



云环境下一一种低成本的数据安全存储和处理框架

张丽敏

(西安外事学院现代教育技术中心 西安 710077)

摘要:提出了一种基于云计算的数据存储和处理框架,支持用户对大规模数据集进行低成本、安全的共享和处理。该框架研究了基于矩阵的数据挖掘算法,重点是安全的 Top- k 特征向量算法。该算法采用了一种迭代处理模型,授权用户可以与云端展开交互以获得预期结果,交互过程中源矩阵和中间结果均处于保密状态。另外,基于 Paillier 加密算法和高效的随机向量扰动策略,实现了新算法的安全性。实验结果表明,提出的方法对大型矩阵具有可拓展性,且客户端成本较低。

关键词:云计算;数据挖掘;迭代;Paillier 加密;安全性

doi: 10.11959/j.issn.1000-0801.2015049

Low Cost Safety Data Storage and Processing Framework Under Cloud Environment

Zhang Limin

(Modern Educational Technology Center, Xi'an International University, Xi'an 710077, China)

Abstract: Cloud-based data storage and processing framework, which enables users can economically and securely share and handle big datasets with low cost, was proposed. The framework studied data mining algorithms based on the matrix, whose focus was the secure Top- k feature vector algorithm. The algorithm used an iterative processing model and the authorized user interacted with the cloud to achieve the expected result. In this process, the interaction of source matrix and intermediate results were kept confidential. In addition, the security of this algorithm was guaranteed by using Paillier encryption and a random perturbation technique. Experimental results show that the proposed method is scalable to big matrices while requiring low client-side costs.

Key words: cloud computing, data mining, iteration, Paillier encryption, security

1 引言

随着网络服务、移动应用和传感器网络的不断发展,从多个分布式数据源中采集而成的数据集规模越来越大,数据集的用途越来越广。例如,移动设备的用户可以生成用户间交流情况、用户对产品的兴趣及用户位置、用户语

音等多种有用信息。这些数据集已经成为数据拥有者的重要资产。这一现象也为数据存储、共享和分析带来了多种挑战:移动设备和传感器等大多数数据采集器可以用于存储和处理数据的资源非常有限,数据只能被迫发送给服务器或云端;采集来的数据可能非常敏感,对数据传输、存储和处理过程的安全性要求更高。安全地处理和分析大数据

收稿日期:2014-09-25;修回日期:2015-01-28

基金项目:陕西省教育厅 2014 年科学研究项目(No.14JK2117)

Foundation Item: Scientific Research Fund of Shaanxi Provincial Education Department in 2014(No.14JK2117)

需要分配大量的计算资源,而云计算是实现这一目的的理想平台^[1]。

参考文献[2]提出一种方法,可以将计算任务安全外包。然而,该方法基于 Gentry 的全同形加密算法^[3],计算成本和密文体积太大,不具有可行性。Naehrig 等人^[4]提出一种安全的外包解决方案,只需要加密方法“稍微”同形即可,但是仍然存在密文体积过大的问题。Atallah 等人^[5]提出一种适用于线性方程和矩阵相乘大规模应用系统的安全外包方案。但是该方案会泄露隐私信息,依赖于非串通服务器,导致严重的通信开销,因此仍显不足。

为此,文中提出一种基于云技术的数据存储和处理框架,使用户花费较小成本便可对从分布式数据源中采集来的大型矩阵进行安全共享和处理。在该框架下,数据贡献者可以通过服务提供商(如数据拥有者)提供的移动应用,把数据向量(如描述用户在社交网络服务上交互情况的向量或者描述用户感兴趣话题的向量)提交给云端。数据拥有者授权数据消费者(即社交服务提供商的数据挖掘团队)展开分析。使用该框架,数据拥有者和数据消费者不再需要拥有自己的计算基础设施,只需要一台 PC 与云基础设施交互即能完成高强度的数据计算任务。本文框架的主要问题就是数据的安全性。当数据传输到云端后,数据拥有者便失去了对数据的控制权,而底层的基础设施由并不可靠的云供应商提供,在云环境下实现数据的安全处理就非常困难。本文研究的目的就是使受到信任的相关方在不可靠的云基础设施顶层展开协作,对加密后的数据进行安全计算。

