初始聚簇中心并广播*/
while E is not stable do
{ STTP site S1:
    for i=1 to k do{ /*产生两个 p 维随机向量*/
    V=Random_Vector(p); /* V={v1,v2,…,vp}*/
    V'= Random_Vector(p); /* V'={v1',v2',…,vp'}*/
    for j=1 to p do{
    send(vj,vj') to site Sj; /*发送扰乱值给每个站点*/
    }
for each slave site Sj(1≤j≤p−1) do
  { receive ({c1,c2…,ck}); /*接收聚簇中心*/
    for each data object d∈DBj do
    partition(d,{c1,…,ck}); /*计算 d 与所有全局聚簇中心的距离*/
      for i=1 to k do{ /*根据最近原则确定 d 所属聚簇*/
      computing(cji,nji); /*计算 k 个局部聚簇信息*/
      receive(vj,vj'); /*接收 STTP 发送来的扰乱值*/
      Vj=expand(vj); /*将扰乱值 vj 扩展为 m 维的扰乱向量 Vj*/
      dji= cji×nji; /*计算局部聚类中心与相应数据点个数的乘积*/
      dji'= dji+Vj; /*扰乱向量加密局部聚类中心与相应数据点个数的乘积*/
      nji'= nji+vj'; /*扰乱值加密数据点个数*/
    }
    send ({(dj1',nj1'),(dj2',nj2'),…(djk',njk')}) to master site; /*向主站点传送局部聚簇信息*/
```

```
  }
master site Sp:
  { for each data object d∈DBp do
    partition(d,{c1,c2,…,ck}); /*计算 d 与所有全局聚簇中心的
    距离*/
    for i=1 to k do{ /*根据最近原则确定 d 所属聚簇*/
    computing(cpi,npi); /*计算 k 个局部聚簇信息*/
    receive(vp,vp'); /*接收 STTP 发送来的扰乱值*/
    Vp=expand(vp); /*将扰乱值 vp 扩展为 m 维的扰乱向量 Vp
    */
    dpi=cpi×npi; /*计算局部聚类中心与相应数据点个数的乘积
    */
    dpi'= dpi+Vp; /*扰乱向量加密局部聚类中心与相应数据点
    个数的乘积*/
    npi'= npi+vp'; /*扰乱值加密数据点个数*/
    }
    For j=1 to p-1 do
    receive ({(dj1',nj1'),(dj2',nj2'),…(djk',njk')}); /*主站点接收
    从站点 i 的聚簇信息*/
    for i=1 to k do
    ci=(d1i'+d2i'+…+dpi')/(n1i'+n2i'+…+npi'); /*主站点计算 k 个全局聚簇中心*/
    broadcast ({c1,c2,…,ck}); /*向从站点广播全局聚簇中心*/
    computing(E); /*计算全局目标函数 E*/
  }
}
Array Random_Vector (int p) /* 随机生成 p 维向量[v1,v2,…,vp],
要求向量元素之和为 0*/
{   sum=0;
    for j=2 to p do{
    vj←random();
    sum+ = vj;
    }
  v1 = - sum;
  return([v1,v2,…,vp]);
}
```

3 实验结果与分析

为了研究本文提出的算法 PPDK-Means 的性能，我们使用 3 台微机构成 100Mb 的局域网，微机配置为 Intel Pentium Ⅳ 2.93GHZ/512MB，开发环境为 JBuilder 2006 Enterprise。利用 Java 实现了 PPDK-Means 算法和未加隐私保护的分布式聚类算法 DK-Means。实验数据源如表 1 所列，其中，Iris[10]是植物样本数据库，KDD-CUP-99 800 和 KDD-CUP-99 8000 是从 KDD-CUP-99[11] 中分别随机抽取 800 个记录和 8000 个记录构成的数据库。

表 1 测试数据集

数据集	对象个数	属性维数
Iris	150	4
KDD-Cup-99 800	800	34
KDD-Cup-99 8000	8000	34

(1)聚类精度

Modha 和 Spangler[12] 利用了数据的分类信息来评价聚类结果的好坏，即当数据有分类信息时，可认为该分类信息在一定程度上表达了数据的一些内部分布特性。如果该分类信息没被聚类算法利用，则可以用它来评价聚类性能，其度量标准 Micro-precision 定义如下：

$$\text{Micro-precision}=\frac{1}{n}\sum_{i=1}^{k}\alpha_i$$

其中，n 为数据集样本总数，k 为聚类的类数，α_i 为聚类的类 i 与已知数据集类别对应后，类 i 中被正确归为相应类别的样本个数。Micro-precision 的值越大，表示在该数据集上聚类效果越好。这种度量方法适合聚类时产生固定类数的算法，如 K-means 等，因此实验中采用该标准对各算法的精度进行比较，实验结果取 10 次实验的平均值，聚类精度如图 1 所示。实验结果表明 PPDK-Means 与不加隐私保护的聚类算法 DK-Means 在聚类精度上是相同的，经过隐私保护进行的分布式聚类并没有改变聚类的结果。

