数据流场景下内存受限本地差分隐私协议鲁棒性评估与增强

田月池[1,2,3]，彭锦钰[1,2,3]，李效光[4]，张芊龙[1,2,3]，李凤华[1,2,3]，左金鑫[1,3]，牛犇[1,3]
（1.中国科学院信息工程研究所，北京 100085；2.中国科学院大学网络空间安全学院，北京 100049；
3.网络空间安全防御全国重点实验室，北京 100085；4.西安电子科技大学网络与信息安全学院，陕西 西安 710126）

摘要：本地差分隐私（LDP）因其本地扰动的特性易受数据投毒攻击，现有研究多聚焦于静态数据协议，对内存受限场景下采用域压缩技术的流式数据采集分析协议鲁棒性研究不足。针对BGR、DSR、BDR、CNR这4种主流LDP流式高频项识别协议展开研究。首先，提出一种攻击驱动的鲁棒性评估框架，设计基于最大化频率提升的数据投毒攻击，通过分析协议对该攻击的抵抗性来评估协议的鲁棒性。其次，将攻击成功率（ASR）作为评估指标，在3个现实数据集上评估发现DSR协议鲁棒性最高；同时，发现存储空间大小依然对协议鲁棒性有显著影响。最后，提出一种基于分布一致性的攻击检测方案，不依赖具体协议设计，通过数据频率重建技术解决受限内存下全域频率缺失的问题。实验表明，该方案在低隐私预算或高投毒比例下检测效果显著。

关键词：本地差分隐私；鲁棒性评估；数据流挖掘；内存受限；数据投毒
中图分类号：TN92
文献标志码：A
doi：10.11959/j.issn.1000–436x.TXXB260318

Evaluating and enhancing the robustness of memory-bounded LDP protocols for streaming data

Tian Yuechi[1,2,3], Peng Jinyu[1,2,3], Li Xiaoguang[4], Zhang Qianlong[1,2,3],
Li Fenghua[1,2,3], Zuo Jinxin[1,3], Niu Ben[1,3]
1. Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100085, China
2. School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China
3. State Key Laboratory of Cyberspace Security Defense, Beijing 100085, China
4. School of Cyber Engineering, Xidian University, Xi'an 710126, China

Abstract: Local differential privacy (LDP) is vulnerable to data poisoning attacks, yet the robustness of memory-bounded streaming data protocols with domain compression remains understudies. Four mainstream LDP streaming frequent item identification protocols (BGR, DSR, BDR, and CNR) were investigated. Firstly, an attack-driven evaluation framework was proposed, and a data poisoning attack maximizing frequency gain was designed. Secondly, using attack success rate (ASR) as the evaluation metric, experiments on three datasets showed that DSR was the most robust protocol, and storage size significantly impacts robustness. Finally, a protocol-agnostic attack detection scheme based on distribution consistency was proposed, utilizing data frequency reconstruction to address missing global frequencies. Experimental results demonstrate that the proposed scheme achieves significant detection performance under low privacy budgets or high poisoning ratios.

Key words: local differential privacy, robustness evaluation, data stream mining, memory bounded, data poisoning

收稿日期：2026–06–03；**修回日期：**2026–07–06
通信作者：左金鑫，zuojinxin@iie.ac.cn
基金项目：国家自然科学基金资助项目（No.62332018, No.62402363）；陕西省博士后科研项目资助（No.2024BSHSDZZ106）
Foundation Items: The National Natural Science Foundation of China (No.62332018, No.62402363), Shaanxi Province Postdoctoral Research Funding Program (No.2024BSHSDZZ106)

0　引言

本地差分隐私（local differential privacy, LDP）是无可信中心环境下一种标准的隐私数据采集与统计分析技术，已经在实际场景中得到了广泛的应用。例如，Google 公司利用 LDP 来统计 Chrome 浏览器中用户的网页浏览行为[1]，Apple 公司利用 LDP 来识别高频的 Emoji 表情及 Safari 浏览器中用户的媒体播放偏好[2]，微软公司利用 LDP 来分析 Windows 10 系统中应用程序的使用情况[3]。研究发现，LDP 协议本身存在固有的安全缺陷[4-8]。由于 LDP 中的扰动步骤在用户本地执行，因此恶意用户可以利用该特性来伪造扰动结果，执行数据投毒攻击，进而有效篡改最终统计结果。已有研究证实此类攻击在多种统计任务中普遍存在，并对这些协议抵御投毒攻击的鲁棒性进行了分析，包括频率估计协议[4-5]、分布估计协议[7]、键值对收集协议[8]、均值方差估计协议[6-7]及高频项识别协议[4-5]。

现有研究深入探讨了 LDP 协议面临的攻击和鲁棒性，但是主要聚焦于静态数据采集协议的攻击设计以及对应的鲁棒性分析，对于现实中更常遇到的动态流式数据采集分析协议缺乏研究。流式数据是一种随时间动态变化的数据形态，相比于静态数据，流式数据具有更加广泛的应用，例如用户消费流水、家庭住户用电情况等。在流式数据采集分析协议中，用户周期性地向服务器发送扰动后的敏感信息，以便服务器分析得到准确的用户特征。与静态数据分析不同，由于用户需要长期向服务器发送扰动结果，存储开销巨大，因此流式数据采集分析协议通常会采用数据压缩技术以确保在内存空间有限的情况下依然可以获得准确的统计结果。