2 背景知识

本节简要描述了文中用到的特征值分解以及 Paillier 加密等背景知识。 $\mathbb{Z}_q^n$ 表示模为 q 的 n 维整数向量的集合, $\{b_i\}$ 表示向量(或数值)集合,其中 $i=1, \dots, k$ 。

2.1 基于幂迭代的特征值分解

特征值分解是数据分析的重要工具。例如,主成份分析法就依赖于特征值分解^[6]。在信息检索领域,特征值分解常用来确定文本语料库词组和文档间的关系,这些领域的矩阵规模可能非常大。因此,复杂度为 $O(n^3)$ 的直接计算方法不具有可行性。相反,对于大型矩阵,往往使用基于幂迭代^[7]的方法求解 Top- k 特征向量。假设 $A \in \mathbb{R}^{n \times n}$ 是 $n \times n$ 维实数矩阵, b_0 是随机 n 维向量,基于幂迭代的特征值分解算法如下。

$b_0 \leftarrow$ 随机维向量

for $i \leftarrow 1$ to k do

$b_i \leftarrow Ab_{i-1} / \|Ab_{i-1}\|$

幂迭代算法的其他运算,成本为 $O(n)$

end for

生成特征向量且成本为 $O(n)$ 的后处理

在该迭代框架中,最复杂的运算是 $b_i \leftarrow Ab_{i-1} / \|Ab_{i-1}\|$,其他运算只与 b_i 和部分 $k \times k$ 维辅助矩阵有关。通常只需要部分特征向量,因此 k 较小(如 $k=10$)。幂迭代方法其余计算过程的复杂度只有 $O(kn)$ 。如果矩阵存储于云端,则云端负责最复杂的部分,而客户端负责剩余低成本运算。

2.2 Paillier 加密

全同形加密方法允许加密数据在不加密的情况下执行加法和乘法运算。Paillier 加密方法是一种部分同形加密方法,其效率远高于参考文献[3]中的全同形方法,但是只支持同形加法,即:

$$E(x)+E(y)=E(x+y) \quad (1)$$

在 Paillier 加密方法中,乘法操作不得运行于 $E(x)$ 和 $E(y)$ 之上。然而,如果有操作数没有加密(如 y),则乘法运算可表示为:

$$E(xy)=E(x)\text{mod_power } y \quad (2)$$

其中,mod_power 表示模幂运算^[8]。为了便于阐述,本文使用来 $(E(x))^y$ 表示 $E(x)\text{mod_power } y$ 。已经证明,Paillier 加密的安全性很强,可以满足语义安全的定义。

3 云环境下 Top- k 特征向量安全计算方法

本节描述了本文方法的主要组件。首先,给出总体计算框架,描述客户端和云端组件的功能;其次,给出威胁模型,进而提出基于 Paillier 加密和随机扰动的安全幂迭代算法和云端 MapReduce 算法;最后,分析本文方法的安全性和经济性。结果表明,本文方法可以实现云端计算成本最小化,同时保证处理大型矩阵的优异性能和安全性。

3.1 计算框架

本文方法涉及四方:云端、数据拥有者、数据采集器/贡献者和经过授权的数据用户。

图 1 为云环境下矩阵数据框架,阐述了各方间的关系和交互。非云端方(数据拥有者)、数据采集器/贡献者和经过授权的用户是可以信任的。数据拥有者对数据的各种权力进行控制,分发公共密钥,指示数据采集器/贡献者上传采集来的并且经过公共密钥加密的数据。数据拥有者可以

