软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software, [doi: 10.13328/j.cnki.jos.006042]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

持续监控下差分隐私保护*

梁文娟[1,2,3], 陈红[1,2], 吴云乘[1,2], 赵丹[1,2], 李翠平[1,2]

[1](数据工程与知识工程国家教育部重点实验室(中国人民大学),北京 100872)

[2](中国人民大学信息学院,北京 100872)

[3](河南大学计算机与信息工程学院,开封 475001)

通讯作者: 陈红, E-mail: chong@ruc.edu.cn

摘 要: 近年来,随着信息技术的发展及物联网技术的兴起,出现了越来越多的持续监控应用场景,如智能交通实时监控、疾病实时监控、智能基础设施应用等.在这些场景中,如何对参与者持续分享的数据进行隐私保护面临重大挑战.差分隐私是一种严格和可证明的隐私定义,早期差分隐私研究大都基于一个大规模、静态的数据集做一次性的计算和发布.而持续监控下差分隐私保护需对动态数据做持续计算和发布.目前,持续监控下差分隐私保护是差分隐私领域新的研究热点之一.本文对持续监控下差分隐私保护的已有研究成果进行总结.首先对该场景下差分隐私保护模型进行阐述;然后重点介绍了持续监控下满足 event 级、user 级和 w-event 级隐私保护的实现方案.在对已有研究成果深入对比分析的基础上,指出了持续监控下差分隐私保护的未来研究方向.

关键词: 差分隐私;持续监控; event-级隐私; user 级隐私; w-event 级隐私;
中图法分类号: TP311

中文引用格式: 梁文娟,陈红,吴云乘,赵丹,李翠平.持续监控下差分隐私保护.软件学报. http://www.jos.org.cn/1000-9825/6042.htm

英文引用格式: Liang WJ, Chen H, Wu YC, Zhao D, Li CP. Differential Privacy under Continual Observation. Ruan Jian Xue Bao/Journal of Software, (in Chinese). http://www.jos.org.cn/1000-9825/6042.htm

Differential Privacy under Continual Observation

LIANG Wen-Juan[1,2,3], CHEN Hong[1,2], WU Yun-Cheng[1,2], ZHAO Dan[1,2], LI Cui-Ping[1,2]

[1](Key Laboratory of Data Engineering and Knowledge of Ministry of Education (Renmin University of China), Beijing 100872, China)

[2](School of Information, Renmin University of China, Beijing 100872, China)

[3](College of Computer and Information Engineering, Henan University, Kaifeng, 475001, China)

Abstract: With the development of information technologies and Internet of Things (IoT) technologies, there are more and more scenarios under continual monitoring, such as transportation monitoring, disease monitoring and smart infrastructure etc. In these scenarios, how to protect the privacy of continuous sharing data is facing major challenges. Differential privacy is a rigorous and provable privacy definition. Earlier research on differential privacy has focused on “one-shot" release on a static dataset. However, Differential privacy under continual observation focuses on the continuous computation on the dynamic dataset. Now it has become one of the research hotspots.This paper surveys the state-of-the-art techniques on differential privacy under continual observation, and focuses on summarizing existing schemes that provide event-level privacy、user-level privacy and w-event privacy. Following a comprehensive comparison and analysis of existing techniques, further research prospects are put forward.

Key words: Differential privacy; continual monitoring; event-level privacy; user-level privacy; w-event privacy.

随着信息技术的发展及物联网技术的兴起,出现了越来越多的持续监控应用场景,如健康管理、交通管理、

* 基金项目:国家自然科学基金(61532021, 61772537, 61772536, 61702522).
Foundation item: National Natural Science Foundation of China (61532021, 61772537, 61772536, 61702522).
收稿时间: 2018-07-06; 采用时间: 2019-12-20; jos 在线出版时间: 2020-04-21

智能建筑、智能基础设施与应急响应等.在这些场景中,持续监控的成功实施依赖于对参与者持续共享的个体信息或大量密集传感器(例如照相机、手机、WiFi 接入点、信标)传输的流式数据进行实时监控和分析.由于持续监控下持续共享的数据中包含大量个体隐私数据,如果不对其进行隐私保护,会存在隐私泄露的风险.用户对隐私泄露的顾虑会限制参与者分享个体信息的意愿,从而阻碍持续监控应用的发展[1,2].

为保护用户数据隐私,近两年很多国家和企业针对数据保护和个人信息的隐私保护问题制定了相关的法律政策.如2016 年,欧盟制定的《一般数据法案》(General Data ProtectionRegulation, GDPR)[1]中,明确了规定了用户对个人信息的知情权及被遗忘权. 2017 年,我国在《中华人民共和国网络安全法》[2] 中,加强了对个人信息保护的政策规定.很多信息交流平台及各种APP如Facebook、Twitter、美团、京东都制定并发布了自己的用户隐私保护政策.除制定相关法律政策,隐私保护问题的相关技术研究也逐渐受到重视.

差分隐私是一种定义极为严格的隐私保护模型,相比于近年来的k-匿名[3], l-多样性[4]和t-紧密性[5]等需要基于特殊攻击假设和背景知识的隐私保护技术,差分隐私因能够防止攻击者拥有任意背景知识下的攻击并提供有力的隐私保护,受到了极大关注并被广泛研究.早期差分隐私研究[6-10]主要是基于一个可信的管理者持有一个大规模、静态的数据集,面向特定查询任务,研究如何在保证用户隐私的前提下提高查询结果的可用性.由于基于的数据集是静态的,因此计算都是一次性的.然而,持续监控下差分隐私保护是一个新的差分隐私应用场景.在该场景中,需对动态数据做持续计算和发布,如何保护参与者持续更新的个体信息面临重大挑战.下面以几个具体持续监控下场景示例分析持续监控下差分隐私保护的必要性及挑战.

在交通实时监控[11,12]中,各种车辆实时提供位置给google等数据统计公司,公司统计大量参与者位置信息后,实时发布交通区域图.基于上述统计结果,无人驾驶汽车可以进行最优路线规划,城市交通管理部门也可提供实时事故管理来保证道路通行能力.车辆的一次位置分享,称之为一次事件,如何能提高持续发布统计结果的可用性,同时保护用户参与事件不泄露是隐私保护目标.除上述示例,还有很多应用场景都需进行持续监控下隐私保护.如在疾病实时监控[3] 中,用户通过与APP实时交互,持续回答一些关于症状的查询,可以确定自己是否患有该疾病;APP通过持续统计参与用户的信息,实时监控区域疾病人数并提高疾病管理响应时间.在能源部门[13-16],智能电网革命根据对能源供应和需求进行严格和高粒度的实时计量,从而提高实时需求响应质量,并对不可预测的能源需求来保证电网的稳定性和可靠性.在搜索引擎、在线零售商和社交网络中[17,25],需要持续统计用户数据来及时发现有社会价值和经济价值的信息;在一些政治活动中,网站持续调查民众意见并持续更新候选人支持率.在上述应用中,用户持续参与统计,计量设备持续输送流式数据都会增加隐私泄露风险.如果没有用户隐私保护机制,参与者分享个体信息的意愿会受到限制.

传统差分隐私场景下,往往根据一次参与事件对发布结果的最大影响(敏感度)添加噪声;以发布计数任务为例,一次事件的参与与否所带来的发布结果敏感度为1,添加$O(1/\epsilon)$的拉普拉斯噪声即可满足ϵ-差分隐私;但在持续监控下(如交通实时监控任务),一次参与事件不仅对当前发布值有影响,对后续时刻发布值都会有影响;假设时序长度为T,添加$O(T/\epsilon)$的拉普拉斯噪声才能实现一次事件的差分隐私保护.同时,由于时序上往往存在个体的多次参与事件,所带来的隐私泄露风险更大,因此需添加更多的噪声才能实现对多个事件的隐私保护.这就存在以下问题,一是随着时序长度的增加,发布结果可用性变差,如何降低噪声量对时序长度的依赖面临较大的挑战;二是为保护个体的多次参与事件,隐私预算需在时序上进行分配,如何提高时序上隐私预算的利用率,从而提高发布结果可用性也是一个挑战;三是上述任务只是基于计数的简单发布任务,发布任务自身敏感度低;如果是复杂分析任务的持续监控,监控任务自身的发布敏感度很大,所需添加的噪声量会更大,在这种情况下,如何保证发布结果可用性也存在很大挑战.

持续监控下差分隐私保护研究具有实际的应用背景和重要的研究意义.目前已有一些持续监控下差分隐私保护的研究成果.本文综述持续监控下差分隐私保护的最新研究进展和研究方向;一方面对持续监控下差分

1 https://en.wikipedia.org/wiki/General_Data_Protection_Regulation

2 http://www.spp.gov.cn/xwfbh/wsfbt/201705/t20170509190088.shtml

3 https://h1n1.cloudapp.net

隐私的定义、实现机制和持续监控方案的评估标准进行总结;另一方面,对当前持续监控下差分隐私保护的已有研究方案进行总结分析,其中着重总结满足 event 级、user 级和 w-event 级隐私保护的实现方案.最后,提出持续监控下差分隐私保护的未来研究方向.

本文第 1 节总结持续监控下差分隐私保护模型;第 2 节总结主要研究方案分类;第 3、4、5 节分别对持续监控下差分隐私保护的现有研究方案进行概括,并对研究方法进行对比和分析;第 6 节提出持续监控下差分隐私保护的未来研究展望;最后第 7 节总结全文.

1 持续监控下差分隐私保护模型

本节主要对持续监控下的差分隐私定义、实现机制及隐私保护方案评估标准等几个方面进行总结.

1.1 传统差分隐私定义

设数据集 D 和 D′, D 和 D′数据集属性结构相同,如果两者的对称差 $|D-D'|=1$,则称 D 和 D′为邻居数据集.也就是说, D 和 D′相差一条记录信息.

定义1[18]. 差分隐私. 给定一个随机化的算法 M, P_M 为 M 的所有可能的输出集合,如果算法 M 在任意邻居数据集D 和 D′上的输出结果O ($O \in P_M$)满足下列不等式,则 M 满足ϵ-差分隐私.

$$\Pr[M(D) \in O] \le \Pr[M(D') \in O] \times e^{\epsilon} \quad (1)$$

隐私预算参数ϵ表示隐私保护程度,ϵ 越小隐私保护程度越高.

1.2 持续监控下差分隐私定义

直观的讲,持续监控下的数据发布[19]可以看作是在一个离散的时间间隔序列$t=\{t_1,t_2,\dots,t_n\}$上,针对每个时间间隔 t_i ,算法重复接受输入、计算、然后输出的过程,而其差分隐私保护则是对时间序列上这个重复的过程及发布结果序列进行差分隐私处理以满足隐私保护要求.

在上节传统差分隐私定义中,差分隐私要保证基于邻居数据集上统计结果的概率比值不大于e^{ϵ},也就意味着在数据集中添加或删除某用户的参与信息后,发布结果变化不大,从而实现在静态数据集上个体一次参与事件的隐私保护.在持续监控下,差分隐私也要保证基于邻居数据集的统计结果满足上述要求.但该场景中,用户在不同的时间会有多次的参与事件.随着参与事件的增加,隐私泄露风险往往更大.如何在时序上对用户多次参与事件进行保护是持续监控下隐私保护的目标.如图 1 中持续监控场景示例,在时间序列$t=\{t_1,t_2,\dots,t_n\}$上,针对每个t_i,监控该时刻位置 L 的人数.

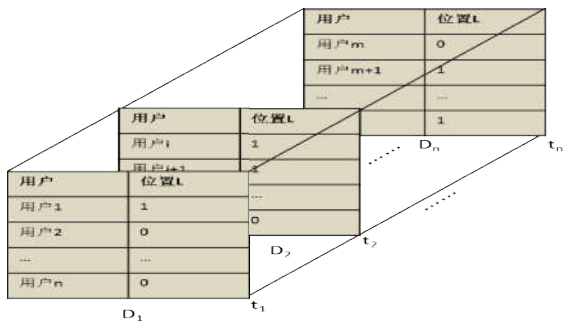

Fig.1 Example of continual monitoring

图 1 持续监控场景示例

假设图 1 中 D_i 代表 t_i 时刻的数据集, D_i中每行代表一个参与者在该时刻是否在位置 L,其值为 0 或 1, 1 表示在,0 表示不在.在持续监控下,数据集为一个动态序列$D=\{D_1,D_2,\dots,D_n\}$.每个时刻t_i,需要发布数据集 D_i 中位置 L 属性列上的 count 计数值.用户在某时刻 D_i中的一条记录称之为一次参与事件.随着时间的更新,每个 D_i 中都会存在某个参与者的参与事件,也就是说,D中存在用户的多次参与事件,而这些事件是持续监控下要保护的

用户隐私.

结合具体问题的监控事件,持续监控下差分隐私保护需要对基于监控事件数据流的统计分析做差分隐私保护处理.假设用$S=\{x_1x_2x_1x_1x_3\ldots\ldots\}$表示一个持续监控事件数据流,X表示数据流S中元素的值域,$x_i\in X$表示某用户一次参与事件.持续监控下差分隐私保护是基于邻居数据流进行定义.根据现有方案能提供的差分隐私保护级别,邻居数据流分为三种级别:event-级、user-级和w-event 级.三种隐私保护级别的邻居数据流及相应的差分隐私保护定义如下.

1.2.1 Event-级差分隐私

Event 级邻居数据流:在持续监控下,如果存在$x,x'\in X$,从S 中任意将某个x替换为x'后变为S′,称 S 和 S′为持续监控下 event 级邻居数据流.其含义是从监控数据流中添加(删除)某用户的一次参与事件.

定义2[20]. Event-级差分隐私. 给定一个随机化的算法 M，P_M 为 M 的所有可能的输出集合,对于算法 M 在任意 event 邻居数据流S 和 S′上任意的输出结果O ($O\in P_M$),如果其满足下列不等式,则 M 满足ϵ-差分隐私.

$$\Pr[M(S)\in O]\le \Pr[M(S')\in O]\times e^{\epsilon} \quad (2)$$

Event 级差分隐私保护能保证在持续监控生命期中,用户的一次事件对整个监控结果影响受隐私预算ϵ的控制.ϵ 越小,隐私泄露风险越低,反之越高.

1.2.2 User-级差分隐私

User 级邻居数据流:在持续监控下,如果存在$x,x'\in X$,从S 中将某用户对应的所有事件x都替换为x'后变为S′,称 S 和 S′为持续监控下的 user 级邻居数据流.其含义是从监控数据流中添加(删除)某用户的所有参与事件.

定义3[20]. User-级差分隐私. 给定一个随机化的算法 M,P_M 为M的所有可能的输出集合,对于算法 M 在任意 user 级邻居数据流S 和 S′上的输出结果O ($O\in P_M$)满足下列不等式,则 M 满足ϵ-差分隐私.

$$\Pr[M(S)\in O]\le \Pr[M(S')\in O]\times e^{\epsilon} \quad (3)$$

User 级差分隐私保护是对持续监控生命期中某用户的所有参与事件对整个监控结果的影响进行控制,也就是对持续发布统计结果的隐私泄露风险进行了限制.

1.2.3 w-event 级差分隐私

w-event 级邻居数据流:给定一个正整数 w,如果两个流前缀S_t和S_t'满足:(i)对于每个$i\in[t]$且$S_t[i]\neq S_t'[i]$都有 $S_t[i]$和$S_t'[i]$是相邻的;(ii)对于每个 $i_1<i_2$且 $S_t[i_1]\neq S_t'[i_1], S_t[i_2]\neq S_t'[i_2]$都有$i_2-i_1+1\le w$,则称两个流前缀$S_t$和 S_t'是 w-event 级邻居数据流.