(2)聚类效率

算法 PPDK-Means 与 DK-Means 的执行时间如图 2 所示。实验结果表明 PPDK-Means 比 DK-Means 执行效率低。其原因很显然，因为 PPDK-Means 增加了半可信第三方与其余各站点的通信。

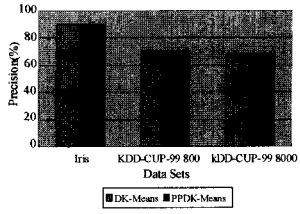

图 1 PPDK-Means 与 DK-Means 聚类精度比较

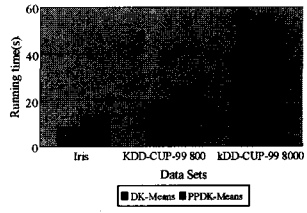

图 2 PPDK-Means 与 DK-Means 执行时间比较

(3)安全性分析

PPDK-Means 算法中共有 3 类站点，分别为：主站点 S_p，STTP 站点 S_1 以及从站点 $S_2,S_3,\cdots,S_{p-1}$。对于主站点 S_p，除了本站点局部聚类外，接受了各个站点经过伪装的局部聚类信息，所以不能根据其不真实的聚类信息推断出各个站点的数据点分布情况。对于 STTP 站点 S_1，除了本站点局部聚类外，负责产生随机向量，分发扰乱值给各个站点，但各个站点通过扰乱值伪装的局部聚类信息只发送给 S_p，所以 S_1 也不能推断出其余站点的聚类信息。对于从站点 $S_2,S_3,\cdots,S_{p-1}$，只是根据接受到的全局中心点信息进行本站点局部聚类，这类站点进行的通信来源有 3 个方面：①接受 S_1 站点的扰乱值。②发送经过伪装的局部聚类信息给站点 S_p。③接受来自 S_p 站点的全局中心点信息。除此之外，不进行任意两站点间的通信，过程中并没有暴露局部信息的隐私。

结束语 分布式隐私保护聚类算法 PPDK-Means 采用安全多方计算技术，利用半可信第三方产生随机向量，伪装各个站点所要通信的中心点信息，最终达到隐私保护的目的。在数据挖掘过程中增加隐私保护技术，使其在发现知识的同时，又保护了数据安全。这是一项非常重要的研究工作。

参 考 文 献

[1] Kantabutra S, Couch A L. Parallel k-means clustering algorithm on Nows. NECTEC Technical Journal, 2000, 1(6): 243-247

[2] 郑苗苗，吉根林. DK-Means-分布式聚类算法 K-DMeans 的改进. 计算机研究与发展，2007，44 (suppl)：84-88

(下转第 105 页)

接，测试数据集包含了两百万个数据连接，每条数据样本有41个属性，描述了网络连接的基本特征、内容和通信量统计等方面的信息。数据集包括含有标识的训练数据和未加标识的测试数据，共有1种正常的标识类型 normal 和22种训练攻击类型。另外有14种攻击仅出现在测试数据集中。

5.2 实验结果及分析

由5.1节中选取的数据包 kddcup_data_10percent，模拟真实网络环境中入侵行为较少、中等和较多的情况，分别形成3组数据集，每组数据集中的训练集和测试集的详细情况如表1所列。

表1 数据集样本组成情况

组	训练/测试	总数	正常	异常			
				DOS	R2L	U2R	Probing
一	训练	2000	1960	24	10	2	4
	测试	2500	2430	42	18	4	6
二	训练	2500	2310	114	47	10	19
	测试	3000	2770	138	57	12	23
三	训练	3000	2460	324	135	27	54
	测试	3500	2870	378	157	31	64

采用表1中的训练集，按照本文提出的基于 FP-Growth 改进算法的数据挖掘模块分别对3组训练集进行学习训练，将异常模式提取出来，形成规则库，然后用测试集分别进行实验测试，实验结果如表2所列。

表2 基于 FP-Growth 改进算法的数据挖掘实验结果

组	检测率	误报率	漏报率	训练时(秒)
一	96.01%	3.71%	14.28%	10.31
二	94.93%	4.33%	13.91%	12.69
三	93.34%	5.85%	10.31%	15.02

采用未改进的 FP-Growth 算法进行实验的结果如表3所列。

表3 未改进的 FP-Growth 算法实验结果

组	检测率	误报率	漏报率	训练时间(秒)
一	94.88%	4.52%	25.71%	20.25
二	93.73%	5.05%	20.86%	24.94
三	91.74%	6.79%	14.92%	31.16

比较表2和3可以看出：

(1)相对于未改进的 FP-Growth 算法，3组训练时间分别降低 49.09%，49.12% 和 51.80%，由此表明，采用 FP-Growth 改进算法可以大大提高系统的检测效率。

(2)对于每一组数据，改进后的 FP-Growth 算法的检测效率提高 1.19%～1.75%，误报率降低 13.84%～17.52%，漏报率降低 30.90%～44.46%，检测性能有较大改善。

结束语 将数据挖掘技术应用到入侵检测系统是目前入侵检测研究的重要方向，本文设计的基于数据挖掘的分布式网络入侵检测系统采用了改进的 FP-Growth 的关联分析算法和基于分箱统计的 FCM 网络入侵检测技术，有效地解决了数据挖掘速度问题，增强了入侵检测系统的检测能力，为系统管理人员网络维护与保障提供了坚实的基础。

参 考 文 献

[1] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. KDD Cup 1999 Data

[2] 关健. 入侵检测系统数据分析方法及其相关技术的研究. 博士学位论文. 哈尔滨工程大学，2004

[3] 史志才，季振洲，胡铭曾. 分布式网络入侵检测技术研究. 计算机工程，2005，31(13)