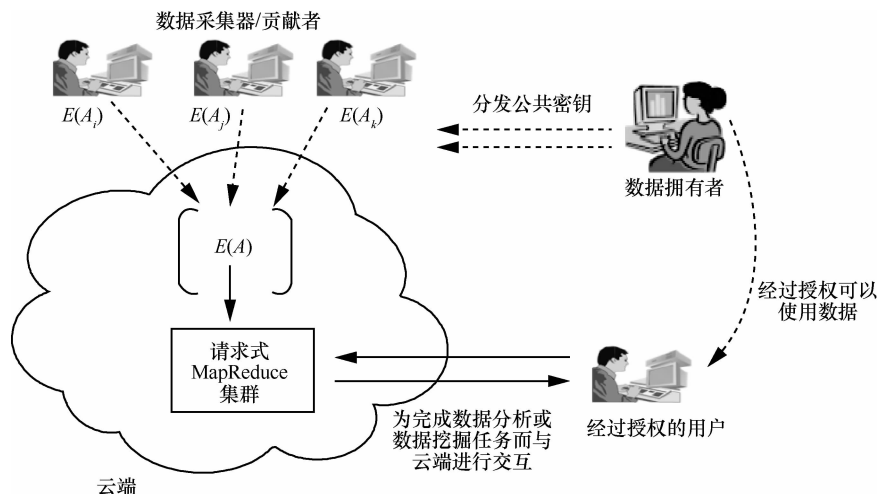


图1 云环境下矩阵挖掘框架

自己处理汇总数据或者授权其他用户使用加密数据。每个数据采集器可以贡献小部分数据,如一行矩阵。实际例子包括一段时间内与其他用户的交互或者对某些条目的建议。数据拥有者或授权用户可以为完成数据分析或数据挖掘任务而与云端交互。请注意,加密后数据的规模远大于初始数据。授权用户无法承担在本地下载经过加密的大型矩阵及计算任务。相反,他们希望充分利用云计算的优势,尽量降低云端的成本。

3.2 威胁模型

本文的安全分析以所讨论的架构的重要特征为基础,认为如下假设是合理的。

- 云提供商是不可靠的,可能会秘密观察数据和数据的计算过程,进而发现有用信息,如特征向量。
- 只有数据拥有者和授权用户可以使用专有矩阵。数据拥有者可以授权部分可靠用户使用数据,可靠用户不会故意泄密。不会出现内部人员攻击现象。
- 客户端系统和通信信道经过妥善的安全处理,隐私矩阵或计算结果不会泄露。
- 敌人只能看到经过安全处理的矩阵和提交的参与矩阵一向量安全计算的明文向量,其他信息均无法看到。

通过实施合适的安全规则便可以满足这些假设。攻击者的目标是想要恢复或估计矩阵元素和计算结果,即特征值和特征向量。

3.3 数据采集

数据表示:为了使用 Paillier 加密系统,所有数据需要转化为非负的整数。然而,实践中是在实数域处理矩阵。实

现预期精度(如 d 位小数)的合理方法是将初始数值与 10^d 相乘,从而提高数值的标度,舍弃剩余小数位。此外,需要对数据进行移位(shift)以保证数域为正。这一过程是完全可逆的,因此可以准确恢复结果。

提交数据:为了采集数据,数据拥有者将生成一个 n 维随机向量 b_0 ,且 $b_0 \in \mathbb{Z}_q^n$ 。 b_0 利用 Paillier 公共密钥加密,即 $E(b_0) = (E(b_{01}), \dots, E(b_{0n}))$,然后分发给数据采集器。

数据采集器 i 将其矩阵 A_i 的部分行以加密的形式提交给云存储。此外,它们将利用如下同形算法计算 $E(A_i b_0)$ 的结果,并将其提交给数据拥有者。假设 A_i 某一行 a ,则:

$$E(ab_0) = \sum_{k=1}^n E(a_k b_{0k}) = \sum_{k=1}^n (E(b_{0k}))^{a_k} \quad (3)$$

请注意, $E(A_i b_0)$ 中的元素数量与采集器将要提交给云端的行的数量相同,且该数量通常为 1。最后,数据拥有者收集所有的 $E(A_i b_0)$,对其解密以寻找 Ab_0 。

3.4 幂迭代协议

为了在幂迭代时保护提交给云端的明文向量的安全性,得到授权的数据用户必须执行数个步骤,以便为扰动方法做好准备。然后,客户端与云端展开协作,在迭代中完成安全矩阵一向量乘法运算。

备好扰动池。经过授权的数据用户将从数据拥有者接收 $E(b_0)$ 、 $E(Ab_0)$ 以及解密密钥,然后选择 m 个 n 维随机向量,并将其发送给云端,其中 m 较小(如 $m=5$)。这些随机向量将被用于每次迭代时扰动及保护向量 $\{b_i\}$ 。本文将其表示为种子随机向量 $\{s_i\}$,其中 $i=1, \dots, m, s_i \in \mathbb{Z}_q^n$ 。