定义4[21].w-event 级差分隐私. 给定一个随机化的算法M,P_M 为M的所有可能的输出集合,对于算法 M 在任意 w-event 级邻居数据流S_t 和S_t'上的任意的输出结果O ($O\in P_M$)满足下列不等式，则 M 满足ϵ-差分隐私.

$$\Pr[M(S_t)\in O]\le \Pr[M(S_t')\in O]\times e^{\epsilon} \quad (4)$$

w-event 级隐私可以看作是 user 级隐私的拓展,它实现在任意w滑动窗口内用户的 user 级的隐私保护.它也是 event 和 user 级的一个中间级隐私保护,当$w=1$时,保护级别下降为 event 级,当w为T时,则为 user 级隐私.

在上述的三种定义中, event-级所提供的隐私保护强度最小,user 级隐私保护强度最大,而w-event 级隐私保护是上述两种方案的折衷.隐私预算参数 ϵ 用来控制对持续监控下基于不同级别邻居数据集的输出结果的概率比值,也就是方案的隐私保护程度, ϵ 越小隐私保护程度越高,反之越低. ϵ的取值往往要结合监控所需要达到的隐私保护度和发布结果的可用性两个因素进行综合的衡量.

1.3 噪声机制

噪声机制是实现差分隐私保护的主要技术,常用的噪声添加机制分别为拉普拉斯机制[22]与指数机制[23].添加的噪声量与查询任务的全局敏感度和隐私预算大小密切相关.

定义5[22]. 全局敏感度. 设有函数 $f:D\to R^d$,输入为一数据集,输出为一 d 维实数向量.对于任意的邻居数据集D 和 D′,

$$GS_f=\max_{D,D'}\|f(D)-f(D')\|_1 \quad (5)$$

称为函数f的全局敏感度,$\|f(D)-f(D')\|_1$指的是$f(D)$和$f(D')$的 1-阶范数距离.函数的全局敏感度由函数本身

决定,不同的函数有不同的全局敏感度.敏感度越小,所需添加的噪声越少.

定理 1[22]. Laplace 机制. 给定数据集 D,设有函数$f:\mathrm{D}\to\mathrm{R}^{d}$,其敏感度为$\Delta f$,若算法 M 的输出结果满足下列不等式,则 M 满足ϵ-差分隐私.

$$\mathrm{M}(\mathrm{D})=\mathrm{f}(\mathrm{D})+\mathrm{Lap}\left(\frac{\Delta\mathrm{f}}{\epsilon}\right)^{\mathrm{d}} \tag{6}$$

其中,$\mathrm{Lap}((\Delta\mathrm{f})/\epsilon)^{\mathrm{d}}$为相互独立拉普拉斯噪声,噪声服从$\mathrm{b}=\Delta\mathrm{f}/\epsilon$的拉普拉斯分布.$\mathrm{Lap}(\mathrm{b})$的概率密度函数为$p(x)=\frac{1}{2b}\exp\left(-\frac{|x|}{b}\right)$.

拉普拉斯机制只适合于数值型输出的查询函数.许多应用中查询结果为非数值型.对此,McSherry 等人提出了指数机制.其实现核心是设计打分函数u,为输出域中每个输出项打分,分值越高,被选择输出的概率越大.

定理 2[23]. 指数机制. 设定一个打分函数$\mathrm{u}:(\mathrm{D}\times\mathrm{O})\to\mathrm{R}$,如果算法 M 满足下列等式,则 M 满足$\epsilon$-差分隐私.

$$\mathrm{M}(\mathrm{D},\mathrm{u})=\left\{\mathrm{r}:\Pr[\mathrm{r}\in\mathrm{R}]\propto\exp\left(\frac{\epsilon\mathrm{u}(\mathrm{D},\mathrm{r})}{2\Delta\mathrm{u}}\right)\right\} \tag{7}$$

其中$\mathrm{r}\in\mathrm{O}$,表示从输出域O中选择的输出项,$\Delta\mathrm{u}$为打分函数u的全局敏感度.

1.4 组合特性

差分隐私保护具有两个组合性质:序列组合性和并行组合性,在持续监控下仍然适用.

性质 1[22].序列组合性. 设有算法$\mathrm{M}_1,\mathrm{M}_2,\ldots,\mathrm{M}_\mathrm{n}$,其隐私预算分别为$\epsilon_1,\epsilon_2,\ldots,\epsilon_\mathrm{n}$.那么对于同一数据集D,由这些算法构成的组合算法 $\mathrm{M}(\mathrm{M}_1(\mathrm{D}),\ldots,\mathrm{M}_\mathrm{n}(\mathrm{D}))$提供 $(\sum_{i=1}^{n}\epsilon_\mathrm{i})$ -差分隐私.

性质 2[22].并行组合性. 设有算法$\mathrm{M}_1,\mathrm{M}_2,\ldots,\mathrm{M}_\mathrm{n}$,其隐私预算分别为$\epsilon_1,\epsilon_2,\ldots,\epsilon_\mathrm{n}$.那么对于不相交的数据集$\mathrm{D}_1,\mathrm{D}_2,\ldots,\mathrm{D}_n$,由这些算法构成的组合算法 $\mathrm{M}(\mathrm{M}_1(\mathrm{D}_1),\ldots,\mathrm{M}_\mathrm{n}(\mathrm{D}_n))$提供 $(\max\epsilon_\mathrm{i})$-差分隐私保护.

序列组合性表示对同一数据集多个差分隐私保护算法的序列组合算法,所提供的保护水平为全部隐私预算之和.并行组合性表示对不同数据集保护的多个差分隐私保护算法的组合算法,所提供的保护水平为算法中的最低的保护水平,也就是隐私预算的最大值.

1.5 持续监控下差分隐私保护方案评估

满足差分隐私保护的方案在实现隐私保护同时,也要兼顾发布结果的可用性.因此方案的评估包含以下两个方面:

(1) 隐私保护评估:差分隐私处理核心是隐私预算分配和敏感度计算;隐私预算ϵ代表着发布任务的隐私保护程度.隐私预算一旦耗尽,方案的差分隐私保护特性将被破坏.评估方案是否能实现隐私保护,主要是对其差分隐私处理过程中的隐私预算分配策略进行分析,计算发布过程所分配的隐私预算是否符合隐私预算的要求,同时分析敏感度的计算是否正确.

(2) 算法结果误差:为评估方案发布结果的可用性,往往计算真实结果与经差分隐私处理后的结果之间的误差.根据持续监控下问题的不同,误差采用的衡量方法往往不同;如简单聚集统计发布问题中,常对发布结果误差上界进行证明,并在实验评估中采用相对误差[46]、绝对误差[21]等度量标准;而其他问题会根据自身问题特点定义不同的标准.如轨迹保护中,采用 JS 散度[64]、F-Score[64]等可用性衡量标准.

2 持续监控下差分隐私保护方案分类

持续监控下差分隐私保护研究对推动智能信息技术的发展具有重要的意义,其隐私保护问题也是急需解决的问题.本文将现有持续监控下差分隐私保护方案分为 3 类，分别为持续监控下 event 级别保护方案、持续监控下 user 级别保护方案和持续监控下 w-event 级别保护方案.其中持续监控下 event 级别差分隐私保护方案主要围绕单值计数和持续发布、直方图持续发布、heavy hitter 持续监控和位置发布等持续监控问题,针对如何降低时序发布的高敏感度提出相应的解决方案;持续监控下 user 级别和w-event 级别的保护方案主要围绕简单

聚集信息持续发布、分布式数据流阈值函数持续监控、集值对更新发布和轨迹发布问题,针对如何提高时序上隐私预算的利用率,同时降低发布任务的高敏感度提出相应的解决方案.每个分类对应的方案示例见表 1.

Table 1 Existing research on differential privacy under continual observation

表 1 持续监控下差分隐私保护方案分类

主要方案分类	方案示例
持续监控下 event 级隐私保护方案	Two-level[25]、Tree Counter[20,25]、Hybrid[25]、Partition[29]、PeGaSus[30]、SampleEMD[31]、RetroGroup[32]、PDCH-LU[33]、PIM[36,37]、TPL[38]
持续监控下 user 级隐私保护方案	DFT[44]、FAST[46]、CTS-DP[49]、DSAT[50]、SafeZone[55,56]、Prefix[59]、N-Grams[60]、Cluster[61,62]、Homogeneous[63]、Hierarchical[64]、PTCP[65]、
持续监控下w-event 级隐私保护方案	BA[21]、BD[21]、GA+MMD[71]

3 持续监控下 event-级隐私保护方案

Event-级别持续监控隐私保护主要基于count计数的发布任务进行研究,主要包括:单值计数和的持续监控保护、直方图持续发布的隐私保护、heavy hitter 和位置信息发布等特定任务的持续监控保护.

3.1 单值计数和持续监控保护

目前,大部分 event-级别的持续监控保护研究都是基于单值计数和的持续监控保护问题,该问题描述如下:假设存在一个随时间不断更新的事件(event)数据流-基于动态数据抽象得到,其中0代表监控的事件没有发生,1代表事件发生.如何能持续更新发布数据流中1的计数和,同时保证持续发布满足差分隐私保护要求是所需完成的目标.该问题形式化如下:时间序列上对应的事件数据流为$\sigma \in \{0,1\}^N$, $N = \{1,2,3,\dots\}$为一组正整数的集合; $\sigma_T \in \{0,1\}^T$代表σ的长度为T的前缀数据流;$[T] = \{1,2,3,\dots,T\}$表示数据流长度;$\sigma(t) \in \{0,1\}$表示时间 t_i $(i \in N)$ 对应的数据流段中1的计数值.$c(t) = \sum_{i=1}^{t} \sigma(i)$表示数据流$\sigma_t$中的1的计数和.针对该问题,持续监控下隐私保护目标:对每个时间 t_i ,持续更新发布$c(t)$,持续发布过程满足定义2中 event-级差分隐私要求.

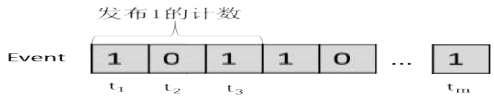

Fig.2 Continual release of the running sum

图 2 单值计数和的持续发布

单值计数和问题在很多应用场景上都可适用.如:特定位置的交通实时监控、特定网页的用户点击行为实时监控、传感器网络中某 IP 地址访问的持续监控、智能基础设施的持续监控、疾病控制中心对患某种疾病人数的持续监控、满足特定谓词条件的持续查询等等.

3.1.1 基本发布方案

方案1是文献[18,22]中差分隐私实现机制的直接应用.其基本思想是根据已知的数据流长度T,在每个发布时刻$t_i \in T$,对前缀数据流σ_t的真实计数和$c(t)$加上独立的拉普拉斯噪声后发布.由于所加噪声量由每个时刻所分配的隐私预算和发布任务的敏感度决定;在该方案中,每个发布时刻 t_i 分配的隐私预算为ϵ;发布任务的敏感度为T,因为数据流中任一个事件的变化对后续每个时刻$\sigma(t_i)$都可能会影响1,所以对发布任务$c(t_i)$的最大影响为T.随机化算法M在每个时间点 t_i 生成一个独立随机噪声$\zeta_{t_i} \sim Lap(T/\epsilon)$,发布结果为 $\alpha_{t_i} = c(t_i) + \zeta_t$.根据差分隐私的组合性质,该方案满足 event 级ϵ-差分隐私.经分析证明,发布结果的渐近误差上界为$O(T/\epsilon)$.方案有如下两个缺点:(1)由于每个发布时刻添加的噪声量与数据流长度T成正比,所以数据流长度越大,算法发布结果的可用性越差.(2)由于要根据数据流长度T来对发布结果添加噪声,该方案只能对长度上限已知的数据流进行隐私保护,无法对无限数据流进行差分隐私保护.

方案2则是差分隐私实现机制[22]与文献[24]思想的结合应用.其基本思想是在每个时刻$t_i \in T$,对 t_i 对应的数据流段中的真实计数值$\sigma(t_i)$而非计数和值$c(t_i)$进行噪声干扰.每个时间 t_i 所分配的隐私预算为ϵ,由于$\sigma(t_i)$只跟当前时间有关,跟前后时间所对应的计数值都无关,所以计数任务的敏感度为1.因此,在每个时刻,随机化算法

M都生成一个独立随机噪声$\gamma_t \sim Lap(1/\epsilon)$,对$\sigma(t_i)$加上拉普拉斯噪声$\alpha_{t_i} = \sigma(t_i) + \gamma_{t_i}$.由于监控目标是每个时刻的计数和$c(t_i)$,所以差分隐私保护机制M在时刻$t_i$的发布结果为$M(t_i) = \sum_{i \le t_i} \alpha_{t_i}$.经分析证明,该方案满足 event 级$\epsilon$-差分隐私,算法渐近误差上界为$O(\sqrt{T}/\epsilon)$.方案具有以下特点:由于每个时刻添加的噪声与数据流长度无关,所以可以处理无限长数据流.但由于算法误差结果与T仍有较大的依赖关系,所以发布结果可用性虽然比方案1有所提高,但依然很差.

Two-level 方案[25]:为降低发布结果误差对 T 的依赖,Two-level 方案对方案 2 进行了拓展,核心思想是对数据流进行分组,每组是一个连续的时间区间.不够一组的元素采用方案 2 进行差分隐私保护;分组之上,将每个组计数和看作一个元素,再次采用方案 2 思想对每组计数进行隐私保护.假设每组大小为 B,数据流长度为 T,在发布时刻t_i,首先将t_i拆分为组:$t_i = qB + r$,先以组为单位,对每个组计数加上$Lap(1/\epsilon)$噪声:$\beta_i = \sum_{k=t-B+1}^{t} \sigma(k) + Lap(1/\epsilon)$.不够一组的元素(r个),每个元素加上$Lap(1/\epsilon)$噪声$\alpha_i = \sigma(i) + Lap(1/\epsilon)$.因此,每个时刻$t_i$的发布结果为$M(t_i) = \sum_{i=1}^{q} \beta_i + \sum_{i=qB+1}^{t_i} \alpha_i$.假设$B = 3$, $t_i = 7$,则分组如图3所示,β_1、β_2和α_7都被加上拉普拉斯噪声$Lap(1/\epsilon)$,输出计数值为$M(7) = \beta_1 + \beta_2 + \alpha_7$.

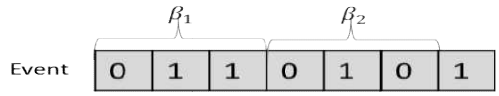

Fig.3 Example of Two-level

图 3 Two-level 方案示例

由于每个时刻t_i的值$\sigma(t_i)$只会影响其所在的组或α_t,Two-level 满足 event 级2ϵ-差分隐私.经分析证明,该方案发布结果渐近误差上界为$O(\sqrt{t/B + B}/\epsilon)$.假设$B = \lfloor\sqrt{T}\rfloor$,则算法渐近误差上界为$O(T^{1/4})$.相比较方案 1 和方案 2,Two-level 方案降低了对 T 的依赖,提高了发布结果的可用性,并且可以对无限数据流进行隐私保护处理.

3.1.2 基于二叉树的发布方案

上述三种基本方案的发布结果误差对数据流长度T都有较大的依赖性,发布结果可用性较低.**Tree Counter**[20,25]方案基于上述问题进行改进,其核心思想是将长度为T的数据流构造成一棵完全二叉树,树的叶节点存储t_i时刻的元素计数值$\sigma(t_i)$,内部节点p_{sum}则存储其覆盖区间[i, j]的叶节点的计数和,其值为$p_{sum} = \sum_{k=i}^{j} \sigma(k)$.因此,一个高度为$l$的$p_{sum}$节点对应$2^l$个叶节点的计数和.如图 4 所示,[1-4]代表的p_{sum}节点高度为2,代表数据流[1,4]区间内元素的计数和.