本文针对内存受限场景下的 LDP 流式数据采集分析协议的鲁棒性展开研究，主要包括 BGR、DSR、BDR、CNR 这 4 种内存受限场景下具有代表性的流式数据采集分析协议[9]，目的是分析数据流中的高频项，而高频项分析是数据分析任务中的基础任务之一，通常作为很多复杂任务的前序工作。BGR 在用户端基于全数据域使用广义随机响应（generalized randomized response, GRR）进行数据扰动，以实现隐私保护的数据采集，并在服务器端利用 HG（HeavyGuardian）数据结构结合指数式衰减（exponential decay, ED）策略动态更新计数并驱逐低频项，从而在有限内存空间内实现对高频项的跟踪与存储。在此基础上，DSR、BDR 和 CNR 使用不同的数据预处理和扰动方式，使协议适用于多种场景。具体而言，DSR 通过将低频数据统一表示为一个固定值来压缩数据空间，尽管这样会引入偏差导致精度降低，但可以使数据分析过程的存储空间占用和计算开销最小化；BDR 通过分割隐私预算来分步执行热项判定与值扰动，避免了频繁切换数据域带来的计算开销，在计算开销和分析精度上达到了适中的量级；CNR 利用双 HG 结构回收被丢弃的冷项数据，进一步利用了低频信息，实现了最高的分析精度，但同时会带来额外的计算和存储开销。

本文提出了一种攻击驱动的鲁棒性评估框架，核心思路是设计最优的攻击策略来获得最大化的攻击效果，并在此基础上量化协议对攻击的鲁棒性。为此，首先设计了一种频率最大化攻击，攻击者的目标是通过发送投毒数据，使服务器错误地将攻击者预设的目标项识别为高频项。受最大化频率提升攻击[5]的启发，本文设计了一种周期性的投毒攻击策略，通过交替“集中发送目标项”与“分散发送目标项”这两个过程，在实践中实现了有效的攻击。其次，提出了一种新的鲁棒性评估指标，即攻击成功率（attack success rate, ASR）。ASR 通过评估攻击后目标项被错误识别为高频项的比例来度量协议的鲁棒性，ASR 越高表明攻击越成功，则协议的鲁棒性越低。

利用所提框架对现有协议进行分析，结果表明不同的协议在攻击下展现出不同的鲁棒性。具体而言，DSR 协议在多数情况下具有最高的鲁棒性，因为其采用了更严格的空间压缩，有效限制了攻击者的能力。之前的工作发现频率估计协议在频率最大化攻击下存在隐私性与安全性折中的性质，即隐私预算 ε 越小，攻击效果越好，协议的鲁棒性就越差。尽管流式数据采集协议依然以频率估计协议为基础模块，但是本文发现它们在所提攻击下并没有表现出上述隐私性与安全性折中的性质，原因在于，针对流式数据的攻击利用了压缩数据机制的漏洞，该过程对 ε 引入的本地差分隐私噪声不敏感，隐私预算的大小无法显著影响最终的攻击成功率。另外，之前的研究只发现了协议设计或协议参数与

鲁棒性之间的关系，而本文研究发现协议的鲁棒性和存储空间的大小之间存在着一种全新的关系。当HG的缓存空间较大（例如可缓存200条数据项）时，协议具有较低的鲁棒性。

为了增强LDP协议抵御攻击的能力，本文进一步提出了一种基于分布一致性的攻击检测方案。该检测方案不依赖具体的协议设计，相比于基线检测方案，能够显著提升在不同协议上的检测精度。由于攻击者构造的投毒数据不服从LDP的扰动概率分布，因此该检测方案的核心思路是通过对比用户上传的扰动结果和真实LDP的扰动概率分布是否一致来判断是否存在攻击。然而，有限的内存空间为检测方案的设计带来了新的挑战。由于HG仅存储Top-k高频项及其频率，因此服务器无法直接获得全数据域上所有数据项的频率信息，难以对真实的LDP扰动概率分布进行分析。为了解决这一问题，本文采用一种数据频率重建，技术来帮助攻击检测进行LDP扰动分析。具体而言，该方案首先统计HG中记录的Top-k高频项频率，在此基础上对全域频率直方图进行重建，以获得所有数据项的频率估计。然后，合成一批服从估计频率的数据，并将这些数据再次进行LDP扰动。由于服务器上不存在投毒数据污染，第二次的扰动结果可以被视为“无攻击”的参考基准。通过对比用户上传的扰动结果和服务器上模拟的扰动结果，就可以判断用户上传的数据中是否存在投毒数据。实验分析发现，所提检测方案在ε较小或者投毒数据比例较大时具有显著的检测效果。同时，由于CNR使用了双HG的数据结构，因此CNR下投毒数据的攻击信号相对其他协议更强，所以所提检测方案在CNR上检测效果更佳。

本文主要研究工作如下。

（1）建立了针对内存受限LDP协议的攻击驱动评估框架。针对流式场景下投毒数据极易被HG的指数式衰减策略驱逐的挑战，设计了适应性的攻击策略，旨在规避该策略的驱逐以提升投毒数据被识别为Top-k的成功率。

（2）通过实验与理论分析，揭示了数据流的LDP协议的鲁棒性规律。实验结果表明，针对流式数据的攻击对隐私预算ε引入的本地差分隐私噪声不敏感，未表现出传统的隐私性与安全性折中特性；同时，发现了协议鲁棒性与存储空间大小之间的全新关系，即内存的增加导致协议鲁棒性的降低。

（3）提出了一种基于分布一致性的攻击检测方案。针对内存受限导致域信息被截断、服务器无法直接获取全域分布情况的问题，采用数据频率重建技术，从Top-k高频项的统计结果中恢复全域频率直方图，从而在不需要额外信息的情况下有效识别投毒攻击。