对每个随机向量 s_i ,在云端按照如下步骤进行安全的 $A s_i$ 计算。鉴于 Paillier 加密方法的同形属性,对于结果向量

$(As_i)_j$ 的第 j 个元素,有:

$$E((As_i)_j) = E\left(\sum_{k=1}^n A_{jk}s_{ik}\right) = \sum_{k=1}^n (E(A_{jk}))^{s_{ik}} \quad (4)$$

其中, s_{ik} 表示向量 s_i 的第 k 个元素, A_{jk} 表示矩阵 A 的第 (j,k) 个元素。将 $E(As_i)$ 发送回客户端, 经过解密留作稍后处理。在准备阶段后, 经过授权的用户保留随机向量 $S=\{s_i\}$ 及结果向量 $A_s=(As_i), i=1, \dots, m$ 。

迭代阶段从随机向量 b_0 开始, 执行 $b_{k+1}=Ab_k/\|Ab_k\|$ 及幂迭代方法中描述的其他低成本步骤。在每次迭代时必须保护 b_i , 否则, 将会泄露特征向量。利用如下方法来保护计算的隐私性, 其安全性将在稍后进行详细分析。

为了通过 Paillier 同形运算, 从 $E(A)$ 和 b_i 中计算出 $E(Ab_i)$, b_i 不得被加密。在将 b_i 发送给云端前, 本文设计了一种扰动方法来保护 b_i 。基本思路是使用一个随机向量 r_i , 并将 $\bar{b}_i$ 发送给云端。

$$\bar{b}_i = b_i + r_i \bmod q \quad (5)$$

其中, q 表示较大的随机素数, q 足够大以包含应用域中的所有数值。利用准备阶段生成的种子随机向量来设计 r_i 。

$$r_i = \sum_{l=1}^m \alpha_{il}s_l + \sum_{j=0}^{i-1} \beta_{ij}b_j \bmod q \quad (6)$$

其中, $i=1, \dots, k$, α_{il} 和 β_{ij} 从 $\mathbb{Z}_q$ 中随机选择。在扰动中包含 $b_j (j=1, \dots, i-1)$ 的目的是增强安全性, 这一点将在稍后分析。于是, 在准备阶段及先前步骤中计算出了 $\{As_k\}$ 和 $\{Ab_j, j < i\}$, 且 Ar_i 为:

$$Ar_i = \sum_{k=1}^m \alpha_{ik}As_k + \sum_{j=0}^{i-1} \beta_{ij}Ab_j \bmod q \quad (7)$$

此时, 只需要客户端的向量运算, 成本为 $O(n)$ 。

客户端设置 $\bar{b}_i = b_i + r_i \bmod q$, 利用式 (4) 计算 $E(A\bar{b}_i)$, 并把结果返回。客户端解密 $E(A\bar{b}_i)$ 以获得 $A\bar{b}_i$ 。 Ar_i 已知, 于是有 $Ab_i = A\bar{b}_i - Ar_i \bmod q$, 下次迭代时便可以计算 $b_{i+1} = Ab_i / \|Ab_i\|$ 。其他运算步骤的复杂度为 $O(kn)$, 客户端即可轻松完成。

3.5 云端 MapReduce 计算

经过 Paillier 方法加密的数据, 其体积要远大于未加密时。设密钥为 1 024 bit, 64 bit 双型初始数值加密后为 2 048 bit, 增长了 32 倍。任何基于 Diffie-Hellman 或大整数因式分解的加密方案均无法避免该成本。于是, 普通规模的问题转化为“大数据”问题。明确了这一问题后, 本文设计了基于 MapReduce 的同形矩阵一向量相乘算法。客户端

把受到扰动的向量 $\bar{b}_i$ 作为参数传递给 MapReduce 程序, 云端计算并返回 $A\bar{b}_i$ 。下面描述云端计算 $E(A\bar{b}_i)$ 时的 MapReduce 方法。

MapReduce 程序非常简单。map 函数使用安全矩阵一向量相乘计算式 (式 4), 并发送利用行号表示的结果。根据行号将 map 输出, 进行分割和排序, 再发送给相应的 reducer, reducer 将数据段写入磁盘。因为使用二进制表示法进行矩阵元素加密, 所以本文设计了专门的输入/输出格式类以处理二进制数据。针对加密后矩阵的 MapReduce 矩阵一向量乘法程序如下。

map($E(\bar{A}), b_i$)