[4] Gopalakrishna R, Spafford E H. A Framework for Distributed Intrusion Detection Using Interest Driven Cooperating Agents. Department of Computer Science, Purdue University, May 2001

[5] Fayyad U M, Piatesky-shapiro G, Smyth P. Advances in knowledge discovery and data mining. Galifornia: AAAI/MIT Press, 1996

[6] Ramaswamy S, Rastogi R, Shim K. Efficient algorithms for mining outliers from large data sets // Proceedings of the ACM SIGMOD International Conference on Management of Data. Dallas, TX, USA, 2000: 427-438

[7] Han Jia wei, Chee S H S, Chiang J Y. Issues for On-Line Analytical Mining of Data Warehouses

[8] Fuchsberger A. Intrusion Detection Systems and Intrusion Prevention Systems. Information Security Technical Report. 2005, 10: 134-139

[9] Kim G H, Spafford E H. Experiences with tripwire: Using integrity checkers for intrusion detection[R]. West Lafayette, USA: Purdue University, Depatment of Computer Sciences, 1994

[10] Lee W, Stolfo S J, Chan P K, et al. Real time data mining-based intrusion detection[A] // Proceedings of 2nd DARPA Information Survivability Conference and Exposition (DISCEX)

(上接第102页)

[3] Prodio H, Lawrence H. Scalable Clustering: A Distributed Approach // Proc. IEEE Int'l Conf. Fuzzy Systems. Budapest, Hungary. ETATS-UNIS, 2004, 7(1): 143-148

[4] Januzaj E, Kriegel H P, Pfeifle M. DBDC: Density Based Distributed Clustering // Proc. the 9th Int'l Conf. Extending Database Technology. Heraklion, Greece: Springer, 2004: 88-105

[5] Januzaj E, et al. Scalable Density Based Distributed Clustering // Proc. the 8th Eur. Conf. Principles and Practice of Knowledge Discovery in Databases. Paris: Springer, 2004: 231-244

[6] Vaidya J, Clifton C. Privacy-Preserving K-Means Clustering Over Vertically Partitioned Data // Proc. of the 9th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining. Washington, DC, 2003: 206-215

[7] Lin X, Clifton C, Zhu M. Privacy Preserving Clustering with Distributed EM Mixture Modeling. Knowledge and Information Systems, 2005, 8: 68-81

[8] Clifton C, Kantarcioglu M, Vaidya J, et al. Tools for Privacy Preserving Distributed Data Mining. SIGKDD Exploration. 2002, 4(2): 28-34

[9] Zhang Guo-rong, Yin Jian. Preserving clustering over distributed data. Computer Engineering and Applications, 2007, 43(18): 165-167

[10] http://www.ics.uci.edu/~mlearn/databases/iris/

[11] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[12] Modha D S, Spangler W S. Feature weighting in k-means clustering. Machine Learning, 2003, 52(3): 217-237