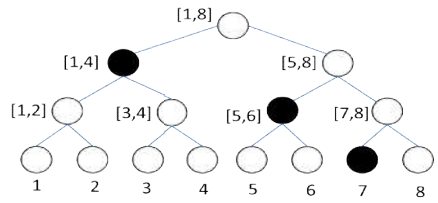

Fig.4 Example of Tree Counter [25]

图 4 Tree Counter 方案示例 [25]

对任意时刻t_i,$c(t_i)$都可以表示为一组p_{sum}节点值的求和.如图4中,当$t_i = 7$时,$c(7)$可以由p_{sum}节点[1,4]、[5,6]和$\sigma(7)$求和得到.因此,在每个发布时刻t_i,先找出这组p_{sum}节点,如图 4 中标黑色的节点即为对应的p_{sum}节点组.根据二叉树的性质,t_i时刻发布值的计算中参与求和的p_{sum}节点数最多为$\log T + 1$.根据p_{sum}节点计算$c(t_i)$,发布任务的敏感度为$\log T + 1$.算法为每个参与求和的p_{sum}节点添加满足$lap((\log T + 1)/\epsilon)$的噪声,即可保证发布结果满足 event 级$\epsilon$-差分隐私.可以看出,采用二叉树的发布方案使$t_i$时刻发布算法的敏感度由$O(T)$降为了$O(\log T)$,降低了发布结果中添加的噪声量.经分析证明,Tree Counter 发布结果的渐近误差上界为$O((\log T)^{1.5}/\epsilon)$,进一步提高了发布结果的可用性.但其有以下两个缺点:(1)处理的数据流必须是有限长度,不能处理无限长的情况;(2)提供的是 event 级别的隐私保护,隐私保护级别较弱.

针对 Tree Counter 第一个缺点,文献[25]以 Tree Counter 和 two-level 两种机制为基础,提出一种混合机制 **Hybrid** 机制,该方案可以实现对无限数据流 event 级的差分隐私保护.其主要思想是将所有待发布的时间分为两种类型:第一类是可以表示为 2 的幂集的时间点集合;第二类是非 2 幂集的时间点集合.对每个发布时刻 t_i,随机化发布机制 H 根据 t_i 类别采用不同的保护机制.如果 t_i 为 2 的幂级,采用L机制(改进的 two-level 机制)发布;如果 t_i 不为2的幂级,采用 M (Tree Counter 机制)发布. L 机制算法思想类似于 two-level 机制,不同的是 two-level 机制采用定长分组,而L机制则采用变长分组,只要时间点为2的幂集($2^k(k \in Z)$),就把此前的时间区间分为一组.基于分组,添加敏感度为1,隐私预算为ϵ的拉普拉斯噪声$lap(1/\epsilon)$.对非2的幂集的时刻 t_i,如 $t_i \in (2^k, 2^{k+1})$,则采用长度上限为$T = 2^k$的M (Tree Counter)机制发布方案.假设$\tau = t_i - T$,则 t_i 时刻的发布为$H(t_i) = L(T) + M_T(\tau)$.为更好地理解该方案,给出以下示例:假设为 $t_i = 8$,因为 $t_i = 2^3$,所以在计算 t_i 发布结果的过程中,共分 4 组,每个组的分界点分别为{1,2,4,8}.对每个组,都需要对组内计数和添加$lap(1/\epsilon)$.因此发布结果为$L(t) = \sum_{i=1}^{8}\sigma(i) + 4 \cdot lap(1/\epsilon)$;假设 $t_i = 12$,因为 $t_i \in (2^3, 2^4)$,则对$[1,8]$采用L机制发布,而对于$[9,12]$采用上限为$T = 8$的M机制进行发布.因此, $H(12) = L(8) + M_T(4)$.经分析证明,混合机制渐近误差上界为$O((\log t)^{1.5}/\epsilon)$,满足 event 级别差分隐私.Hybrid 机制既提高了发布结果的可用性,同时也解决了 Tree Counter 只能处理有限长数据流的问题.

虽然上述改进方案提高了发布结果的可用性,也解决了处理的数据流长度受限的问题,但还有以下缺点:(1)由于发布结果包含噪声,所以出现发布值不为整数,且相邻时刻发布结果的差值不满足实际统计约束的不一致问题.(2)由于数据流实时处理的特点是一次遍历,所有数据使用后并不保存,因此面向数据流的数据统计涉及到中间统计值的保存,如果中间值被攻击者捕获,则上述隐私保护方案不能提供有力的保护.

针对(1),文献[25]对单值计数和持续发布中的一致性问题进行了定义:对任意数据流σ,一个随机化的发布机制M如果是一致性的,必须满足如下条件:对任意时刻 t_i,$Pr(M(\sigma)(t_i) - M(\sigma)(t_{i-1}) \in \{0,1\}) = 1$.同时,Hybrid Mechanism 对一致性问题进行了处理:对所有的时刻 t_i,如果 $M(\sigma)(t_i) > M(\sigma)(t_{i-1})$,则 $M(\sigma)(t_i) = M(\sigma)(t_{i-1}) + 1$,否则$M(\sigma)(t_i) = M(\sigma)(t_{i-1})$.针对(2),文献[20,25]对 Tree Counter 和 Hybrid 两种机制进行改进,改进后的方案可以实现 Pan-privacy,其含义是指攻击者即使攻击获取了数据流处理过程中的中间值,结合持续观察的发布结果,也无法判断个体事件是否发生,从而实现了隐私与安全的结合.

在很多实时监控的应用场景中,往往近期发生的事件对监控更有意义,而发生时间较远的事件对监控价值较低.针对这个问题,文献[28]提出基于衰减的单值计数和持续监控保护方案 **DecayedSum**,并分析证明了方案中发布结果误差的上界.方案中提出了三种数据衰减模型的差分隐私保护机制,分别是基于滑动窗口的保护、基于指数机制衰减和基于多项式衰减的保护.第一种方案每次发布只对最近窗口(宽度为W)内的数据流进行统计,并保证发布的单值统计和$F_w(j,W) = \sum_{i=j-w+1}^{j}\sigma(i)$ 满足 event 级ϵ-差分隐私,其发布结果的误差上界为$O(\log W/\epsilon)$.第二种方案保证基于指数衰减模型的单值计数和$F_e(j,\alpha) = \sum_{i=1}^{j}\sigma(i)\alpha^{j-i}$的持续发布满足 event 级$\epsilon$-差分隐私,发布结果的误差上界为$O(\log(\alpha/(1-\alpha))/\epsilon)$,其中 α为指数衰减因子.第三种方案保证基于多项式衰减模型的单值计数和$F_p(j,c) = \sum_{i=1}^{j}\sigma(i)/(j-i+1)^c$ 的持续发布满足 event 级ϵ-差分隐私,发布结果误差上界为$O((1/\epsilon)(1/(c\beta^2))\log(1/(1-\beta)))$,其中$c$为多项式衰减因子,$\beta$为错误控制参数.三种隐私保护方案都是基于Tree Counter方案的拓展.在第一种方案中,将整个时间序列T按照窗口宽度W进行划分,每个窗口内维护一个高度为$\log W + 1$的完全二叉树,整个时间序列分成多个二叉树的集合.由于每个 j 时刻的发布结果只考虑W内的计数统计,所以最多只需$\log W + 1$个节点参与统计,因此,敏感度为 logW+1,所需添加的噪声为$Lap((\log W + 1)/\epsilon)$.后面两种方案都是整个时间序列维护一棵二叉树,为确定添加的噪声,必须计算基于两种衰减函数的敏感度λ_1和λ_2.基于所分配的隐私预算和敏感度,两种方案中添加的噪声分别为$Lap(\lambda_1/\epsilon)$和$Lap(\lambda_2/\epsilon)$.DecayedSum 可以实现对无限长数据流的隐私保护,更符合实时监控的场景.

3.1.3 基于分区的发布方案

基本方案和二叉树方案中没有考虑数据流自身特点对发布结果的影响.文献[29]提出了一种适合稀疏数据流的自适应分区方案 **Partition**.该方案通过结合数据流的变化特点降低发布问题的敏感度,提高发布结果的可

用性. Partition 基本思想为按提前设定的阈值,将长度为T的稀疏数据流划为一组连续分区的集合,每个分区的计数和基本相同,为接近于阈值的一个统计值.假设分区后分区个数为m(m ≪ T),用分区的计数和来代替 Tree Counter 的叶节点,然后基于 Tree Counter 基本思想进行隐私保护.由于持续监控发布结果基于分区的统计值进行计算,分区个数m又远远小于T,因此发布任务的敏感度降低,提高了算法的可用性.算法过程描述如下:对每个区j,首先设置一个$count_j$,用来记录分区j内的计数和;从j-1 分区的下个时刻t 开始,依次统计分区 j 的真实计数和$count_j$,:$count_j$ += σ(t);计算当前分区j 加噪后的计数值:$\widehat{count}_i = count_j + Lap(1/\epsilon)$;如果$\widehat{count}_i$大于该分区设定的一个阈值$\hat{T}_j$,则分区j生成;重复执行这个分区的过程,最终将整个数据流分为一组连续的m个分区的集合$P = \{[1, s_1], [s_1 + 1, s_2], \ldots, [s_{m-1}, T]\}$.分区示例过程如图 5:

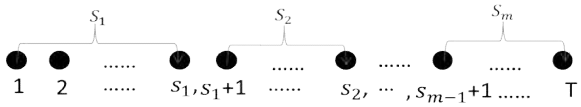

Fig.5 Example of adaptive partition

图 5 Partition 自适应分区示例

在发布某个时刻 t 的统计值时,结合 Tree Counter 的思想,将每个分区的统计值作为叶节点,构造完全二叉树,发布结果的计算就依赖于每个分区的结果统计.因此,发布的结果误差降低了对数据流长度 T 的依赖,极大地提高了算法的可用性.经分析证明,算法的渐近误差上界降为$O(\log T + (\log^2 n)/\epsilon)$,其中 n 是数据流中 1 的总计数,在稀疏数据流中n ≪ T.因此,该方案只适用于稀疏数据流的应用环境.

文献[30]提出的**PeGaSus**方案同样采用分区思想对面向数据流的持续监控问题进行了研究. PeGaSus 方案由三个构件组成: Pertuber、Grouper 和 Smoother.为保证ϵ-差分隐私,隐私预算 ϵ 分为两部分:$\epsilon = \epsilon_p + \epsilon_g$.$\epsilon_p$用于 Pertuber,$\epsilon_g$用于 Grouper,Smoother 是后处理过程,不需要分配隐私预算.

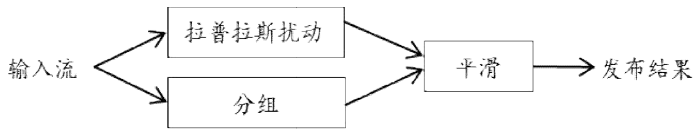

Fig.6 Process of PeGaSus [30]

图 6 PeGaSus 过程图[30]

Pertuber 用来对每个时刻t的计数进行加噪处理:$\sigma(t) + lap(1/\epsilon_p)$.Grouper 根据数据特点对数据进行自适应的分区,分区原则是将统计值变化不大的连续时间分成一个区,数据流被分成一组分区的集合.为了捕捉统计值的变化,方案设计一个偏差函数$dev(C_t[G]) = \sum_{i \in G} |\sigma(i) - \sum_{i \in G} \sigma(i)/|G||$用来获取分区值的变化信息;如果变化大于设定的阈值θ,则当前分区分配完毕;从下个时刻开始,重新开始新分区的划分.分区后,Smoother 结合分区结果和 Pertuber 加噪后的统计值,采用分区平均值、中位数等平滑处理技术,提高特定查询任务发布结果可用性.

PeGaSus 与 Partition 分区策略不同点在于:(1)Partition 是将数据流分为一组连续分区,每个分区内的计数和基本相同; (2)而 PeGaSus 分区的思想是将计数值相似的的时间点分到一个分区,不同分区间计数和可能变化较大; (3)Partition 基于分区的计数和加噪,而 PeGaSus 则是对每个时间点的计数值加噪;(4) PeGaSus 采用了偏差函数帮助 Grouper 进行分区,而 Partition 是通过提前设定的一个阈值进行分区;(5)Partition 只针对单值计数和发布问题进行隐私保护,但 PeGaSus 能实现面向数据流的多种持续查询任务类型的隐私保护,如单目标的持续监控、具有层次关系的多目标持续监控、单点数据的持续查询和多种时间跨度的持续查询等多种查询类型.

PeGaSus 方案中,因其支持多种查询类型,提出对具有层次关系的查询任务关系进一步降低查询任务的敏感度,提高发布结果的可用性.文献[31]则也利用查询任务之间的关系来降低查询结果的噪声.文中提出了两种方案: SampleDP是针对查询任务的步长满足一定约束条件(长步长是短步长的整数倍)的情况下,采用动态规划算法找出能使添加噪声量最小的步长组合;而 SampleEMD是将第一种方案扩展到更一般的情况:采用 EMD 相

似性计算方法来选取与原有步长集分布相似的抽样步长集,达到降低噪声，提高查询结果可用性的目的.

3.2 直方图发布的持续监控保护

直方图是一种直观的数据分布表示形式,它按照某属性或属性集将数据集信息划分成不相交的桶,每个桶内有一个统计数字表示其特征.直方图的持续发布是基于时间序列上的动态数据,持续发布直方图统计信息.图7表示在时间序列上,持续监控位置 L1 到 L7 上人数的直方图示例.

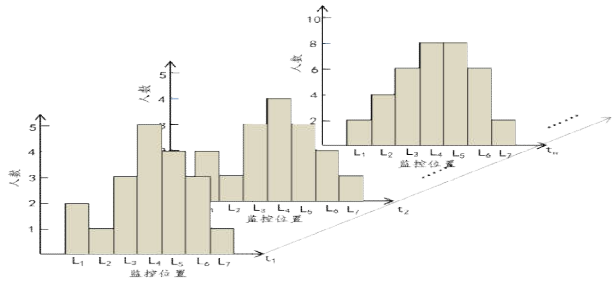

Fig.7 Example of histogram publication under continual monitoring

图 7 持续监控下直方图发布示例

文献[32]针对无限数据流中直方图的持续发布问题提出一种满足 event 级的隐私保护方案 RetroGroup. 跟单值计数和持续监控保护方案一样,RetroGroup 在时间序列上每个发布点所分配的隐私预算也为ϵ.但为了提高时序上发布结果的可用性,RetroGroup 采用回溯分组来降低持续发布问题的敏感度.其基本思想是根据相邻时间直方图的相似性,将发布数据相似的连续时间分为一组,以组为单位对要发布的直方图加拉普拉斯噪声.因此,RetroGroup 方案包含两个过程:相似性计算和直方图发布.为满足ϵ差分隐私,相似性计算过程分配的隐私预算为ϵ_1,发布过程分配隐私预算为$\epsilon_2 = \epsilon - \epsilon_1$.根据差分隐私的序列组合性质,该方案满足$\epsilon$-差分隐私.