/* $E(\bar{A})$: 分配给具体 map 的加密矩阵 A 的部分行; $\bar{b}_i$: 客户端发送的受扰向量 */

for ($\bar{A}$): $E(A_i)$ 的每一行 do

发送 $\left\langle j, \sum_{k=1}^d (E(A_{jk}))^{\bar{b}_{ik}} \right\rangle$

end for

partition (j, nr) /* j : 行号; nr : reduce 操作的总次数 */

返回 $\lfloor j/nr \rfloor$

reduce($\langle j, v \rangle$) /* j : 行号; v : 结果 */

发送 $\langle j, v \rangle$

3.6 安全性分析

本文算法包括 3 个部分: 数据采集、客户端的随机扰动、基于 Paillier 加密方法的矩阵一向量相乘。只要保证 Paillier 方法安全, 那么第一部分和第三部分就能安全实现。因此, 本文算法的安全性主要取决于第二部分。在扰动准确阶段, 云服务提供商可以采集扰动池中的初始随机种子向量 $S=(s_1, \dots, s_m)$ 及逐渐生成的扰动向量 $\{\bar{b}_i\}$ 。此外, 对手可能会知道, b_i 在次迭代内收敛于与目标特征值相对应的主导特征向量。于是, 有如下的推论。

推论 1 已知的 $\{\bar{b}_i\}$ 和 S 不会泄露 $\{b_i\}$ 的任何信息。

证明 首先证明 $\bar{b}_i$ 的情况, 其他情况类似。设 $a_i=(\alpha_{i1},$

$\dots, \alpha_{ik})$, 根据式 (5) 有 $\bar{b}_i = b_i + r_i = b_i + S_{a_i} + \sum_{j=0}^{i-1} \beta_{ij}b_j \bmod q, i=1, \dots, k$ 。

因此, $\bar{b}_1 = b_1 + r_1 = b_1 + S_{a_1} + \beta_{1,0}b_0 \bmod q$ 。已知 $\bar{b}_1$, 未知 r_1 , 如果对手可以根据从 $\mathbb{Z}_q^n$ 中随机均匀选取的样本推测出 b_1 , 则该对手即可把 r_1 和随机向量区分开来。

设 $r_1 = S_{a_1} + e_1 \bmod q$, 其中 $e_1 = \beta_{1,0}b_0$ 且 a_1 保密。如果 $S, \beta_{1,0},$



b_0 均匀随机选取, 则 $\langle S, r_i \rangle$ 和从 $\mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^n$ 随机均匀选取的样本之间的区分问题就转变为有误差学习决策问题^[9]。根据参考文献[9]可知, 如果 e_1 随机选取且 a_1 保密, 则 r_1 无法和随机均匀选取的样本区分开来。因此, b_1 也无法与随机均匀选择的样本区分开来。相同的结论也可以推广至 $i > 1$ 且未知量更多的情况。

因为 r_i 无法和随机均匀样本区分开, 所以无论 b_i 如何出现 (b_i 与充分大的 i 类似), $\{\bar{b}_i\}$ 也无法和任意随机向量集合区分开。因此, $\{\bar{b}_i\}$ 序列不会帮助敌人获得更多 $\{b_i\}$ 相关信息, 推论得证。

3.7 成本分析

在本文框架中, 数据采集器必须要加密一个向量, 加密次数为 $O(n)$ 次; 进行一次同形点积, 成本与向量加密类似 (见第 4 节); 把加密后的向量传输给云端, 成本为 $O(n)$ 。总体来说, 其通信和计算成本为 $O(n)$ 。

授权用户必须准备一个扰动池, 传输成本和解密成本均为 $O(nm)$, 其中 m 很小。在幂迭代中, 为了获得 k 个特征向量需要 $O(km)$ 次解密, 为了存储加密后的数值需要 $O(km)$ 空间。当 k 和 n 较小时, 客户端成本也较小。总体来说, 一台 PC 可以应付这一工作负载, 这一迭代对用户充分利用云计算的效益具有重要意义。