1 相关工作

1.1 面向数据流的LDP协议

连续数据流的实时收集在工业监控、用户行为分析等领域具有广泛应用[10]，如何在保证数据效用的同时保护用户隐私是当前的核心问题之一。Bao等[11]针对定期更新的数据流收集，提出一种相关高斯机制，利用数据流的时间自相关性以及公共已知自相关模式，向基础高斯机制注入优化的时间相关噪声来减少整体噪声，从而提高数据的可用性。Joseph等[12]提出了在LDP下进行连续数据共享的框架，能够在时间变化中维护最新的统计数据，并且隐私预算根据子群体分布的变化而不是收集周期的数量而消耗，以减少整体隐私消耗。Ren等[13]提出了一个用户划分框架，避免了LDP对预算分割的高灵敏度，同时降低了通信开销。Wang等[14]提出一种用于发布实时数据流的差分隐私技术并将其扩展至本地差分隐私设置中，设计了基于最大期望（EM）算法的阈值优化器截断数据，设计了在线一致性算法并结合平滑器，以确保噪声估计的一致性并减少噪声。综上，针对连续数据流的LDP工作主要集中在提升统计精度和降低通信开销上。

为了解决海量数据带来的内存限制问题，近似数据结构被广泛引入隐私保护协议中，主要包括基于草图（sketch-based）和基于计数器（counter-based）两类。在基于草图的方案中，Sánchez-Macián等[15]研究了计数最小草图（count-min sketch, CMS）的差分隐私问题，提出了一种能抵御特定背景知识攻击的高频项提取算法；Wang等[16]提出了一种内存高效的草图结构，用于滑动窗口下的在线流采样与近似直方图构建。然而，基于草图的方案会对数据流中的所有项进行哈希记录，导致内存消耗较大，且直接检索高频项需要额

外的计算开销。基于计数器的机制通过维护固定数量的计数器直接保留高频项，主要包括 Misra-Gries、Lossy Counting 和 Space-Saving 这 3 种机制。然而，Misra-Gries 的“全局减一”策略会导致频率低估，且在长尾分布下极易将真实的次高频项误删[17]；Lossy Counting 的“周期清理”机制易在突发流量下导致不可控的内存峰值溢出，并截断处于增长初期的真实数据[18]。为了克服上述缺陷，Space-Saving[19]采用最小计数值淘汰策略，精准保留高频项。HG-LDP[9]等工作以此为底层数据结构，提升了内存利用率的同时，也因计数器资源争夺引入了安全隐患。因此，本文聚焦基于 Space-Saving 结构的流式 LDP 协议，以 HG-LDP 协议为典型代表，评估其在投毒攻击下的鲁棒性。

1.2　LDP 协议鲁棒性分析

恶意数据提供者通过上传虚假数据以影响服务器的数据分析结果，已成为部署 LDP 时面临的一大挑战。Cheu 等[4]系统地研究了 LDP 协议在面对恶意攻击时的脆弱性并建立了理论基础，证明当隐私级别较高或输入域较大时，任何非交互式 LDP 协议本质上都容易被小部分用户的操纵严重破坏。在此基础上，Cao 等[5]提出针对 LDP 协议频率估计和高频项识别的数据投毒攻击，将其形式化为单目标优化问题，以最大限度地提高目标项的频率收益。Li 等[6]提出了一种在用户输出域投毒的细粒度攻击方法，旨在通过较少恶意用户操纵均值和方差等统计结果为其预设值。Li 等[7]通过攻击驱动的方法评估针对数值属性的 LDP 协议鲁棒性，提出新的跨协议攻击收益度量指标，以评估 LDP 协议在不同数据集和不同攻击背景下的鲁棒性。针对键值数据 LDP 协议，Wu 等[8]提出了一种新的投毒攻击策略，将同时最大化目标键频率和均值的多目标问题形式化为双目标优化问题。

上述工作集中于静态数据场景下的 LDP 协议鲁棒性分析，对于数据流环境下的鲁棒性研究相对不足。尽管最近的一项研究[20]首次将对 LDP 鲁棒性的研究扩展到了流式 LDP 场景，以进行频率估计和均值估计等通用统计任务，但该研究假设系统资源充足，忽略了数据流对内存带来的挑战。为弥补这一局限性，本文关注内存受限场景下针对数据流的 LDP，设计相应的评估方法，重点探究存储空间等实际约束对其鲁棒性的影响。

2　预备知识

2.1　本地差分隐私

ε-事件级本地差分隐私[21-22]。随机化机制 $\mathcal{M}$ 满足 ε-事件级本地差分隐私（其中 $\varepsilon>0$），当且仅当对于任意一对相邻数据序列 V 和 V'，以及任意输出子集 $S\subseteq \mathrm{Range}(\mathcal{M})$，满足 $\Pr[\mathcal{M}(V)\in S]\leqslant \mathrm{e}^{\varepsilon}\Pr[\mathcal{M}(V')\in S]$。其中，参数 ε 为隐私预算，其越小则攻击者越难以区分 $\mathcal{M}(V)$ 和 $\mathcal{M}(V')$，即隐私程度越高。

广义随机响应（generalized randomized response, GRR）机制[23]是经典随机响应在多值离散域上的推广，同样满足本地差分隐私。d 表示域 Ω 的大小。每个具有私有值 $v\in\Omega$ 的用户以概率 p 报告真实值 $v'=v$，以概率 q 报告一个随机采样的值 $v'\in\Omega$，且 $v'\neq v$；其中，$p=\dfrac{\mathrm{e}^{\varepsilon}}{\mathrm{e}^{\varepsilon}+d-1}$，$q=\dfrac{1}{\mathrm{e}^{\varepsilon}+d-1}$，$d$ 是数据域的大小。