假设用$\widetilde{\mathrm{H}}_{\mathrm{i}}$代表$\mathrm{t}_{\mathrm{i}}$时刻要发布的直方图,$\widetilde{\mathrm{H}}_i^j$表示直方图的第j个桶的统计值.在每个时刻$\mathrm{t}_i$,计算直方图的每个桶$\mathrm{B}_j$的统计值 $\widetilde{\mathrm{H}}_i^j$和 $\widetilde{\mathrm{H}}_{i-1}^j$的差异d[j],将d[j]变化不大的连续时间分为一组$\mathcal{G}(\mathrm{B}_j)$.为了提高相似性计算效率,方案采用了抽样技术,该技术在提高效率的同时也具有增大隐私预算的效用,从而会降低拉普拉斯噪声.但抽样也会带来抽样误差,因此方案中对抽样误差和差分隐私误差做了定量化的分析.基于相似性计算进行分组后,每个组包含多个时间点,大小为$|\mathcal{G}(\mathrm{B}_j)|$,对组内的平均计数值添加拉普拉斯噪声后的发布结果为 $\tilde{\mathrm{H}}_i^j = \sum_{H_k \in \mathcal{G}(\mathrm{B}_j)} H_k / /\mathcal{G}(\mathrm{B}_j) /+\mathrm{Lap}(1/(\mathcal{G}\mathrm{B}_j \epsilon 2))$.可以看出,采用回溯分组策略后,每组添加的拉普拉斯噪声量由 $\mathrm{Lap}(1/\epsilon 2\ )$降为了 $\mathrm{Lap}(1/(|\mathcal{G}(\mathrm{B}_j)|\epsilon_2))$,发布结果可用性得到了提高.

RetroGroup 与 PeGaSus 分区有些相似,都是将数据变化不大的时间序列分为一组;但其加噪方式不一样,RetroGroup 是基于分组加噪,而 PeGaSus 是基于每个时间点加噪,而后针对特定查询任务进行分组平滑处理.

3.3 Heavy hitter的持续监控保护

Heavy hitter 的持续监控是指对数据流中的元素及其频数进行统计,实时发布超出阈值的元素及其频数，即 heavy hitter.在该问题中, heavy hitter 元素及其频数都是需要保护的隐私.文献[33]针对三种监控场景提出了对应的隐私保护方案:面向单数据流 heavy hitter 持续监控的隐私保护方案 PMG、面向单数据流滑动窗口内 heavy hitter 持续监控的隐私保护方案 PCC 和面向分布式数据流滑动窗口内 heavy hitter 持续监控的隐私保护方案 PDCH-LU.

PMG 方案基于未做隐私保护的面向数据流中 heavy hitter 统计的算法 MG[34],采用差分隐私技术对其进行隐私保护.在 MG 算法中,给定数据流长度 T 和错误控制参数λ,算法执行过程中会持续维护一个长度为$\beta = \mathrm{O}(1/\lambda)$的计数器组,通过计数值可以统计出 heavy hitter 及其频数.为满足 event 级ϵ差分隐私,PMG 在每个发布时刻分配的隐私预算为ϵ,由于 MG 算法的发布敏感度为 β,因此对发布结果添加隐私预算为ϵ,敏感度为β的噪声,即可满足ϵ差分隐私保护.

基于PMG,PCC方案按照时间宽度W将整个数据流分为了一系列的窗口.在每个窗口内,采用二叉树结构统计其窗口内的 heavy hitter,其中二叉树叶节点代表$w_0 = \lambda W/4$个时间点的统计值,内部节点代表其覆盖的叶节点的时间区间内的统计值.如果持续发布W滑动窗口内的 heavy hitter 统计信息,需要基于这组二叉树进行统计.而每次的发布参与统计的二叉树的节点数最多为$\log W + 1$.

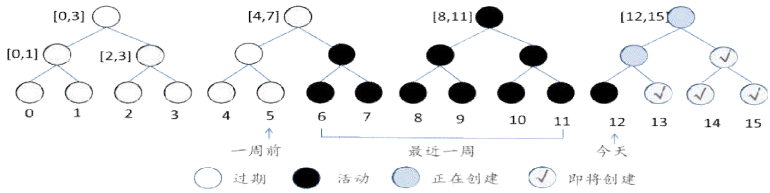

Fig.8 Example of PMG (W=7)[33]

图 8 PMG 方案示例(W=7) [33]

根据二叉树中参与统计的节点创建时间,所有节点可以分为图 8 中所示的 4 类:过期、活动、正在创建和即将创建.由于实时发布只关注W宽度的统计值,因此每次发布时最多只会关注最近两个窗口内的二叉树的统计信息.只有活动节点才需要占用内存空间.因此,PCC 的空间代价为$O((1/\lambda)\log^2(1/\lambda))$,计算性能较快.同时由于每个节点记录$w_0$个时间间隔,每隔$w_0$才需要和分布式系统的中心节点进行通信,因此降低了通信带宽.

PDCH-LU[33]则基于PCC做了进一步的改进,将其改为面向分布式环境.为了进一步降低通信带宽,PDCH-LU 首先采用 Lazy 更新策略降低总的通信代价:只有各分布式节点的统计值更新够大的情况下才与中心节点进行统计信息的通信;其次,在每次通信时,采用 Bloom Filter 技术[35]降低每次的通信带宽.在隐私保护方面,各分布式节点的单数据流采用PCC的隐私保护方案,为了对统计信息来源的各分布式节点进行保护,该方案采用了文献[44]中的加密保护方案.

文献[26]基于抽样和随机响应技术对数据流中 heavy hitter 元素频数信息的持续发布进行差分隐私保护.该方案首先对所有元素集合X随机抽样出m个元素,其计数值存放在数组M中.统计之前,采用随机响应技术对数组M的每一个计数值初始化为:$b_x \sim D_0(\epsilon)$, $D_0(\epsilon)$指以等概率初始化为'0'或'1';统计中,对数据流中出现的每个元素,如果在 M 中,则其对应的统计值做更新:$b_x \sim D_1(\epsilon)$, $D_1(\epsilon)$指产生1的概率为$1/2 + \epsilon/4$,产生0的概率为$1/2 - \epsilon/4$.基于M,持续计算并发布 heavy hitter 元素频数.

文献[27]同样对数据流中heavy hitter元素频数的持续发布进行差分隐私保护.主要实现技术是数据流处理的sketch技术和指数机制.由于基于sketch的数据流处理会维护一个记录中间统计值的sketch向量sk(a),为实现差分隐私,先采用服从$\mu_\epsilon = \exp(\epsilon/4 \cdot q(sk(a), sk(a)^{priv}))$的指数机制噪声对sk(a)进行初始化, $q(sk(a), sk(a)^{priv})$是指数机制的效用函数.然后基于初始化后的sk(a),采用数据流统计算法对sk(a)中的元素统计值进行更新,最后再对sk(a)进行满足μ_ϵ指数机制的噪声处理.基于噪声处理后的sk(a),持续发布 heavy hitter 元素频数信息.

文献[26,27]都考虑了攻击者获取到中间统计值时的隐私保护处理,所以方案满足 pan-privacy[19],实现了差分隐私与安全的结合,其隐私保护强度更高.两种方案的区别是文献[26]只能对计数值增量更新情况下进行隐私保护,而文献[27]却可以处理全动态更新情况下(计数值可以增加,也可以减少)的隐私保护.

3.4 位置发布的持续监控保护

文献[36,37]提出一种基于时序关联的位置持续发布的隐私保护方案PIM.方案中相邻时刻用户位置的关联关系用马尔科夫模型建模和预测.在每个时刻,找出用户的"δ-位置集"ΔX(t时刻用户可能会存在的位置及其对应的概率).差分隐私发布机制保证,在每个时刻t,用户的真实位置与ΔX中任一位置出现的概率基本一样.即使攻击者能根据转移概率推测出ΔX,也不能发现用户的真实位置.具体实现方案是在每个时刻t,基于$t-1$时刻的后验概率p_{t-1}^+和用户的马尔科夫转移概率矩阵M,计算用户 t 时刻转移位置的先验概率向量$p_t^- = p_{t-1}^+ M$;根据该向量构建t时刻的"δ −位置集"ΔX_t,然后计算发布任务的敏感度K,并产生一个满足ϵ_t的噪声,对t时刻的用户位置加噪后发布.为进一步提高发布结果的可用性,在计算敏感度时,方案采用平面各向同构机制对K进行各项同性转

换,并利用K-范数机制来降低发布所需噪声.

文献[38]则针对位置聚集统计信息持续发布的隐私保护问题,对时序关联所造成的隐私泄露做了定量的分析.用 TPL 代表该方案.方案问题场景如下:假设攻击目标u_i在 t 时刻的位置为l_i^t,攻击者A_i^T已经掌握除 l_i^t 之外的所有用户的时序数据的关联信息$D_k^t = D^t - \{l_i^t\}$.关联信息采用马尔科夫链建模,分为前向马尔科夫转移概率矩阵(P_i^B)和后向马尔科夫转移概率矩阵(P_i^F),其示例见图 9.

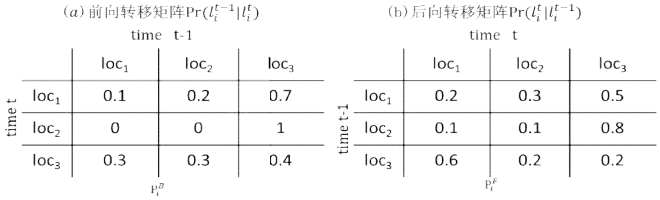

Fig.9 Example of Temporal Correlations [38]

图 9 时序关联关系示例[38]

基于$A_i^T(P_i^B, P_i^F)$,方案中对时序数据相关性所造成的隐私泄露量(TPL(M^t))给出了定量化的计算公式,即为所有用户中最大的隐私泄露风险值.该隐私泄露量的计算包含前向转移关系隐私泄露(BPL)的计算和后向转移关系隐私泄露(FPL)的计算,为了提高计算效率,方案将隐私泄露量的求解问题转换为一个线性分式规划问题,并提出了多项式时间内的求解算法.

上述方案都是考虑了时序数据相关性的隐私保护方案.目前,关联数据发布的隐私保护研究提出了新的隐私保护模型 Pufferfish[39,40]和 Blowfish[41].同时也有一些面向关联数据发布的差分隐私保护文献,如文献[42,43]针对静态社交网络和关联数据集的发布问题提出了相应的解决方案.其主要思想是对数据之间的关联进行建模,根据模型确定关联敏感度,基于关联敏感度确定关联数据的噪声添加方案.虽然这些方案不是直接针对时序关联数据发布的隐私保护方案,但可以拓展到持续监控场景下.

3.5 Event级隐私保护方案小结

总结来说, 持续监控下 event 级隐私保护方案核心都是围绕如何提高时序数据发布的高可用性问题进行研究,其目标都是满足 event 级ϵ-差分隐私的前提下,如何能降低发布结果误差,提高发布结果可用性.本小节从方案优缺点、发布结果误差等方面对现有 event 级隐私保护方案进行总结对比分析,详细见表 2; 然后指出现有方案的不足及未来研究趋势.

(1) 从持续监控生命期上看,所有 event 级别持续监控保护方案中,不需要在时序上进行隐私预算分配.但由于时序发布任务的高敏感度,数据流的持续监控保护长度受限.如方案1和 Tree Counter 中,必须根据数据流长度 T 添加噪声,所以只能保护定长数据流;其余方案虽然噪声添加方案与数据流长度无关,但随着处理数据流长度的增加,大部分方案发布结果可用性变差.只有 Decayed Sum 中隐私保护方案更符合实时监控特点,能实现对无限长数据流的监控保护.因此,如何在保证 event 级隐私和高可用性的前提下,实现对无限长数据流的持续监控保护值得进一步研究.

(2) 从降低敏感度技术看, event发布方案主要采用分区技术或二叉树技术来降低发布敏感度.其中基于分区的方案有 Two-level、Hybrid、Partition、PeGaSus、RetroGroup;虽然都是分区,但分区原则不一样,Two-level 是定长分区、Hybrid 是非定长分区、Partition 是每个分区计数和大致相同,而 PeGaSus 和 RetroGroup 则是将数据变化不大的连续时间分为一组.基于二叉树的方案有 Tree Counter、Hybrid、Decayed Sum、PDCH-LU.所有的分区方案和二叉树方案只是用于简单计数统计的查询任务;虽然采用降低敏感度技术提高了发布结果可用性,但随着数据流的增加,发布结果的可用性依然会变差. 如何结合特定持续发布任务设计更好的降低敏感度方案来提高发布结果可用性值得进一步研究.

(3) 在 PeGaSus 和 RetroGroup 中分区方案,采用了平滑技术提高发布结果可用性,如采用分区内平均值或中间值来对发布结果进行近似处理.分区技术是为降低时序上的噪声误差的常用技术,如何结合更好

的平滑技术提高基于分区的发布结果可用性值得进一步研究.

(4) 从数据处理方式上看,根据时序数据的处理方式,方案分为离线处理和在线处理.所有方案都可实现实时在线处理.结合特定持续发布任务实现在线处理的 event 级隐私保护方案也值得进一步研究.

Table 2 Comparation of schemes on event-level privacy under continual monitoring

表 2 Event 级别持续监控保护方案对比

算法	主要优点	主要缺点	保护等级	算法误差
方案1[18,22]	实现简单,没有中间值的存储	算法可用性差,只能处理定长数据流,对 T 线性依赖	event 级	$O(T/\epsilon)$
方案2[24]	实现简单,可以实现无限长数据流处理	算法可用性较差,需存储中间值,对 T 仍有较大依赖	event 级	$O(\sqrt{T}/\epsilon)$
Two-level[25]	采用分区提高可用性,可以实现无限长数据流处理	算法可用性较差,需存储中间分组节点统计值	event 级	$O(\sqrt{t/B+B}/\epsilon)$
Tree Counter[20]	采用二叉树结构提高了算法的可用性	只能处理定长数据流,需存储二叉树内部节点p_{sum}值	event 级	$O((\log T)^{1.5}/\epsilon)$
Hybrid[25]	采用混合策略提高算法可用性,能处理无限长数据流,具有一致性处理	需存储中间节点分组计数值和p_{sum}值	event 级	$O((\log t)^{1.5}/\epsilon)$
Decayed Sum[28]	离当前时间越近的数据占的统计比重越大,采用二叉树结构提高可用性	只适用于单值计数和持续发布问题,不能适用于复杂统计分析发布	event 级	$O(\frac{1}{\epsilon}\log W)$ $O(\frac{1}{\epsilon}\log\frac{\alpha}{1-\alpha})$ $O(\frac{1}{\epsilon}\frac{1}{c\beta^2}\log\frac{1}{1-\beta})$
Partition[29]	采用自适应分区进一步提高了算法可用性,能处理无限长数据流	只适用于稀疏数据流的发布统计	event 级	$O(\log T+(\log^2 n)/\epsilon)$
PeGaSus[30]	(分区+平滑技术+查询任务关系)策略提高了可用性,能处理无限长数据流	数据变化较快情况下,可用性提高不明显	event 级	$O(\log T+(\log^2 n)/\epsilon)$
SampleEMD[31]	利用查询任务关系提高发布可用性	只适合多种查询任务步长之间具有关系的发布分析	event 级	无分析
RetroGroup[32]	采用相似性计算和回溯分组,降低了发布敏感度	统计值变化较快情况下可用性提高不明显	event 级	无分析
PDCH-LU[33]	能处理无限长数据流,适用于分布式环境	只适用于 heavy hitter 特定问题发布	event 级	无分析
文献[26]	能在增量更新下处理无限长数据流,能实现 pan-privacy	只适用于简单聚集统计问题	event 级	无分析
文献[27]	能在全动态更新下处理无限长数据流,实现 pan-privacy	只适用于简单聚集统计问题	event 级	无分析
PIM[36,37]	利用数据相关性提高隐私保护强度	计算敏感度耗费时间	event 级	无分析
TPL[38]	对数据相关性所造成的隐私泄露进行定量分析	预处理过程耗费时间	event 级	无分析

4 持续监控下 user-级隐私保护方案

User 级的持续监控保护方案主要围绕简单聚集统计信息的持续发布、分布式数据流的阈值函数持续监控、集值对的增量更新发布、轨迹数据发布等几个问题进行分析总结.