云端为存储经过加密的矩阵元素, 需要的空间成本为 $O(n^2)$, 还需要基于 MapReduce 的安全矩阵一向量相乘导致的计算成本。因为主要成本是 map 阶段, Hadoop 集群有 p 个 map 点, 因此总成本为 $O(n^2/p)$ 。存储成本与经过加密的数值数量成正比, 且与密钥尺寸有关。

4 仿真实验

为了证明本文方法的有效性, 设计了一系列实验来评估数据采集器、授权用户和云端三方的处理效率。

4.1 配置

客户端机器配置为 128 GB RAM、4 个四核 AMD 处理器。利用赖特州立大学的 Hadoop 集群来测试云端 MapReduce 程序。集群配置为 16 个从属节点, 运行 Apache Hadoop version 1.0.3。每个从属节点的配置为 16 GB RAM、4 个四核 AMD 处理器、16 个 map 槽、12 个 reduce 槽、64 MB HDFS 块。云端 MapReduce 程序用 Java 部署, 客户端程序用 C++ 和 GMP 库 (gmplib.org) 部署。

由于密钥低于 1 024 bit 的 Paillier 加密方法是不安全的^[10], 因此本文实验中使用 1 024 bit 密钥, 利用仿真矩阵

展开实验。初始矩阵使用双精度值作为元素 (每个数值 8 byte), 如第 3.3 节所示, 这些矩阵转化为长整数以便加密, 维持足够的精度。在幂迭代算法中使用加密后矩阵作为 MapReduce 程序的输入。

4.2 数据采集器的成本

框架下的每个数据采集器将生成一个或一些向量, 对向量加密, 然后将其提供给云端。此外, 它还将进行安全的向量点积运算以生成 $E(Ab_0)$ 。因此, 数据采集器的主要成本在于向量加密、安全向量点积运算以及传输, 这些成本由 Paillier 加密成本 (即密钥尺寸) 及维数确定。

本文使用二进制编码方法实现被加密数据量最小化, 同时实现通信和云存储最小化。一般来讲, 二进制表示的数据加密后的尺寸是加密密钥的两倍, 例如 64 bit 双精度数值使用 1 024 bit 键值加密后将成为 2 048 bit。表 1 给出了 10 000 维向量使用 1 024 bit 键值时的比较情况。

表 1 10 000 维向量导致的数据采集器成本

编码	加密/s	点积/ms	体积/MB	压缩后/MB
文本	56	3.7	6.1	3.5
二进制	56	3.7	2.5	2.5

与二进制法相比, 未经压缩的大整数的原始文本表示法的空间成本比其高出 150% (压缩后高出 40%), 但是计算时间基本相同。鉴于加密后数据的随机性, 二进制表示法的压缩效果有限。

4.3 迭代时的客户端成本

本文算法每次迭代时, 客户端都会从云端接收到加密后的向量 $E(A\bar{b}_i)$, 对其解密, 重新生成一个明文受扰向量, 并将其提交给下次迭代。本节将评估相应步骤的通信、存储和计算成本。

通信成本包括从云端接收到加密后向量以及将明文受扰向量返回给云端, 使用字节数量来表示这些成本。根据先前的讨论, 10 000 维的加密后向量将会花费 2.56 MB, 且字节需要传输给客户端。比较而言, 返回的由 10 000 个长整数组成的明文向量有 80 KB, 这些数据与维数成线性关系。

计算步骤包括解密向量并构建新的明文向量。假设扰动池包括 10 个随机生成的向量, 表 2 给出了不同维数的成本分布。可以看出, 解密步骤的用时最长, 因为使用多核处理器后解密步骤可以并行执行, 所以这一成本可以进一步降低。

表 2 客户端的计算成本

维数/维	$E(A\bar{b}_i)$ 体积/MB	解密 $E(A\bar{b}_i)$ /s	其他处理/ms
10 000	2.56	31	5
30 000	7.68	94	16
50 000	12.7	149	26

存储成本包括加密向量和扰动池中的明文向量。根据第 3.5 节的 MapReduce 计算可知,每次迭代后生成的 b_i 将被加入到扰动池。然而,迭代次数通常较小,因此扰动池的存储量较小即可。例如,有 10 000 个维度且扰动池的尺寸为 10,则 10 次迭代只需要 2 MB 的内存来存储明文长整数向量。