2.2　数据流高频项检测

高频项检测任务旨在从数据流中统计各项的频率，从而识别频率超过预设阈值或排名前列的少数核心项（Top-k），广泛应用于热点检测、趋势分析、推荐系统等场景[24-27]。在图 1 所示的高频项统计场景中，用户将本地用户端实时产生的项计数发送到服务器，服务器根据所有用户的本地流查找项目域中的 Top-3 高频项。简单地统计和存储数据流中出现的所有项和频率会给系统带来巨大的存储负担。用户提交的数据流往往包含敏感的个人信息，如果不加保护地向服务器提供这些数据流，用户隐私可能面临严重威胁。尤其在 Top-k 检测问题中，低频项虽然不是统计目标，但占据数据域的大部分，且可能暴露特定用户群体的敏感信息。

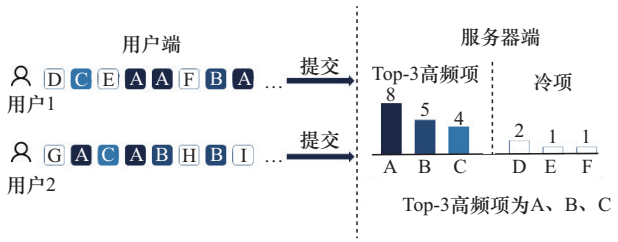


图 1　Top3 高频项统计示例

HG-LDP[9]是一种在有限内存下从数据流中检测 Top-k 高频项的本地差分隐私解决方案。考虑到现实世界数据流通常表现出高度倾斜的长尾特性，即数据域主要由极少数高频项主导，而其余绝大多数

项则相对稀疏或从未出现，该框架假设数据流服从Zipf分布（其参数形式为$f(i) \propto \frac{1}{i^{\alpha}}$，其中$\alpha$为偏度参数），从而实现对存储空间的高效利用。对于每个数据项，它在用户端对原始数据进行随机化以满足本地差分隐私，并在服务器端遵循指数式衰减策略将随机化后的项目插入HG中。收到查询后，它会对存储的信息进行去偏处理以发布频率。本节将描述其核心存储机制以及基于该机制的4种具体LDP协议。

核心存储组件HG是一种节省空间的键值对数据结构[28]，用于跟踪高频项。为了适应内存限制，HG的大小K远小于数据域大小d。HG利用指数式衰减策略来更新计数并处理内存溢出。当服务器收到一个项目e时，分为如下3种情况：若e已存在于HG中，则其计数加1；若e不在HG中且存储未满，则插入e并将其计数设为1；若e不在HG中且存储已满，则执行衰减操作，HG中频率最低的项目的计数以概率$P = b^{-c}$减少1，其中，$b > 1$为常数，c为当前计数值，若衰减后计数降为0，该项被驱逐并由新项目e替换，新项目的计数初始化为1。基于此机制的4种具体协议如下。

（1）BGR协议。其直接将广义随机响应与HG集成。用户在全域上使用广义随机响应对数据进行随机化，生成随机值y并发送给服务器。具体而言，真实项以概率$p = \frac{\mathrm{e}^{\varepsilon}}{\mathrm{e}^{\varepsilon} + d - 1}$保持不变，以概率$q = \frac{1}{\mathrm{e}^{\varepsilon} + d - 1}$变为域内的其他项。服务器根据指数式衰减策略将$y$整合进HG。由于存储的计数值包含噪声，服务器在发布统计结果前需按标准广义随机响应去偏方法处理。对于HG中的每个项v，其去偏后的计数为$\tilde{f}_v = \frac{C_v - Nq}{p - q}$，其中，$C_v$为HG中记录的噪声计数，$N$为服务器接收到的总数据量。

（2）DSR协议。其基于指数式衰减策略，在大多数情况下不需要冷项具体值的观察。若HG中最弱的键值对并未即将被驱逐，即最低计数值$C_{\min} > 1$，用户在由热项和代表所有冷项的特殊符号⊥组成的缩小数据域HG.$C \cup \{\perp\}$上进行随机化。此时目标域大小为$K + 1$，若原始项属于HG，则在该缩小数据域上应用广义随机响应，保持不变的概率为$p_2 = \frac{\mathrm{e}^{\varepsilon}}{\mathrm{e}^{\varepsilon} + K}$，扰动概率为$q_2 = \frac{1}{\mathrm{e}^{\varepsilon} + K}$；若原始项不属于HG，则将其视为⊥并应用相同的广义随机响应。仅当最弱的键值对即将被驱逐，即$C_{\min} \leqslant 1$时，才切换回全域随机化以识别具体冷项，此时概率参数与BGR协议完全相同。在服务器端，若收到具体冷项则按指数式衰减策略插入；若收到⊥，则直接对最弱的键值对执行衰减操作。服务器在每次切换随机化机制时，必须先使用当前机制的去偏公式对HG中所有的计数值进行去偏，随后将这些计数值乘以新机制去偏公式的分母，以防止在后续累加和最终发布时发生重复去偏。

（3）BDR协议。其将隐私预算ε分为两部分：ε_1用于判断项是否为热项，ε_2用于值的扰动。首先，使用ε_1决定是否如实报告该项属于HG，判断正确的概率为$p_1 = \frac{\mathrm{e}^{\varepsilon_1}}{\mathrm{e}^{\varepsilon_1} + 1}$，判断错误的概率为$q_1 = \frac{1}{\mathrm{e}^{\varepsilon_1} + 1}$。然后，根据判断的目标域对项进行随机化。若判断为热项，则在当前HG包含的项构成的域上进行随机化，该域大小为K，若原始项确实属于该域，则应用广义随机响应扰动，保持不变的概率为$p_2 = \frac{\mathrm{e}^{\varepsilon_2}}{\mathrm{e}^{\varepsilon_2} + K - 1}$，扰动概率为$q_2 = \frac{1}{\mathrm{e}^{\varepsilon_2} + K - 1}$；若原始项实际为冷项，则均匀映射到该域的任意一项。若判断为冷项，用户需进一步检查最低计数值$C_{\min}$，当$C_{\min} \leqslant 1$时，使用ε_2在冷项域上进行随机化并发送具体项，该冷项域大小为$d - K$，此时若原始项为冷项，则保持不变的概率为$p_3 = \frac{\mathrm{e}^{\varepsilon_2}}{\mathrm{e}^{\varepsilon_2} + d - K - 1}$，扰动概率为$q_3 = \frac{1}{\mathrm{e}^{\varepsilon_2} + d - K - 1}$，若原始项为热项则均匀映射到冷项域的任意一项；当$C_{\min} > 1$时直接发送符号⊥。服务器在发布结果时，使用统一公式对所有计数进行去偏：$\tilde{f}_v = \frac{C_v - \gamma_h N(p_1 q_2 - \frac{q_1}{K}) - \frac{N q_1}{K}}{p_1(p_2 - q_2)}$，其中$\gamma_h$为数据流中热项的真实比例。