4.1 简单聚集统计信息的持续发布保护方案

简单聚集统计指基于 count、sum、average 等聚集函数的信息统计.实时发布简单聚集统计信息对很多重要的数据挖掘应用具有很重要的意义.如图 10 所示,假设一个 GPS 服务提供商实时收集用户位置、速度、活动类型等信息,然后统计出如某时段特定区域的用户数等统计信息,提供给第三方研究机构,第三方基于这些信息可以挖掘用户的常去区域,公众的兴趣,路段的交通堵塞等信息.但第三方机构往往是不可信的.因此,在简单聚集统计信息的持续发布过程中,有必要对其进行隐私保护.本节主要针对简单聚集统计信息的持续发布问题,对现有的几种 user 级隐私保护方案进行总结和对比分析.

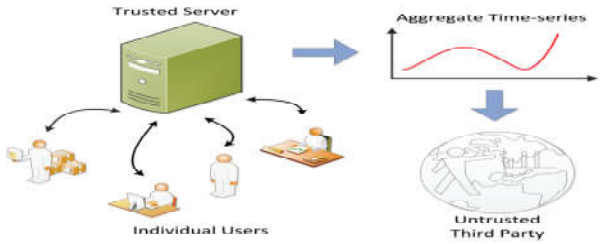

Fig.10 Contimual release of simple aggregation statistics [46]

图 10 简单聚集统计信息持续发布[46]

4.1.1 基本发布方案

基本发布方案是差分隐私实现机制[22]的直接应用,为满足 user 级的差分隐私,基本方案是将隐私预算 ϵ 平均分配到时间序列上每个发布时间点,假设时间序列长度为T,则每个发布点所分配的预算为ϵ/T.根据差分隐私的序列组合性质,则整个时间序列上的持续发布满足ϵ差分隐私.由于每个发布点所分配的隐私预算跟T成反比,发布结果误差为$\theta(\mathrm{T})$,发布结果可用性较低.为提高可用性,降低误差,很多文献基于采样的方法,选出有代表性的采样点进行发布,具体方案如下.

4.1.2 基于傅里叶级数变换的采样发布方案

文献[44]提出一种基于傅里叶级数变换的采样发布方案 DFT,该方案满足 user 级隐私.其基本思想是:选取$\mathrm{d}(\mathrm{d} \ll \mathrm{T})$个最具有代表性的时刻进行发布,其他时刻的结果用相邻时刻结果近似计算获得.对每个代表性的时刻添加满足ϵ/d的拉普拉斯噪声,整个发布结果序列满足ϵ差分隐私.具体过程为根据用户预先设定的傅里叶系数个数d,先对时间序列上的真实统计值X做离散傅里叶变换,取其前d个傅里叶系数F^d;第j个傅里叶系数可表示为$\mathrm{DFT(X)}_j = \sum_{i=0}^{T-1} e^{\frac{2\pi\sqrt{-1}}{T}ji} x_i$;然后对$\mathrm{F}^d$中每个系数添加满足隐私预算为$\epsilon/\mathrm{d}$的拉普拉斯噪声,生成加噪后的$\tilde{\mathrm{F}}^d$.用$\mathrm{PAD}^T(\tilde{\mathrm{F}}^d)$代表向$\tilde{\mathrm{F}}^d$中填充了$\mathrm{T}-\mathrm{d}$个 0 后的长度为 T 的序列;第j个时刻的发布结果 R 通过对$\mathrm{PAD}^T(\tilde{\mathrm{F}}^d)$进行反傅里叶变换获得$\mathrm{IDFT(X)}_j = (\sum_{i=0}^{T-1} e^{-\frac{2\pi\sqrt{-1}}{T}ji} x_i)/T$.该方案优点是将发布结果的误差由$\theta(\mathrm{T})$降为$\theta(\mathrm{d})$,但缺点是只适用于时序长度 T 已定的情况,不能处理无限长数据流.同时,由于存在傅里叶变换和逆变换等计算,该方案并不能适用于面向数据流的实时监控,只能进行离线时序数据的处理.

4.1.3 基于 PID 机制和滤波技术的发布方案

文献[46]提出了一种基于 PID 机制[45]的自适应抽样发布方案 FAST,该方案满足 user 级隐私保护.与 DFT 不同,该方案可以处理实时数据流,效率较快.FAST 核心策略是采用自适应采样技术和滤波技术来提高发布结果的可用性,其基本框架如图 11 所示,其中自适应采样技术是为了降低时间序列上总的隐私预算消耗,而滤波技术是为了提高每个发布点上发布结果的可用性.

自适应抽样机制的核心思想是设定整个时序发布次数上限值为$\mathrm{M}(\mathrm{M} \ll \mathrm{T})$,基于 PID 控制原理,根据历史采样频率和反馈信息来调整新的抽样频率,选择代表性的抽样点进行发布,使得最终的采样发布次数保持为M.由此每个采样点所分配到的隐私预算由ϵ/T增大为ϵ/M,提高了时序上隐私预算的利用率.而 PID 技术的实现核心是设置反馈信息,定义为每个采样时刻的发布结果误差值$\mathrm{E}_{k_n} = |\hat{x}_{k_n} - \hat{x}_{k_n}^-|/\max\{\hat{x}_{k_n}, \delta\}$, k_n表示采样时刻, $\hat{x}_{k_n}$是采样时刻的发布值, $\hat{x}_{k_n}^-$是观测值,根据误差值调整当前的抽样频率.

滤波技术的核心思想是建立一个状态空间模型,根据模型预测值和观测到的噪声统计值,采用后验估计对要发布的结果进行校正,提高发布结果的可用性.所建立的状态空间模型包括相邻时刻统计值预测模型和噪声处理模型,相邻时刻处理模型为$x_k = x_{k-1} + \omega, \omega \sim N(0, Q)$, x_k代表k时刻的真实值,ω为高斯白噪声,也称为处理噪声,Q为方差.噪声处理模型为$z_k = x_{k-1} + \upsilon, \upsilon \sim Laplace(0, M/\epsilon)$,$z_k$为k时刻加噪后的统计值,$\upsilon$是加入的拉普拉斯噪声.在每个抽样发布时刻k,基于状态空间模型获得x_k和z_k,然后采用卡尔曼滤波的后验估计方法,对要发布的结果$\hat{x}_k$进行校正.具体步骤为:根据先验估计得到的k时刻噪声值$\hat{x}_k^- = \hat{x}_{k-1}$,然后基于$\hat{x}_k^-$和$z_k$的值,采用梯度下降算法调整卡尔曼收益参数$\mathrm{K_k}$,目标是使得k时刻的后验估计值$\hat{x}_k$的方差最小.然后根据$\mathrm{K_k}$可以计算出 k 时刻的最优发布值$\hat{x}_k = \hat{x}_k^- + \mathrm{K_k}(z_k - \hat{x}_k^-)$进行发布.

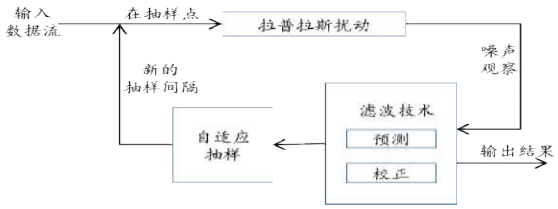

Fig.11 Framework of FAST [46]

图 11 FAST 框架[46]

文献[47]将 FAST 应用到了二维空间的区域实时计数监控问题中.文献提出了两种估计算法来降低噪声误差:一个是时序估计算法,核心思想就是 FAST 的应用,基于采样和卡尔曼滤波后验估计方法去降低时序数据发布的总误差;另一个是空间估计算法,基于 quadtree 建立一个空间索引结构,基于该结构对具有相似统计值的区域进行分组,然后基于分组添加拉普拉斯噪声,降低由于数据稀疏所造成的添加噪声过大的问题.

文献[48]则将 FAST 应用到了用户的网页浏览行为的持续监控问题中.文献将网页监控问题分为了单值持续监控和多值持续监控两种情况.在单值持续监控方案中,基于 FAST 思想,建立状态空间模型,采用卡尔曼滤波方法对观测的噪声值进行校正,提高发布结果可用性.在多值目标监控中,时序处理模型结合了网页的马尔科夫转移矩阵来进行预测.虽然方案满足 user 级隐私保护,也提高了发布结果的可用性,但还有一些缺点可进一步改进.如随着监控的网页数目m的增加,多变量的时序空间模型处理的时间复杂度非常大,达到$O(m^3)$,因此可以考虑采用如稀疏矩阵、矩阵的秩和矩阵分解技术来进一步提高发布效率.

文献[49]基于滤波技术提出了一种时序关联数据的发布方案 CTS-DP,保证在数据关联的情况下,发布的结果序列与添加关联噪声后的序列满足差分隐私要求,同时攻击者不能通过修正技术清洗掉添加的噪声,还原真实数据.文献[50]则将滤波技术应用到多输入输出(MIMO)事件数据流中,利用滤波降低发布结果噪声.文献[51-53]将滤波应用到监控实例(实时交通速度估计和建筑物占用情况预测)中,并提出了噪声添加的优化方案.

上述方案都是面向无限数据流实时发布方案,文献[54]借鉴 PID 反馈机制提出一种面向离线动态数据集的直方图发布的隐私保护方案 DSAT,该方案实现 user 级隐私保护.方案的基本思想是基于 PID 反馈机制进行动态调整抽样发布的频率,反馈公式为$E_i = |C_i/i - C/N|$;C_i是当前已发布次数,如果在该时间点之前发布频率过高,则调大阈值,降低采样频率;反之调小阈值,增加采样频率,以保证在时间序列内固定发布C次数据.与 FAST 方案相同的是, DSAT 也采用 PID 机制进行采样发布,都能实现 user 级的隐私保护;不同的是,两者的 PID 反馈公式设计不同;同时,FAST 是一种实时处理方案,而 DSAT 是离线数据处理方案.

4.2 分布式数据流阈值函数的持续监控保护

上述所有方案是基于简单聚集函数的持续监控保护方案.在数据流的实时监控任务中,往往存在一些复杂的统计分析任务(如阈值函数监控),也需要对其进行隐私保护.文献[55,56]以分布式数据流为应用场景,分别对数据流中元素信息增益值的阈值函数和统计值列表中r百分位值的阈值函数进行持续监控保护.问题可以抽象化为$f(\cdot) > \tau, f(\cdot)$表示基于计数的统计任务,τ表示阈值.在文献[55]中$f(\cdot)$表示某元素的信息增益值的持续计算,在文献[56]中则表示对基于所有分布式站点生成的统计值列表中不小于r%个列表元素的最小值的持续计算.阈值函数监控就是持续监控上述统计值是否大于提前设定的阈值τ.

文献[55]中信息增益阈值函数的持续监控以垃圾邮件关键词监控场景为例,描述如下:假设存在k个分布式站点,1个中心节点,持续监控关键词t在各分布式数据流滑动窗口(宽度为W)内的邮件中出现的信息增益值$IG(t,W) = \sum_{\alpha\in\{1,2\},\beta\in\{1,2\}} c_{\alpha,\beta} \cdot log c_{\alpha,\beta}/((c_{\alpha,1} + c_{\alpha,2}) \cdot (c_{1,\beta} + c_{2,\beta}))$,$c_{1,1}, c_{1,2}$分别表示W滑动窗口内包含t的垃圾邮件和非垃圾邮件中所占的比例;$c_{2,1}, c_{2,2}$则分别代表不包含t的垃圾邮件和非垃圾邮件所占的比例.根据t的增益值是否大于某个阈值τ,判断t是否可以被用做垃圾邮件监控.文献[56]r百分位值的阈值函数监控问题描述为:k个分布式节点,1个中心节点,各分布式节点分别执行对应的统计任务,发送统计信息给中心节点,中心节点基于k个聚集统计值列表,找出列表中大于等于 r%个列表值的最小列表元素值X_r,判断X_r是否大于阈值τ.

由于统计过程中涉及到用户隐私,需设计隐私保护方案.两个方案的核心策略都是采用Safe Zone(安全区域)局部约束技术[57]来最大化时间序列上的隐私预算的利用率,并且降低各分布式节点与中心节点的通信代价,延长差分隐私下持续监控生命期.因此,用Safe Zone代表两种方案,其基本思想:将中心节点的整体监控条件转换成一组各分布式站点的局部约束条件(被称为安全区域).如果每个分布式站点的局部统计值在安全区域内,则表示近期的统计值变化不大,不需要中心节点进行通信,节约了一定的隐私预算;否则,花费一定的隐私预算将各分布式站点的局部统计值发送到中间节点,中间结点计算并进行判断.虽然两个方案都采用Safe Zone技术实现,但文献[55]采用球状安全区域,在每个时间间隔做隐私保护处理时,对球状区域半径初始化,各分步式站点与中心节点通信的统计信息和中心节点的信息增益值判断三个过程都添加了拉普拉斯噪声对其隐私进行保护.文献[56]则采用设置一组安全区间$\{[l_{\mathrm{i}}(\mathrm{t_i}), r_{\mathrm{i}}(\mathrm{t_i})]\}(1 \le \mathrm{i} \le \theta)$, l_{i}和r_{i}对应i区间的左右临界值, $\mathrm{t_i}$表示时间, θ为安全区间的数目.做隐私保护处理时,对安全区间的左右临界值分别加拉普拉斯噪声,然后采用指数机制确定各分布式站点统计值所属的安全区间,最后中心节点根据各分布式站点发送的安全区间索引号进行阈值判断.

上述两种方案采用Safe Zone技术与拉普拉斯机制或指数机制相结合,不仅降低了通信量,也延长了持续监控生命期.持续监控保护方案满足 user 级差分隐私保护.两个方案缺点是只能进行有限长的持续监控,如何实现无限长分布式数据流的持续监控保护依然是具有挑战性的研究问题.

4.3 集值对的增量更新发布保护方案

针对集值对的持续发布问题,文献[58]提出一种增量更新发布隐私保护方案 IncBuildTBP,可满足的 user 级隐私保护.方案采用基本隐私预算分配方案:设定数据库更新发布次数上限为U,每次发布分配隐私预算$\epsilon' = \epsilon/(U+1)$.由于集值对发布问题敏感度很高,文献借助一棵基于划分的分类树 TBP-Tree 来对庞大的集值对输出空间进行压缩,以降低集值对发布问题的高敏感度;同时,由于每次更新可能会产生的大量值为0的空节点,为空节点分配隐私预算会影响发布数据的可用性,所以每次更新都对这些节点进行剪枝,以提高发布数据的可用性.文献虽然实现了集值数据的动态的差分隐私发布,但是发布结果的可用性比较低,并且只能实现离线数据的处理.

4.4 轨迹发布的差分隐私保护方案

轨迹是具有时序关系的位置序列,攻击者通过对用户位置的持续监控(轨迹的监控),可以获取隐私信息.现有轨迹数据发布的差分隐私保护方案都是针对离线轨迹数据集,经差分隐私保护处理后,发布"净化版"的轨迹数据,隐私保护级别为 user 级.攻击者通过监控发布的轨迹信息,无法获得个体隐私.在该问题中,假设所有位置总数为m,轨迹最大长度为l,则可能生成的轨迹数目最多为$\sum_{i=1}^{l} m^i$.因此,轨迹发布隐私保护问题的的输出空间非常大,发布任务的敏感度和复杂度很高.为了缩小输出空间,降低敏感度,从而降低所需添加的噪声量,一些文献提出了相应的解决方案.