4.4 云端成本

云端的成本主要来自存储加密后数据及安全矩阵一向量相乘运算。加密后数据的体积非常大,授权用户一般不会本地下载并存储数据。表 3 给出了部分实数,从而更加具体地阐述问题规模。

表 3 用 1 024 bit 密钥时矩阵加密的存储成本

矩阵维数/维	未加密/GB	加密后/GB
10 000	0.8	25.8
30 000	7.2	232.0
5 000	20.0	645.0

很明显,使用传统的方法无法处理如此规模的数据,必须充分利用云端并行处理的特点。云端计算包括安全矩阵一向量乘法运算。

矩阵以行向量集合的形式进行存储,并在 Hadoop 文件系统中分割成大小为 64 MB 的数据块。MapReduce 框架为每个块分配一个 map,使得向量一向量点积集合可以并行出现。由于非常复杂的运算出现于 map 阶段,因此 map 的数量决定了整体性能。

图 2 表明,当加密后矩阵的规模不同时,成本呈现上升趋势。当维数低于 10 000 时,map 阶段在内部 Hadoop 集群上只需一轮即可完成。当维度更多时,需要更多轮 map,总体时间成本与 map 轮数成正比。平均来说,每轮 map 时间约为 30~40 s,这展示了较强的可拓展性。当增加集群的规模(map 槽数更多)可以维持一轮 map 时,总体成本将保持不变(约为 40 s)。

图 3 给出了使用内部集群时一次迭代所产生的总体成本在云端和客户端的分布情况。当维数上升时,总体成本主要取决于客户端成本。客户端和云端成本总体情况表

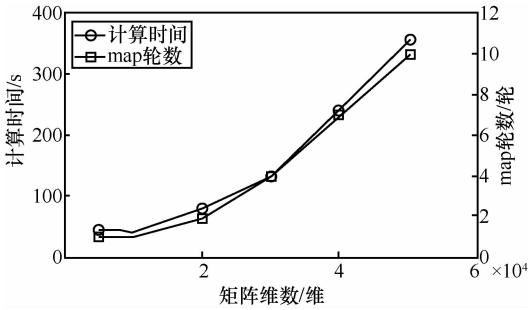


图 2 一次加密后导致的 MapReduce 计算成本

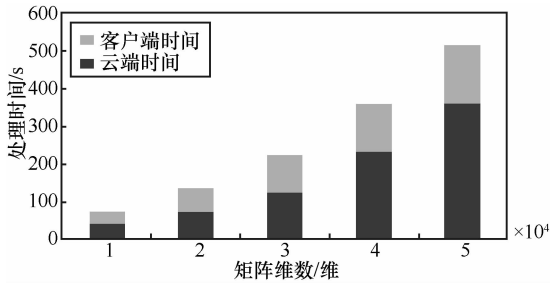


图 3 客户端与云端的成本比较

明,经过 MapReduce 优化部署后,客户端成本较低,云端效率较高。

5 结束语

本文提出一种幂迭代算法,可在云环境下从加密后数据中求得 Top- k 特征向量。基于 Paillier 加密算法和高效率的随机向量扰动策略,实现了本文算法的安全性。经过精心设计后,客户端计算成本实现了最小化。此外,本文还提出一种 MapReduce 程序,可以高效处理云端经过加密的矩阵一向量乘法运算。实验结果证明,本文方法在配置和使用过程中产生的存储和计算成本均在合理范围内,本文算法具有一定的可拓展性。下一步工作将对本文研究进行延伸,提出针对稀疏矩阵的更为高效的处理算法,同时提出不可靠或懒惰型服务提供商检测技术。

参考文献

1 邓维,刘方明,金海等. 云计算数据中心的新能源应用: 研究现状与趋势. 计算机学报, 2013, 36(3): 582~596
Deng W, Liu F M, Jin H, et al. Leveraging renewable energy in cloud computing data centers: state of the art and future research. Chinese Journal of Computers, 2013, 36(3): 582~596
2 Gennaro R, Gentry C, Parno B. Non-interactive verifiable computing: outsourcing computation to untrusted workers. Proceedings of the 30th Annual Cryptology Conference, Santa Barbara, CA, USA, 2010: 465~482

(2015049-7 接排在 2015010-11 后)