（4）CNR协议。其扩展了BDR协议，通过引入另一个HG即LightPart以有效利用冷项报告。在用户端，随机化过程与BDR协议类似，但当判断结果为冷项时，无论$C_{\min}$为何值，用户总是使用ε_2在冷项域上按照BDR协议的冷项域概率参数进行随机化并发送具体项，不再发送⊥。具体地，在服务器端，插入现有HG即HeavyPart失败的项不予

丢弃，而是将其按指数式衰减策略独立插入新的副HG即LightPart中。当HeavyPart中的键值对即将被驱逐时，在LightPart中选取频率最高的候选项插入HeavyPart，并将其计数初始化为1，同时将其从LightPart中移除，以提供更准确的计数。去偏过程与BDR协议完全一致。

3　基于数据投毒的鲁棒性评估框架

3.1　框架概述

本节提出一种攻击驱动的鲁棒性评估框架，通过主动对流式LDP协议实施数据投毒攻击，以攻击效果作为鲁棒性的量化依据。该框架的核心思路如下：在相同条件下，攻击者对某协议的操纵效果越显著，说明该协议的鲁棒性越弱。通过量化攻击在特定条件下的有效性，可将不同协议间鲁棒性的比较转化为攻击有效性的比较，从而实现对各协议抗投毒能力的客观评估。

该框架包含3个步骤。第一步是构建针对流式LDP协议的攻击模型。所研究的协议在底层数据结构与更新策略上存在差异，若采用单一固定的攻击策略，可能对部分协议有效而对另一部分协议无效，进而导致评估结果出现系统性偏差。为此，本文针对各协议共享的底层数据结构HG的脆弱性以及事件驱动策略引入的动态特性，对攻击策略进行定制化设计，以确保攻击对所有待测协议均具备有效性，保证评估的公平性与覆盖性。这种定制化设计使评估结果能够直接反映各协议在面对特定系统级漏洞时的防御能力。第二步是定义鲁棒性评估指标，以攻击成功率量化算法鲁棒性，攻击成功率越高表明协议鲁棒性越弱。第三步是在实验评估的基础上，对各协议的鲁棒性进行理论分析，从理论层面解释不同协议在攻击下表现差异的成因。后续各节将依次介绍威胁模型、鲁棒性评估指标以及理论分析。

3.2　威胁模型

（1）攻击者能力与背景知识

与现有工作保持一致，假设攻击者控制$n_f = \beta n$个恶意用户，其中n为用户总数，$\beta \in (0,1)$表示投毒比例，剩余的$n_g = (1-\beta)n$个用户为良性用户。良性用户上报扰动后的噪声值，而恶意用户向服务器注入一组任意构造的消息$\mathbb{Y} = y_1, y_2, \cdots, y_{n_f}$。此外，由于LDP算法的扰动在用户端本地执行，攻击者掌握系统参数的全部知识，包括隐私预算ε和HG配置，但无法观测服务器的内部状态。

（2）攻击者目标

攻击者旨在提升特定目标项集合的排名，并操纵LDP协议将这些目标识别为前k个高频项。形式上，记$V = v_1, v_2, \cdots, v_m$为$m$个目标项的集合，目标项集合$V$的设定动机源自外部利益驱动。这些项由攻击者根据自身利益主动设定，如电商平台中卖家希望推广其商品或社交平台中营销方希望抬升特定话题热度，完全不依赖于对当前数据流中真实高频项分布的先验观测。事实上，攻击者不需要关心目标项原本是否属于真实的前k个高频项：如果目标项原本不在列表中，攻击旨在将其强行提升为高频项；如果目标项已经在列表中，攻击则会进一步巩固其排名。因此，攻击者只需直接向系统注入目标项即可。具体而言，攻击者利用受控的恶意用户注入构造的消息，增加V中项的计数，试图使其被服务器识别为前k个高频项。

（3）攻击策略

一种朴素的投毒策略涉及恶意用户持续上报同一个目标以激进地提升其频率，然而，这会产生高度重复的模式，显著偏离良性用户的行为，从而易于被检测和抑制。相反，本文提出一种结合分散与集中的投毒策略。首先，恶意用户将报告分散到所有目标，以促使每个目标进入HG。随后，系统地将报告依次集中在一个目标上，巩固每个v_i的计数，以抵抗由ED策略引起的频率衰减，确保其保留在HG中。

形式化地，一个攻击周期包含$m+1$个时刻，具体如下。

分散阶段（1个时刻）：在时刻t，βn个恶意用户将其报告均匀分布在V中的所有项上。

集中阶段（m个时刻）：在时刻$t+1$，所有βn个用户上报v_1；在时刻$t+2$，上报v_2；依此类推，直到上报v_m。

攻击者重复此周期c次，以最大化目标在HG中存活并被识别为前k个高频项的可能性。

为了明确每个时刻的具体投毒内容，集中阶段上报的$v_1, v_2, \cdots, v_m$严格对应上文中由外部利益设定的目标集合V中的元素。在实际操作中，恶意用户将本地差分隐私协议视为黑盒，在对应的时刻直接向服务器上传当前轮次的目标项v_i，不经过用户端