文献[59]提出一种轨迹发布的隐私保护方案Prefix,方案采用前缀树对输出空间进行压缩并添加噪声,发布净化版的轨迹数据集.其基本思想是假设很多轨迹数据具有相同前缀,将所有具有相同前缀的轨迹分到前缀树的同一个分支上,从而避免隐私预算的重复分配.根据前缀树的高度h,隐私预算ϵ被平均分配到各层,每层所分配的隐私预算$\bar{\epsilon} = \epsilon/\mathrm{h}$.从根节点到各层每个节点都对应一条轨迹,每个节点的计数加上满足$\bar{\epsilon}$的拉普拉斯噪声.为了提高发布效率,方案采用阈值剪枝策略,尽早删除不能继续扩展的分支节点.由于加噪后的节点计数会违背一致性约束,方案利用前缀树父子节点之间的约束关系对计数值进行了一致性处理,进一步提高了发布结果的可用性.如果位置空间域较小,前缀树高度较低,Perfix 方案发布结果的可用性较好.如果空间域较大,前缀树高度增加,被划分到同一分支的轨迹将大幅度减少,因此造成发布结果的可用性变低.

为解决 Perfix 方案的问题,文献[60]提出改进的轨迹发布方案 N-Grams,核心思想是采用变长n-gram 模型和马尔科夫模型来发布轨迹数据库.首先基于轨迹数据库,采用n-gram($\mathrm{n} \le \mathrm{n}_{max}$)模型统计空间域中位置之间的转移概率,并记录在自定义的数据结构-探索树中(前缀树结构,高度为n_{max}).隐私预算分配方案如下：和Prefix分配方案一样,探索树每一层节点初始化分配的隐私预算为$\bar{\epsilon} = \epsilon/\mathrm{n}_{max}$.但由于探索树很多分支长度不够n_{max},为

提高隐私预算利用率,从第二层的节点开始,方案对当前节点所在分支的底层长度h_v预测,根据预测值将剩余隐私预算平均分配到该分支剩下的几层节点中,每层节点分配到的隐私预算会有增加,从而能提高了发布结果的可用性.在发布轨迹时,长度不大于n_{max}的轨迹通过遍历探索树进行发布,长度超出n_{max}的轨迹则采用马尔科夫模型预测其计数.由于探索树的高度远远小于Prefix方案中前缀树的高度,该方案解决了前缀树过高带来的低可用性问题.但由于探索树只记录最大长度受限的轨迹的n-gram信息,造成了一定的信息丢失,也会影响发布结果的可用性.

Prefix 和 N-Grams 方案都是基于轨迹中有相同前缀的前提下,采用 n-gram 或前缀树对输出空间进行压缩,来提高发布数据的可用性.然而由于轨迹的异质性,很少轨迹有相同的前缀,Cluster[61,62]在取消上述假设的前提下,设计一种基于聚类的轨迹发布保护方案.该方案核心思想是首先采用指数机制对同一时间的位置进行差分隐私下聚类,每个时间对应的位置聚类为k类;然后基于聚类后的位置,发布轨迹数据集.经过聚类,轨迹输出空间大大降低,从而也降低了发布的敏感度,提高了可用性.

文献[63-65]则提出利用二维空间的参照系统来缩小轨迹输出空间,降低发布敏感度.Homogeneous[63]采用同质参照系统来对用户轨迹进行采样,降低位置空间中位置之间转移的规模.参照系统中,如果设置的粒度越大,轨迹输出空间就越小,反之越大.如将二维空间划分为网格,基于网格参照系统抽样轨迹,网格越大,轨迹就越短,轨迹的输出空间就越小,反之越大.由于同质的参照系统不能很好地反映用户不同速度的轨迹模式,也会造成一定的信息丢失,对此 Hierarchical[64]提出建立层次参照系统$HRS = \{\Sigma_{v_1}, \ldots \Sigma_{v_i}, \ldots, \Sigma_{v_m}\}(v_i/v_{i-1} = 2)$,来捕捉用户不同速度的轨迹,速度较快的轨迹模式采用粗粒度的参照系统进行抽样,速度较慢的采用细粒度的参照系统.基于HRS,对位置空间域进行离散化,对每种参照系统Σ_{v_i},维护一个轨迹前缀树模型.为使这组前缀树更好的体现轨迹特点,方案根据轨迹数据库设计了满足隐私保护的模型选择机制,其任务是从这组前缀树模型中选择合适的前缀树表示不同速度的轨迹,同时对选择的前缀树模型设计更合适的树高和节点间的转移概率,添加拉普拉斯噪声以满足隐私保护要求.最后方案采用基于方向的权重抽样技术修复添加噪声后的轨迹的方向性信息,进一步提高了发布结果的可用性.Hierarchica 和文献[63]方案中的参照系统都是基于空间区域的网格划分,然后基于网格的关键点(achor)对轨迹进行抽样.两种方案的参照系统都没有做隐私保护,对此 PTCP[65]提出基于聚类的参照系统设计,对聚类后每个类的中心(achors)添加拉普拉斯噪声保护.然后基于 achors 对轨迹进行抽样,基于前缀树结构实现轨迹数据发布的差分隐私保护.

4.5 User级隐私保护方案小结

本小节对所有 user 级隐私保护方案从其优缺点,监控任务类型和面向的发布任务等几个方面进行总结和对比分析(见表 3),然后指出现有方案的不足及未来研究趋势.

(1) 在简单聚集统计信息的持续发布方案中,为最大化时间序列上的隐私预算利用率,大部分方案采用抽样技术来提高发布结果的可用性,其原理是结合时序数据的特点,选出代表性的发布点进行发布,降低发布次数,增大每个发布点所分配的隐私预算.但这种方案只是延长持续监控生命期,持续监控数据流长度依然有限. 如何能实现对无限长动态数据的持续监控保护是需要解决的问题.

(2) 在分布式数据流持续监控保护中,现有持续监控保护方案为延长持续监控保护期,采用一系列将全局监控条件转换为局部监控条件的分解技术,但这些技术的时间和空间复杂度很高,极大降低了分布式数据流处理效率.可以对分解技术作进一步的改进处理:如基于凸分解或滤波技术提高分解效率.也可以基于衰减模型的数据流处理模型设计基于隐私衰减的隐私保护方案,实现对无限长数据流的处理.

(3) 在轨迹数据的发布问题中,由于发布任务的输出空间太大,对其进行隐私保护的敏感度和复杂度很高.所以现有轨迹数据的发布方案提出了各种降低输出空间,降低发布任务敏感度的策略,如基于前缀树、n-gram 模型和马尔科夫模型、聚类和参照系统等技术来缩小输出空间后,对其进行差分隐私处理.现有方案发布结果的可用性与实际应用仍有一定的差距,进一步提高发布结果可用性是值得继续研究的问题.同时所有方案都是离线轨迹数据的处理,对于实时轨迹数据的差分隐私发布是值得下一步研究的问题.

(4) 所有 user 级隐私保护方案也分为实时在线处理和离线处理两种方式.除 FAST 和 SafeZone 可以进行在线数据流的处理,其余所有方案都为离线数据处理.同时大多数持续监控任务还都是基于count的简单统计,但实际应用中往往存在一些复杂的数据分析任务.静态数据集上已有针对这些复杂分析任务的差分隐私保护研究工作[66-70],但持续监控下的相关工作还很少.因此,持续监控下差分隐私保护与复杂监控任务的结合还有待进一步的研究.

Table 3 Comparation of schemes on user-level privacy under continual monitoring

表 3 持续监控下 user 级隐私方案的对比分析

算法	主要优点	主要缺点	保护等级	发布任务
基本方案[22]	平均分配,隐私预算分配简单	噪声大,发布结果可用性低	user 级	简单聚集信息发布
DFT[44]	基于傅里叶级数变换选取抽样点进行发布	计算开销大,只能处理离线数据,只能处理定长时序数据	user 级	简单聚集信息发布
FAST[46]	基于 PID 和滤波技术采样发布,可以实时发布数据,效率高,算法误差小	需提前设定发布次数	user 级	简单聚集信息发布
CTS-DP[49]	利用滤波处理相关数据发布	只能处理离线数据	user 级	简单聚集信息发布
DSAT[50]	基于 PID 采样和相邻时刻直方图的相似性计算发布,提高了隐私预算利用	只能处理离线数据和定长更新发布	user 级	直方图发布
SafeZone[55,56]	基于SafeZone采样发布,降低通信量,延长了监控保护生命期	持续监控生命期有限	user 级	阈值函数的监控
IncBuildTBP[58]	时序上平均分配隐私预算,利用TBP分类树降低敏感度	只能处理离线数据,可用性较低	user 级	集值对发布
Prefix[59]	利用前缀树,降低敏感度压缩轨迹	只能处理离线数据,轨迹需有大量相同前缀,可用性低	user 级	轨迹发布
N-Grams[60]	利用 n-gram 和马尔科夫模型降低输出空间,降低敏感度	只能处理离线数据,轨迹子序列需有相同前缀,可用性较低	user 级	轨迹发布
Cluster[61,62]	利用聚类降低输出空间,降低敏感度	只能处理离线数据,聚类造成信息丢失	user 级	轨迹发布
Homogeneous[63]	用同质参照系统降低输出空间,降低敏感度	只能处理离线数据,参照系统造成信息丢失	user 级	轨迹发布
Hierarchical[64]	用层次参照系统降低输出空间,降低敏感度	只能处理离线数据,参照系统造成信息丢失	user 级	轨迹发布
PTCP[65]	增加了对参照系统的隐私保护	只能处理离线数据,参照系统造成信息丢失	user 级	轨迹发布

5 持续监控下w-event 隐私保护方案

从隐私保护强度上看, w-event 隐私是介于 user 级隐私和 event 隐私之间的一种隐私保护方案,本节对持续监控下w-event 隐私保护方案进行总结和分析.

5.1 w-event级隐私保护方案

针对 4.1 节中的简单聚集统计信息持续监控保护问题,文献[21]提出了两种满足w-event ϵ-差分隐私的保护方案:BA和BD.两种方案都能实现对无限长数据流的持续监控保护.根据 1.2.3 节中的w-event ϵ-差分隐私定义,为保证ϵ隐私,在时间序列上的任意w滑动窗口内,分配的隐私预算都必须小于等于ϵ.隐私预算的基本分配方案是将ϵ平均分配到每个窗口内的w个时间点上,每个点的隐私预算为ϵ/w.但该分配方案数据可用性较低,为提高滑动窗口内的隐私预算利用率,方案也采用了采样发布策略,即选出窗口内有代表性的时间点(数据变化较大)作为主动发布点,只有主动发布点上才分配隐私预算;其他时刻做被动发布,不分配隐私预算.两种隐私保护方案都包含两个核心过程:分别是相邻时刻发布值的相似性计算过程和差分隐私发布过程.隐私预算被平均分配到两个过程:$\epsilon = \epsilon_1 + \epsilon_2, \epsilon_1 = \epsilon/2$用于数据相似性计算;$\epsilon_2 = \epsilon - \epsilon_1$用作差分隐私发布.

BD(隐私预算指数机制分配)和BA(隐私预算吸收)都基于上述两个基本过程实现,不同的是窗口内每个采样点的隐私分配方案有区别.下面分别介绍两种方案的具体实现.

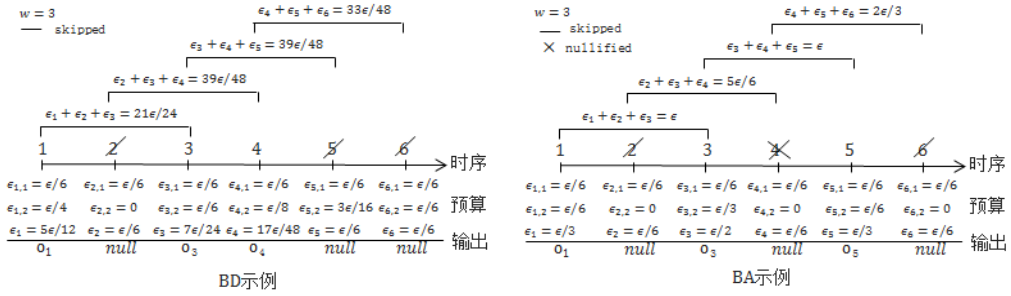

Fig.12 Example of budget allocation under w-event privacy [21]

图 12 w-event 隐私预算分配方案示例[21]

BD方案的特点是将ϵ_1平均分配到每个窗口内的w个时间点上,对于每个时间i,用$\epsilon_{i,1}=\epsilon/(2\cdot w)$做相邻时间数据相似性的计算.如果变化够大,则将$\epsilon_2$按指数递减方式分配给采样点,每个采样点i得到$\epsilon_{i,2}=\epsilon_{rm}/2$,计算方法是先找出当前活动窗口$[i-w+1,i]$内还可用的隐私预算$\epsilon_{rm}=\epsilon/2-\sum_{k=i-w+1}^{i-1}\epsilon_{i,2}$,然后将剩余隐私预算$\epsilon_{rm}$以指数方式分配给采样点i.如果变化不大,则时间i的隐私预算$\epsilon_{i,2}=0$.如图 12 所示的 BD示例中,早期采样点的隐私预算比晚期采样点的隐私预算大,所以当前活动窗口中越靠后的时间点的发布结果所需添加的噪声越大.BD方案的隐私预算会有浪费,因为指数分配方式永远有一部分预算不能利用.

而在BA方案中,用于相似性计算的隐私预算分配方案与BD方案相同,窗口内每个时间点的$\epsilon_{i,1}=\epsilon/(2\cdot w)$;但窗口内用于差分隐私发布的隐私预算分配方案与BD不同.BA将用于发布的隐私预算ϵ_2先初始化平均分配到窗口内的各时间点上$\epsilon_{i,2}=\epsilon/(2\cdot w)$.如果某个时间点i 数据变化不大,则该时间点的$\epsilon_{i,2}$被节约下来,重新设置$\epsilon_{i,2}=0$;节约下的隐私预算用于被后续采样点吸收,以提高后续发布时刻结果的可用性.如果i 数据变化大,则计算在该时刻可以吸收的隐私预算,然后加上i上已有的隐私预算进行发布.图 12 的BA方案示例中,时刻 2 是被动发布,所以该时刻的发布预算由$\epsilon/6$变为 0;时刻 3 采样发布点,吸收了时刻 2 的隐私预算,因此隐私预算由$\epsilon/6$变为了$\epsilon/3$;在BA方案中,有两种被动发布状态,一是 skipped 点,表示数据变化不大需被动发布;二是 nullified 点,表示因窗口内隐私预算用完而被动发布.如图 12BA 示例中,由于时刻 3 吸收了时刻 2 隐私预算,所以时刻 4 即使数据变化较大,但由于没有可以分配的隐私预算,也做了被动发布.BA方案的缺点是会出现预算使用完而某些时间点没有隐私预算只能做被动发布的情况.两个方案共同的缺点是时序上隐私预算分配不均衡,造成每个发布时刻发布结果添加的噪声量不一致.

文献[71]面向无限轨迹流的数据发布提出了一种满足w-event 的隐私保护方案 GA+MMD.该方案基本思想和BD基本相同,但对其做出了以下两个改进:一是考虑了每个用户对个体轨迹长度的隐私保护强度不同,所以方案结合每个用户提出的个性化轨迹长度 (u_i,l_i)的隐私保护要求,在窗口内的每个时刻用于相似性计算的隐私预算为ϵ_1/l_{max},l_{max}是所有用户中最长的轨迹隐私长度;每个时刻用于发布的隐私预算$\epsilon_{i,2}$还是采用指数递减的形式进行分配.二是与BD方案中被动发布时刻采用相邻时刻的发布结果代替,GA+MMD 中采用贪心算法找出活动窗口内与该时刻发布结果最相近的结果进行被动发布.通过上述两种策略,GA+MMD 进一步提高了发布结果的可用性.但方案中对用户的个性化隐私处理比较简单,如何结合现有的个性化差分隐私方案[72-74],实现真正的持续监控下的个性化的轨迹发布值得进一步研究.

5.2 w-event级隐私保护方案的对比分析

本节先从主要优缺点上对持续监控下w-event 隐私保护方案进行对比分析(表 4),然后指出现有方案不足及未来研究趋势.

目前,w-event 隐私保护方案相对较少.上述三种方案都可以实现对无限长数据流的持续监控保护,其保护强度介于 event 级别和 user 级别隐私之间.三种方案实现核心都是考虑如何在w滑动窗口内提高隐私预算的利用率,提高发布结果的可用性.但都存在以下缺点:时序上隐私预算不能充分利用.如何充分利用隐私预算,提高

持续发布结果可用性是值得进一步研究的问题.同时,这些方案都是面向简单聚集统计信息的持续监控保护,如何拓展相应方案到更复杂的持续监控任务中值得进一步研究.

Table 4 Comparation of schemes on w-event privacy under continual monitoring

表 4 持续监控下w-event 隐私保护方案对比分析

算法	主要优点	主要缺点	保护等级	发布问题
BA[21]	回收被动发布点的隐私预算用于采样点发布,能处理无限数据流	有发布点因隐私预算耗尽不能发布	w-event	简单聚集统计
BD[21]	采用指数机制分配隐私预算,能处理无限数据流	各抽样点隐私预算分布不均匀,隐私预算不能充分利用	w-event	简单聚集统计
GA+MMD[71]	考虑用户的个性化的要求,能处理无限数据流	各抽样点隐私预算分布不均匀,隐私预算不能充分利用	w-event	简单聚集统计

6 未来研究展望

本节探讨持续监控下差分隐私保护的未来研究展望,主要分为持续监控下面向分布式数据流的隐私保护研究、持续监控下差分隐私算法通用评价标准的研究、持续监控下关联数据发布的隐私保护研究、持续监控下面向复杂统计任务的隐私保护研究、持续监控下个性化差分隐私保护研究和持续监控下本地化差分隐私保护研究.

6.1 持续监控下面向分布式数据流的隐私保护研究

大数据环境下很多持续监控场景都是基于分布式数据流的实时处理,如 Web 网页的用户点击,网络入侵检测、heavy hitter 实时监控和智能基础设施的实时监控等.在这些应用中,隐私保护问题非常重要.目前虽然已有一些分布式数据流持续监控保护方案,如垃圾邮件关键词的持续监控,r百分位值的持续监控等,但方案还很少.持续监控下分布式数据流统计的隐私保护研究应考虑了以下问题:一是对局部节点的统计过程做隐私保护;二是对中心节点的计算过程做隐私保护;三是要尽量延长持续监控生命期并降低局部站点与中心节点的总通信量;为解决上述问题,现有方案大多先对局部节点和中心节点的计算过程采用噪声机制进行隐私保护,然后采用安全区域分解技术将中心站点的全局监控条件转换为各分布式站点的局部监控条件,只有违反局部监控条件的情况下才分配隐私预算和中心节点通信,从而降低总通信量和隐私预算分配次数.分布式数据流持续监控保护可从以下几个方面进行研究:一是现有方案中分解技术的时间复杂度和空间复杂度很高,可以进一步改进;二是现有方案都是基于对元素精确统计算法做隐私保护处理,由于数据流的处理对时间和空间有严格要求,现有很多数据流处理算法都基于近似技术的统计,如滑动窗口技术、随机采样技术和 sketch 技术等.可以考虑基于这些算法做隐私保护处理,使得持续监控隐私保护方案更符合实际监控场景要求.三是目前很多分布式数据流的持续监控任务如分布式数据流的 top-k 元素监控、网络流量的异常检测、分布式数据流的频繁模式监控等,目前都还没有相关研究.因此,持续监控下分布式数据流统计任务的隐私保护是未来的一个研究方向.

6.2 持续监控下差分隐私算法通用评价标准的研究

在对持续监控下差分隐私发布方案进行评价时,现有方案提出了不同的参数设置以及评估标准.如在面向简单聚集统计发布的持续监控保护方案中,虽然都采用发布结果的误差上界进行评估,却没有考虑相同的评估条件:如采用的实验数据集是否一样,设定的空间约束条件是否一样,算法中一些通用参数(如ϵ)的设置和调整是否一致,上述因素都会影响算法的评估性能.目前这些方案缺乏在统一标准下的相关评估.在面向复杂任务的持续监控保护方案中,为提高发布结果可用性,通常利用数据依赖关系降低噪声.但数据集不同,数据依赖关系也不一样,导致发布结果误差不同.因此,这类隐私保护方案都不包含发布结果误差的对比分析.在实际方案部署时,没有统一的方案对比分析,如何选择一个更好的方案面临重大挑战,开发通用的评价标准对于这类方案也尤为重要.在通用标准的制定中,不仅应考虑通用参数的调整与设置,数据集的特点对发布结果的影响,同时要考虑多样化多角度的评估标准,考虑各种发布问题的情况.所以,持续监控下通用差分隐私算法标准的研究是一个未来研究方向.

6.3 持续监控下关联数据发布的隐私保护研究

现有持续监控下隐私保护方案中,已有一些关联数据发布的隐私保护研究.如在位置发布的event级别保护方案中考虑了时序上用户位置之间关联,并对此进行建模,依据模型对关联关系所造成的隐私泄露量进行分析,降低位置持续发布任务的敏感度.在简单聚集信息发布的 user 级别保护方案中,基于时序数据关联,利用滤波技术提高了发布结果的可用性.在轨迹数据发布问题中,基于时序上用户位置之间关系的假设前提,降低了发布问题的敏感度.针对实际问题中数据关联关系建模,根据模型设计持续监控下隐私保护方案.该类方案具有以下优点:一是更符合真实的应用场景;二是能抵御攻击者具备这些关联信息的攻击,隐私保护更强;三是根据模型的发布往往可以降低发布问题的敏感度,可用性更高.目前持续监控下该类研究工作还较少,方案中对数据关联性的建模方法也比较单一.针对相关数据的隐私保护问题,已有研究工作提出了面向语义的差分隐私保护模型 Pufferfish 和 Blowfish.在持续监控场景下,针对特定监控问题中的关联数据,建立多样化的时序数据关联模型,基于面向语义的差分隐私保护模型,设计更灵活且具有语义的隐私保护方案是一个未来的研究方向.

6.4 持续监控下面向复杂分析任务的隐私保护研究

现有持续监控下差分隐私保护研究大都是基于count计数的简单统计和分析任务.面向复杂分析任务的持续监控保护研究不仅方案少,发布结果可用性也有待进一步提高.实际应用场景中往往存在更多的复杂监控任务,如频繁模式和频繁序列模式监控、持续分类和聚类监控等,这类持续监控任务暂时没有相关研究.持续监控下面向复杂分析任务的隐私保护研究具有以下两个难题:一是复杂分析任务本身输出结果空间就比较大,发布敏感度和复杂度比较高,如何降低发布任务的敏感度,提高发布结果的可用性是一个难题.二是在时序上,随着时间的增加,发布结果的噪声会累积增加,发布结果可用性逐渐变差.针对第一个难题,可以对未作隐私保护的原始算法进行分析,根据算法所采用的数据结构及处理过程,计算其敏感度.同时可改进相关任务在静态数据集上已有的降低敏感度技术[66-70],使其适用于动态发布环境.通过以上策略,提高发布结果的可用性.针对第二个难题,可以考虑利用时序上的数据依赖关系,自适应分配时序上的隐私预算,提高隐私预算的利用率.复杂分析任务的持续监控保护研究是未来的一个研究方向.

6.5 持续监控下个性化差分隐私保护研究

在持续监控场景中,通常参与者对自己的数据有不同的隐私要求,用统一要求来处理所有参与者的信息,可能对一些用户来说,隐私保障太低,而对另外一些用户又可能要求太高.因此,如何结合用户隐私需求,让用户定义个人的隐私级别是一个值得研究的问题[72-74].持续监控下个性化的隐私保护研究需要考虑以下问题:一是不同用户具有不同的隐私保护要求;二是对同一用户,不同的项目也会有不同的隐私保护要求;三是持续监控下,随着时间的推移,用户和项目的隐私级别也会有相应的变化.个性化差分隐私的隐私保护级别跟隐私预算大小相关,隐私预算越小,隐私保护强度越高;隐私预算越大,隐私保护强度越低.在该类研究中,如何针对多个隐私预算,采用随机响应技术或噪声机制等差分隐私实现技术设计满足所有隐私预算要求的实现方案是重点问题.

6.6 持续监控下本地化差分隐私保护研究

目前,持续监控下的差分隐私保护都是基于中心化差分隐私保护技术实现,该技术的前提是数据收集者必须是可信的第三方.随着智能基础设施的普及和GPS定位技术的应用,现有持续监控下的应用场景多为分布式的应用.在这种场景下,数据的收集往往类似于问卷调查,数据的管理者往往是不可信的第三方.同时,由于分布式场景中收集的数据量特别大并且计算代价非常高,中心化的差分隐私技术不适合这种场景.目前一些研究工作[75,76]提出了本地化差分隐私技术(LDP)方案,通过采用随机响应技术对个体数据进行正向和负向的扰动,最终通过聚合大量的扰动结果来抵消添加在其中的正负向噪声,从而得到有效的统计结果.如何将LDP技术应用到持续监控下的数据发布和分析任务中是未来值的研究的一个方向.

7 结束语

随着信息技术的发展及物联网技术的兴起,出现了越来越多的持续监控应用场景.在这些场景中,如何对参与者持续分享的数据进行隐私保护面临重大挑战.本文对持续监控下差分隐私保护领域已有的研究成果进行了总结,首先介绍了持续监控下隐私保护模型,包括不同保护级别的差分隐私模型定义、实现机制和持续监控下差分隐私保护方案的评估原则;然后重点对 event 级、user 级和w-event 级三种隐私保护级别的方案进行总结和分析.在对已有研究成果深入对比分析的基础上,指出了持续监控下差分隐私保护的未来研究方向.持续监控下差分隐私保护的研究成果还较少,仍有诸多关键问题需要进一步的研究.希望本文工作可以为持续监控下差分隐私保护的相关研究人员提供参考.

References:

[1] Sameera Ghayyur, Yan Chen, Roberto Yus, Ashwin Machanavajjhala, Michael Hay, Gerome Miklau, Sharad Mehrotra, IoT-Detective: Analyzing IoT Data under Differential Privacy. In: Proceedings of the 2018 International Conference on Management of Data Conference (SIGMOD), ACM, 2018: 1725-1728. [DOI:10.1145/3183713.3193571].

[2] Cynthia Dwork, Differential Privacy and the US Census, In: Proc. of the 38th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems (PODS), ACM, 2019:1. [DOI: 10.1145/3294052.3322188].

[3] Pierangela Samarati, Latanya Sweeney.Generalizing data to provide anonymity when disclosing information. In: Proc. of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems (PODS), ACM, 1998:188. [DOI: 10.1145/275487.275508].

[4] Machanavajjhala A, Kifer D, Gehrke J, Venkitasubramaniam M. l-diversity: Privacy beyond k-anonymity. ACM Transactions on Knowledge Discovery from Data (TKDD).2007.1(1).3. [DOI: 10.1145/1217299.1217302].

[5] Li N, Li T, Venkatasubramanian S. t-closeness: Privacy beyond k-anonymity and l-diversity. In: Proc. of IEEE 23rd International Conference on Data Engineering (ICDE), IEEE, 2007: 106-115. [DOI: 10.1109/ICDE.2007.367856].

[6] Dwork, C., Roth, A. The algorithmic foundations of differential privacy. Found. Trends Theor. Comput. Sci. 2014. 9(3–4).211–407. [DOI: 10.1561/0400000042].

[7] Tianqing Zhu, Gang Li, Wanlei Zhou, Philip S. Yu. Differential Privacy and Applications. Advances in Information Security 69.2017. 1-222. [DOI: 10.1007/978-3-319-62004-6].

[8] Yonghui Xiao, Li Xiong, Liyue Fan, Slawomir Goryczka, Haoran Li.DPCube: Differentially Private Histogram Release through Multidimensional Partitioning. Transactions on Data Privacy. 2014. 7(3): 195-222.

[9] Chao Li, Gerome Miklau, Michael Hay, Andrew McGregor, Vibhor Rastogi. The matrix mechanism: optimizing linear counting queries under differential privacy, VLDBJ, 2015, 24(6): 757-781. [DOI: 10.1007/s00778-015-0398-x].

[10] Ganzhao Yuan, Zhenjie Zhang, Marianne Winslett, Xiaokui Xiao, Yin Yang, Zhifeng Hao.Low-Rank Mechanism: Optimizing Batch Queries under Differential Privacy. In: Proc. of the VLDB Endowment (PVLDB).2012.5(11):1352-1363.

[11] José Camacho, Pedro Garcia-Teodoro, Gabriel Maciá-Fernández, Traffic Monitoring and Diagnosis with Multivariate Statistical Network Monitoring: A Case Study. IEEE Symposium on Security and Privacy Workshops, 2017: 241-246.[DOI: 10.1109/SPW.2017.11]

[12] Cynthia Dwork, George J. Pappas, Privacy in Information-Rich Intelligent Infrastructure, CoRR abs/1706.01985, 2017.

[13] Barbosa P, Brito A, Almeida H. A Technique to provide differential privacy for appliance usage in smart metering [J]. Information Sciences, 2016, 370–371:355-367. [DOI: 10.1016/j.ins.2016.08.011].

[14] Eibl G, Engel D. Differential privacy for real smart metering data. Computer Science - R&D. 2017.32(1-2): 173-182. [DOI: 10.1007/s00450-016-0310-y].

[15] Donghe Li, Qingyu Yang, Wei Yu, Dou An, Yang Zhang, Wei Zhao. Towards Differential Privacy-Based Online Double Auction for Smart Grid.[J]. IEEE Trans. Information Forensics and Security, 2020, 15: 971-986.

[16] Hui Cao, Shubo Liu, Longfei Wu, Zhitao Guan, Xiaojiang Du.Achieving Differential Privacy against Non-Intrusive Load Monitoring in Smart Grid: a Fog Computing approach. Concurrency and Computation: Practice and Experience 2019,31(22).[DOI: 10.1002/cpe.4528]

[17] Joseph A. Calandrino, Ann Kilzer, Arvind Narayanan, Edward W. Felten, Vitaly Shmatikov."You Might Also Like: " Privacy Risks of Collaborative Filtering. IEEE Symposium on Security and Privacy, IEEE, 2011: 231-246. [DOI: 10.1109/SP.2011.40].

[18] Dwork, C. Differential privacy. In: Proc. of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks (ICALP),ACM, 2006. 88-93. [DOI: 10.1007/11787006.1].

[19] Dwork, C. Differential privacy in new settings.In: Proc. of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms (SODA).ACM, 2010:174-183. [DOI:10.1137/1.9781611973075.16].