的LDP扰动过程。这种直接注入目标项的策略对本文研究的BGR、DSR、BDR和CNR协议均完全适用。由于这4种协议底层均基于广义随机响应或其变体，直接发送目标项本身即可在服务端的无偏估计聚合中产生最大的频率增量。因此，攻击者不需要针对每种协议的内部数据结构单独设计复杂的投毒数据格式。

3.3 鲁棒性评估指标

评估LDP协议的准确性通常使用F1分数、召回率或归一化折损累计增益等全局指标，然而，鲁棒性是独立于准确性和可用性之外的第三个评估维度，现有指标难以反映协议面对投毒攻击时的鲁棒性。投毒攻击的目的是将预设的若干目标项推入Top-k列表，只要目标项进入该列表，即攻击成功。此时，内存中大部分位置可能仍被真实的高频项占据，准确性指标变化往往很小，无法真实反映攻击对LDP协议造成的影响。

为了量化所研究算法的鲁棒性，本文采用ASR作为评估攻击有效性的指标。ASR定义为m个目标项集合$\mathcal{V}$中，攻击实施后成功出现在最终Top-k列表H_k中的项所占的比例。其计算式为：

$$\mathrm{ASR}=\frac{|V\cap H_k|}{m} \tag{1}$$

在此定义下，ASR的取值范围为[0,1]。较高的ASR值表示有较高比例的目标项被识别为高频项，表明该LDP协议的鲁棒性较差。

3.4 理论分析

由于LDP随机噪声、数据流的到达顺序差异以及ED策略共同作用，精确推导目标项被识别为高频项的概率极其困难，因此，在上述攻击策略下，本节从理论上推导攻击目标项在HG中的驻留时长，目标项驻留时间越长，则最终被识别为高频项的概率越大。本节通过推导驻留时长，分析不同参数配置对ASR的影响。

引理1 在良性数据流中，对于每个传入项，HG中最弱项被衰减的概率λ_{decay}计算式为：

$$\lambda_{\mathrm{decay}}=\frac{\mathrm{e}^{\varepsilon}+d-1-K-\pi_H(\mathrm{e}^{\varepsilon}-1)}{\mathrm{e}^{\varepsilon}+d-1} \tag{2}$$

其中，d是域大小，ε是隐私预算，K是HG的大小，$\pi_H=\sum_{y\in H}f_y$是当前HG中的项在数据流中的总频率之和，H表示HG集合。

证明 只有当报告的项$y\notin H$时，才会触发HG中最弱项的递减。通过计算单个报告落入H的概率$P(y\in H)$来推导λ_{decay}。该值聚合了已经在H中的项（保持不变或扰动为H中的另一项）和不在H中的项（扰动为H中的项）的贡献。概率为：

$$P(y\in H)=\pi_H[p+(K-1)q]+(1-\pi_H)Kq=\frac{\pi_H(\mathrm{e}^{\varepsilon}-1)+K}{\mathrm{e}^{\varepsilon}+d-1} \tag{3}$$

衰减概率是该事件的补集，即$\lambda_{\mathrm{decay}}=1-P(y\in H)$，由此得出式(2)。证毕。

引理2 对于HG中计数为C_x的项x，在ED策略下，驱逐x所需的预期衰减触发次数为：

$$N_{\mathrm{cost}}(x)\approx\sum_{y\in H,C_y\leqslant C_x}\frac{b}{b-1}b^{C_y} \tag{4}$$

证明 在ED策略下，当触发衰减时，计数为g的项以b^{-g}的概率递减。这意味着平均需要b^g次触发才能将计数从g减少到$g-1$。

对从C_x到0的减少过程求和，单个项x所需的总触发次数为：

$$\sum_{k=1}^{C_x}b^k=\frac{b(b^{C_x}-1)}{b-1}\approx\frac{b}{b-1}b^{C_x} \tag{5}$$

由于ED策略总是针对最小计数，算法必须按顺序处理并驱逐所有满足$C_y\leqslant C_x$的项y。所需的总触发次数是所有此类项的单独需求之和。证毕。

定理1 在第2.2节描述的攻击策略下，目标项v的期望驻留时间$\mathbb{E}[t]$为：

$$\mathbb{E}[t]\approx\frac{\mathrm{e}^{\varepsilon}+d-1}{n\left[(\mathrm{e}^{\varepsilon}+d-1)-(K+\pi_H(\mathrm{e}^{\varepsilon}-1))\right]}\cdot\frac{b}{b-1}\left(b^{c\beta n(1+\frac{1}{m})}+\sum_{y\in\Omega,y\neq v}^{nf_y\leqslant c\beta n(1+\frac{1}{m})}b^{nf_y}\right) \tag{6}$$

其中，n表示用户数量；d表示域大小；ε表示隐私预算；K表示HG大小；项$c\beta n(1+\frac{1}{m})$表示目标项的初始计数，β是恶意用户的比例，m是投毒目标数量，c是攻击重复周期数；求和项计算了所有良性项y的累积衰减成本，这些项的预期累积计数nf_y小于或等于目标项的计数，其中f_y表示项y的全局频率；π_H表示当前驻留在HG中的项的频率之和。

证明 期望驻留时间推导为驱逐目标项所需的总衰减成本与每个时刻的预期衰减率之比。首先，

关于衰减率，在每个时刻中有 n 个用户提交项。单个项触发衰减事件的概率取决于未命中 HG 集合 H 的可能性，受隐私预算 ε 和当前驻留在 HG 中的项的频率之和 π_H 的影响。这导致每个时刻的预期衰减率为 $n\lambda_{\text{decay}}$。其次，关于总衰减成本，ED 策略始终驱逐计数最小的项。因此，为了驱逐目标项 v，系统必须同时驱逐所有累积计数小于或等于目标计数的良性项 y。目标项 v 具有累积计数 $c\beta n(1+\frac{1}{m})$，贡献了与 $b^{c\beta n(1+\frac{1}{m})}$ 成正比的成本。同样地，任何具有全局频率 f_y 的良性项 y 具有预期计数 nf_y，如果 $nf_y \leqslant c\beta n(1+\frac{1}{m})$，则贡献与 b^{nf_y} 成正比的成本。将这些成本求和并除以每个时刻衰减率即得到最终表达式。证毕。

定理 1 给出的期望驻留时间 $\mathbb{E}[t]$ 与本文鲁棒性指标 ASR 之间存在以下联系。设服务器在时刻 T 发起频率查询，目标项 v_i 的实际驻留时间为 t_i，为了在查询时刻成功进入 Top-k 列表，v_i 需在投毒结束后持续存活于数据结构中。因此，v_i 攻击成功的充分条件可近似为 $t_i \geqslant T - t_{\text{attack-end}}$，其中 $t_{\text{attack-end}}$ 为投毒周期结束时刻。因此，$\text{ASR} = \frac{1}{m}\sum_{i=1}^{m}\Pr[t_i \geqslant T - t_{\text{attack-end}}]$。

驻留时间 t_i 由 HG 衰减率与目标项累计计数共同决定，其均值由定理 1 给出，方差受 ε 与 K 影响。在多攻击目标时，各 t_i 因为具有相同的衰减概率 λ_{decay} 而存在相关性。根据概率论的基本性质可知，在方差有限的情况下，期望驻留时间 $\mathbb{E}[t]$ 的增加会使 $\Pr[t_i \geqslant T - t_{\text{attack-end}}]$ 变大。因此，$\mathbb{E}[t]$ 增加会导致 ASR 增加，两者呈正相关，表明对驻留时间的分析能够有效反映 ASR 的变化趋势。基于此，本文的理论推导以驻留时间为核心，旨在为后续实验中 ASR 的表现提供合理的理论依据。

基于上述理论联系，下面进一步分析定理 1 中各核心参数对期望驻留时间及最终攻击成功率的影响。

投毒比例 β 出现在分子中指数式衰减成本项的指数部分。由于底数 b 大于 1，随着 β 的增加，目标项的初始累积计数呈线性增长，而将其驱逐所需的总衰减成本呈指数级上升。因此，目标项的期望驻留时间 $\mathbb{E}[t]$ 随 β 的增加而指数级增长，表明攻击成功率与投毒比例在理论上呈正相关。

隐私预算 ε 主要通过影响衰减概率 λ_{decay} 作用于驻留时间 $\mathbb{E}[t]$。根据引理 1，当 ε 增大时，LDP 机制引入的随机噪声减少，新到达项未命中 HG 集合 H 的概率随之降低。这导致系统触发指数式衰减操作的整体频率下降。分母的减小使期望驻留时间延长，意味着在高隐私预算下，注入的投毒项面临的驱逐压力更小，理论上将导致协议的鲁棒性降低，即攻击成功率随隐私预算的增加而上升。

内存容量 K（即 HG 大小）直接出现在分母的衰减概率表达式中。随着 K 的增大，新到项未命中 HG 集合 H 的概率下降，进而导致触发衰减事件的频率减小。分母的减小直接导致期望驻留时间 $\mathbb{E}[t]$ 的增加。这表明扩大的存储容量在理论上缓解了内部的驱逐压力，使攻击者注入的项能够更稳定地累积计数并占据内存，从而导致攻击成功率上升。

目标数量 m 位于分子指数项的分母位置，在总投毒比例固定的前提下，增加 m 会导致分配给单个目标项的有效注入率成比例下降，这种稀释效应使分子中的总衰减成本呈指数级下降。因此，期望驻留时间 $\mathbb{E}[t]$ 随 m 的增加而迅速缩短，意味着目标项在内存中面临极大的驱逐压力，难以与真实的高频项竞争，理论上多目标攻击下的攻击成功率将随目标数量的增加而大幅下降。

4　鲁棒性评估实验与分析

4.1　实验设置

实验在配置 32 vCPUs 和 60 GB 内存的 Linux 云服务器上进行，所有算法基于 Java 实现，运行环境为 OpenJDK 17.0.16。对于攻击目标项，随机选择排名在 Top-200 之外的项。为了消除 LDP 噪声和恶意用户位置随机性的影响，结果均为 50 次运行的平均值。此外，为了量化其统计可靠性，后续折线图中的阴影区域为误差带，其宽度代表均数标准误（standard error of mean, SEM）。实验分别在 3 个真实数据集上进行，具体信息如下。

Emoji 数据集：源自社交媒体互动的密集数据集，包含 n=1 092 385 条序列，关联词汇量为 d=1 299 个表情符号。

Retail 数据集：来自比利时零售商店，为稀疏购物篮匿名数据集，包含 n=908 576 条交易序列，涉及 d=16 470 种不同类别的商品。

Kosarak 数据集：包含由匈牙利在线新闻网站生成的用户点击数据流，拥有 n=8 019 015 条记录和 d=41 270 个不同的类别。

对于高频项识别任务，ASR 可能取决于以下参数：恶意用户比例 β、隐私预算 ε、HG 大小 K 以及目标项数量 m。除非另有说明，这些参数的默认值为：$\beta = 0.05$，$\varepsilon = 0.5$，$K = 20$，$m = 4$，此外，设置 $k = 20$，$c = 4$。本节将研究每个参数对 LDP 协议鲁棒性的影响，同时将其余参数固定为默认值。LDP 协议中的其他参数遵循其默认或自适应设置。