[20] Dwork, C., Naor, M., Pitassi, T., and Rothblum, G. N. Differential privacy under continual observation. In: Proc. of the forty-second ACM symposium on Theory of computing (STOC).ACM, 2010:88-93. [DOI: 10.1145/1806689.1806787].

[21] Georgios Kellaris, Stavros Papadopoulos, Xiaokui Xiao, and Dimitris Papadias. Differentially Private Event Sequences over Infinite Streams. Proceedings of the VLDB Endowment. 2014.7(12), 1155–1166. [DOI: 10.14778/2732977.2732989].

[22] Dwork, C., McSherry, F., Nissim, K., Smith, A. Calibrating noise to sensitivity in private data analysis. Theory of Cryptography Conference (TCC). Springer, 2006. 265–284. [DOI: 10.1007/11681878_14].

[23] F. McSherry and K. Talwar, Mechanism design via differential privacy.In: Proc. of the 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS).IEEE, 2007. 94-103. [DOI: 10.1109/FOCS.2007.66].

[24] Warner, S. L. Randomized response: A survey technique for eliminating evasive answer bias. Journal of the American Statistical Association. 1965: 63-69. [DOI: 10.2307/2283137].

[25] T.-H. Hubert Chan, Elaine Shi, Dawn Song.Private and Continual Release of Statistics, ACM Transactions on Information and System Security, 2011. 14(3): 26:1-26:24. [DOI:10.1145/2043621.2043626].

[26] Dwork, C., Naor, M., Pitassi, T., Rothblum, G. N., and Yekhanin, S. Pan-private streaming algorithms. In: Proc. of Innovations in Computer Science (ISC), 2010: 66-80.

[27] Darakhshan J. Mir, S. Muthukrishnan, Aleksandar Nikolov, Rebecca N. Wright. Pan-private algorithms via statistics on sketches. In: Proc. of the thirtieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems (PODS).ACM, 2011: 37-48. [DOI: 10.1145/1989284.1989290].

[28] Jean Bolot, Nadia Fawaz, S. Muthukrishnan, Aleksandar Nikolov, Nina Taft. Private Decayed Predicate Sums on Streams, In: Proc. of the 16th International Conference on Database Theory (ICDT).ACM, 2013:284-295.[DOI:10.1145/2448496.2448530].

[29] Cynthia Dwork, Moni Naor, Omer Reingold, Guy N. Rothblum. Pure Differential Privacy for Rectangle Queries via Private Partitions, In: Proc. of International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT). Springer, 2015:735-751. [DOI: 10.1007/978-3-662-48800-3_30].

[30] Yan Chen, Ashwin Machanavajjhala, Michael Hay, Gerome Miklau.PeGaSus_ Data-Adaptive Differentially Private Stream. In: Proc. of 2017 ACM Conference on Computer and Communications (CCS).ACM, 2017:1375-1388.[DOI:10.1145/3133956.3134102].

[31] Jianneng Cao, Qian Xiao, Gabriel Ghinita, Ninghui Li. Efficient and Accurate Strategies for Differentially-private Sliding Window Queries. In: Proc. of the 16th International Conference on Extending Database Technology (EDBT).ACM, 2013.191–202. [DOI: 10.1145/2452376.2452400].

[32] Rui Chen, Yilin Shen, Hongxia Jin.Private Analysis of Infinite Data Streams via Retroactive Grouping. In: Proc. of the 24st ACM International Conference on Information and Knowledge Management (CIKM).ACM, 2015: 1061-1070.[Doi: 10.1145/2806416.2806454].

[33] T.-H. Hubert Chan, Mingfei Li, Elaine Shi, Wenchang Xu.Differentially Private Continual Monitoring of Heavy Hitters from Distributed Streams. In: Proc. of International Symposium on Privacy Enhancing Technologies Symposium (PETS).ACM, 2012:140-159. [DOI: 10.1007/978-3-642-31680-7_8].

[34] Misra, J., Gries, D. Finding repeated elements. Sci.Comput. Program. 1982.2(2).143–152.[DOI:10.1016/0167-6423(82)90012-0].

[35] Bloom, B.H.: Space/time trade-offs in hash coding with allowable errors. Commun. ACM. 1970.13(7).422–426.[DOI: 10.1145/362686.362692].

[36] Yonghui Xiao and Li Xiong. 2015. Protecting Locations with Differential Privacy under Temporal Correlations. In: Proc. of ACM Conference on Computer and Communications Security (CCS). ACM, 2015:1298–1309. [DOI: 10.1145/2810103.2813640].

[37] Yonghui Xiao, Li Xiong, Si Zhang, Yang Cao.LocLok: Location Cloaking with Differential Privacy via Hidden Markov Model. Proceedings of the VLDB Endowment (PVLDB).2017.10(12): 1901-1904.[DOI: 10.14778/3137765.3137804].

[38] Yang Cao, Masatoshi Yoshikawa, Yonghui Xiao, Li Xiong.Quantifying Differential Privacy under Temporal Correlations. In: Proc. of 2017 IEEE 33rd International Conference on Data Engineering (ICDE).IEEE, 2017.821-832. [DOI: 10.1109/ICDE.2017.132].

[39] D. Kifer and A. Machanavajjhala. Pufferfish: A framework for mathematical privacy definitions. ACM Trans. Database Syst., 2014.39(1):3:1–3:36. [DOI: 10.1145/2514689].

[40] Shuang Song, Yizhen Wang, Kamalika Chaudhuri.Pufferfish Privacy Mechanisms for Correlated Data. In: Proc. of the 2017 ACM International Conference on Management of Data (SIGMOD).ACM, 2017:1291-1306. [DOI: 10.1145/3035918.3064025].

[41] X. He, A. Machanavajjhala, and B. Ding. Blowfish privacy: Tuning privacy-utility tradeoffs using policies. In: Proc. of the 2014 SIGMOD International Conference on Management of Data (SIGMOD), ACM, 2014:1447–1458. [DOI: 10.1145/2588555.2588581].

[42] T. Zhu, P. Xiong, G. Li, and W. Zhou. Correlated differential privacy: Hiding information in non-iid data set. IEEE Transactions on Information Forensics and Security.2015:10(2):229–242. [DOI: 10.1109/TIFS.2014.2368363].

[43] R. Chen, B. C. Fung, S. Y. Philip, and B. C. Desai. Correlated network data publication via differential privacy. The International Journal on Very Large Data Bases (VLDB J), 2014.23(4):653–676. [Doi: 10.1007/s00778-013-0344-8].

[44] Vibhor Rastogi, Suman Nath.Differentially Private Aggregation of Distributed Time-Series with Transformation and Encryption (DFT). In: Proc. of the 2010 ACM SIGMOD International Conference on Management of data (SIGMOD).ACM, 2010: 735-746. [DOI: 10.1145/1807167.1807247].

[45] M. King, Process Control: A Practical Approach. Chichester, U.K.Wiley, 2010. [DOI: 10.1002/9780470976562.ch6].

[46] Liyue Fan, Li Xiong.An Adaptive Approach to Real-Time Aggregate Monitoring With Differential Privacy.IEEE Trans. Knowl. Data Eng. 2014: 26(9): 2094-2106. [DOI: 10.1109/TKDE.2013.96].

[47] Liyue Fan, Li Xiong, Vaidy S. Sunderam. Differentially Private Multi-Dimensional Time Series Release for Traffic Monitoring, In: Proc. of IFIP Annual Conference on Data and Applications Security and Privacy (DBSec).Springer, 2013:33-48. [DOI: 10.1007/978-3-642-39256-6_3].

[48] Liyue Fan, Luca Bonomi, Li Xiong, Vaidy S. Sunderam.Monitoring Web Browsing Behavior with Differential Privacy, In: Proc. of the 23rd international conference on World Wide Web (WWW).ACM, 2014: 177-188. [DOI:10.1145/2566486.2568038].

[49] Hao Wang, Zhengquan Xu.CTS-DP: Publishing correlated time-series data via differential privacy. Knowl.-Based Syst. 2017.122: 167-179.[DOI: 10.1016/j.knosys.2017.02.004].

[50] Jerome Le Ny, Meisam Mohammady, Differentially Private MIMO Filtering for Event Streams. IEEE Trans. Automat. Contr. 2018, 63(1):145-157.[DOI: 10.1109/TAC.2017.2713643].

[51] Jerome Le Ny, Ahmed Touati, George J. Pappas, Real-time privacy-preserving model-based estimation of traffic flows. 2014 International Conference on Cyber-Physical Systems (ICCPS), ACM/IEEE, 2014:92-10. [DOI: 10.1109/ICCPS.2014.6843714].

[52] Hubert Andre, Jerome Le Ny, A differentially private ensemble Kalman Filter for road traffic estimation,In: International Conference on Acoustics, Speech and Signal Processing (ICASSP),IEEE,2017: 6409-6413.[DOI: 10.1109/ICASSP.2017.7953390].

[53] Jun Wang, Rongbo Zhu, Shubo Liu, A Differentially Private Unscented Kalman Filter for Streaming Data in IoT. IEEE Access 6: 2018:6487-6495.[DOI: 10.1109/ACCESS.2018.2797159].

[54] Haoran Li, Li Xiong, Xiaoqian Jiang, Jinfei Liu.Differentially Private Histogram Publication For Dynamic Datasets: An Adaptive Sampling Approach, In: Proc. of the 24th ACM International on Conference on Information and Knowledge Management (CIKM).ACM, 2015: 1001-1010. [DOI: 10.1145/2806416.2806441].

[55] Arik Friedman, Izchak Sharfman, Daniel Keren, Assaf Schuster.Privacy-Preserving Distributed Stream Monitoring. In: Proc. of the 2014 Network and Distributed System Security (NDSS) Symposium.2014:1-14. [DOI: 10.14722/ndss.2014.23128].

[56] Jingchao Sun, Rui Zhang, Jinxue Zhang, Yanchao Zhang.PriStream: Privacy-preserving distributed stream monitoring of thresholded PERCENTILE statistics.In: Proc. of the 35th Annual IEEE International Conference on Computer Communications (INFOCOM).IEEE, 2016:1-9.[DOI: 10.1109/INFOCOM.2016.7524461].

[57] D. Keren, I. Sharfman, A. Schuster, and A. Livne, Shape sensitive geometric monitoring, IEEE Transactions on Knowledge and Data Engineering, 2012.24(8):1520–1535. [DOI:10.1109/TKDE.2011.102].

[58] Xiaojian Zhang, Xiaofeng Meng, Rui Chen.Differentially Private Set-Valued Data Release against Incremental Updates, In: Proc. of International Conference on Database Systems for Advanced Applications(DASFAA).Springer,2013:392-406.[DOI: 10.1007/2F978-3-642-37487-6_30].

[59] Rui Chen, Benjamin C. M. Fung, Bipin C. Desai, Nériah M. Sossou.Differentially Private Transit Data Publication: A Case Study on the Montreal Transportation System, In: Proc. of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD).ACM, 2012:213-221.[DOI:10.1145/2339530.2339564].

[60] Rui Chen, Gergely Ács, Claude Castelluccia.Differentially Private Sequential Data Publication via Variable-Length N-Grams, In: Proc. of ACM Conference on Computer and Communications Security (CCS).ACM, 2012: 638-649. [DOI: 10.1145/2382196.2382263].

[61] Jingyu Hua, Yue Gao, Sheng Zhong.Differentially private publication of general time-serial trajectory data. In: Proc. of 2015 IEEE Conference on Computer Communications (INFOCOM). IEEE, 2015: 549-557. [DOI: 10.1109/INFOCOM.2015.7218422].

[62] Meng Li, Liehuang Zhu, Zijian Zhang, Rixin Xu.Achieving differential privacy of trajectory data publishing in participatory sensing. Inf. Sci.2017.400: 1-13.[DOI:10.1016/j.ins.2017.03.015].

[63] Nikos Pelekis, Aris Gkoulalas-Divanis, et al., Privacy-aware querying over sensitive trajectory data, In: Proc. of 20th ACM Conference on Information and Knowledge Management (CIKM).ACM, 2011:895-904. [DOI: 10.1145/2063576.2063706].

[64] Xi He, Graham Cormode, Ashwin Machanavajjhala, et al., DPT: Differentially Private Trajectory Synthesis Using Hierarchical Reference Systems. In: Proc. of the VLDB Endowment (PVLDB).2015.8(11):1154-1165. [DOI:10.14778/2809974.2809978].

[65] Shuo Wang, Richard O. Sinnott.Protecting personal trajectories of social media users through differential privacy. Computers & Security.2017.67:142-163. [DOI:10.1016/j.cose.2017.02.002].

[66] Sen Su, Shengzhi Xu et al., Differentially Private Frequent Itemset Mining via Transaction Splitting, In: Proc. of 2016 IEEE 36rd International Conference on Data Engineering (ICDE).IEEE,2016.1564-1565.[DOI: 10.1109/ICDE.2016.7498427].

[67] Ning Wang, Xiaokui Xiao et al., PrivSuper: a Superset-First Approach to Frequent Itemset Mining under Differential Privacy, In: Proc. of IEEE 37rd International Conference on Data Engineering (ICDE).IEEE, 2017:809-820.[DOI: 10.1109/ICDE.2017.131].

[68] Dong Su, Jianneng Cao, Ninghui Li, Elisa Bertino, Min Lyu, Hongxia Jin.Differentially Private K-Means Clustering and a Hybrid Approach to Private Optimization. ACM Trans. Priv. Secur. 2017.20(4): 16:1-16:33. [DOI: 10.1145/3133201].

[69] Mehmet Emre Gursoy, Ali Inan, Mehmet Ercan Nergiz, Yücel Saygin.Differentially private nearest neighbor classification. Data Min. Knowl. Discov. 2017.31(5): 1544-1575. [DOI: 10.1007/s10618-017-0532-z].

[70] Dong Su, Jianneng Cao, Ninghui Li, Min Lyu. PrivPfC: differentially private data publication for classification. The International Journal on Very Large Data Bases (VLDBJ). 2018. 27(2): 201-223. [DOI: 10.1007/s00778-017-0492-3].

[71] Yang Cao, Masatoshi Yoshikawa. Differentially Private Real-Time Data Release over Infinite Trajectory Streams. In: Proc. of 16th IEEE International Conference on Mobile Data Management (MDM), IEEE, 2015, 68-73. [DOI: 10.1109/MDM.2015.15].

[72] H. Ebadi, D. Sands, and G. Schneider. Differential privacy: Now it's getting personal. In: Proc. of ACM-SIGACT Symposium on Principles of Programming Languages (POPL). ACM, 2015.69–81. [DOI: 10.1145/2775051.2677005].

[73] Z. Jorgensen, T. Yu, and G. Cormode. Conservative or liberal? Personalized differential privacy. In: Proc. of 2015 IEEE 35rd International Conference on Data Engineering (ICDE).IEEE, 2015.1023–1034. [DOI: 10.1109/ICDE.2015.7113353].

[74] M. Alaggan, S. Gambs, and A. Kermarrec. Heterogeneous differential privacy. J. Priv. Confidentiality, 2016, 7(2):1-28 [DOI:10.1145/2806416.2806546].

[75] R. Chen, H. Li, A. K. Qin, S. P. Kasiviswanathan, and H. Jin. Private spatial data aggregation in the local setting. In: Proc. of 2017 IEEE 36rd International Conference on Data Engineering (ICDE).IEEE, 2016:289–300.[DOI: 10.1109/ICDE.2016.7498248].

[76] Úlfar Erlingsson, Vasyl Pihur, Aleksandra Korolova. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. ACM Conference on Computer and Communications Security, ACM, 2014: 1054-1067. [DOI: 10.1145/2660267.2660348].