4.2　投毒比例的影响

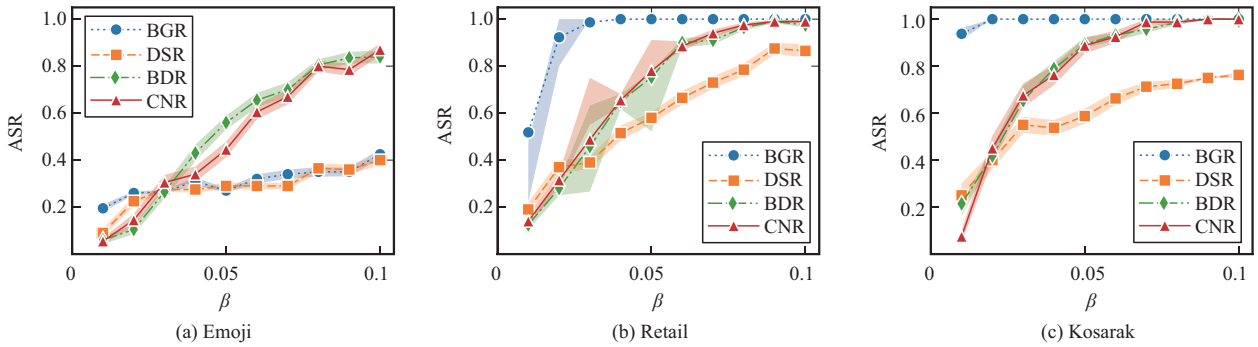
图 2 展示了在 3 个数据集上，4 种 LDP 协议的 ASR 随投毒比例 β 的变化情况。可以观察到，随着 β 从 0.01 增加到 0.1，所有协议的 ASR 均呈现上升趋势，与定理 1 的结论一致。这是由于随着恶意用户比例的增加，目标项的计数累积速率显著提高，当该速率超过指数式衰减速率时，目标项在系统中的期望驻留时间将呈指数级延长，从而使其更容易抵抗衰减驱逐机制并最终进入高频项列表。

此外，在 ASR 上升的参数区间内，呈现出较宽的误差带，该高 SEM 反映了系统正处于投毒攻击的强度与指数式衰减的强度势均力敌的临界状态，目标项能否成功驻留高度依赖于单次运行中的随机事件，导致多次独立实验的结果产生极大差异。当投毒比例进一步增加并完全压倒指数式衰减机制时，ASR 迅速达到饱和，SEM 也随之收敛。

在不同协议的横向对比中，BGR 表现出最弱的鲁棒性。以 Retail 数据集为例，当投毒比例仅为 0.04 时，BGR 的 ASR 迅速饱和达到 1.0。这种脆弱性主要源于 BGR 在原始的完整数据域上运行，去偏公式中的分母 d 极大，巨大的定义域稀释了真正高频项的计数，使攻击者注入的目标项能够以较低的成本在 HG 中驻留。BDR 和 CNR 的鲁棒性表现介于 BGR 与 DSR 之间。尽管这两种协议通过隐私预算分割或双 HG 结构优化了局部扰动过程，增大了真实高频项驻留的概率，但依然将冷域毒数据映射到真实的冷域项目中，为毒数据进入 HG 提供了机会，导致其抗攻击能力有限。

相比之下，DSR 在所有被测协议中展现出最低的 ASR，即最高的鲁棒性，主要原因如下。首先，DSR 利用特殊占位符 $\perp$ 将冷项聚合成一个统一的类别，防止了热数据被扰动为冷域毒数据，从而大幅降低了目标项进入 HG 的概率。同时，域收缩机制大幅减小了去偏公式中的分母。其次，DSR 不分割隐私预算，将全部隐私预算用于项目的扰动，使其去偏因子显著大于 BDR 和 CNR，从而获得了更高的频率估计精度。这种高精度巩固了真实高频项的排序优势，使目标项难以单纯依靠投毒数量来扭转排名。最后，DSR 在数据结构和自适应调节方面的设计进一步限制了攻击效果。与 CNR 提供的缓冲区机制不同，DSR 不设置额外的缓冲区结构，当目标项无法挤入已满的 HG 时会被直接丢弃，切断了毒数据在 HG 外围累积计数并等待进入的路径。这些机制的协同作用使 DSR 在面对集中投毒攻击时表现出最强的防御能力。

实验中观察到的 ASR 与 β 的正相关趋势验证了 3.4 节的理论分析结果。正如定理 1 所证明的，随着恶意用户比例 β 的增加，目标项的累积计数速率显著提高，当该速率超过指数式衰减的速率时，目标项在 HG 中的期望驻留时间 $\mathbb{E}[t]$ 将呈指数级延长，从而使其更容易抵抗衰减驱逐机制并最终进入 Top-k 高频项列表。



(a) Emoji　(b) Retail　(c) Kosarak

图2　ASR随投毒比例的变化情况

4.3　隐私预算的影响

图 3 评估了在 3 个数据集上，ASR 随隐私预算 ε 的变化情况。根据 LDP 研究中的常用参数设置，本节将 ε 的变化范围设置为 0.1～1.0。与传统静态数据协议中常见的“隐私性与安全性折中”现象不同，实验观察到 ε 与 ASR 之间并未表现出明确的单调相关性。随着 ε 的增加，各协议的 ASR 呈现出非单调的波动状态。例如，在 Retail 数据集和 Kosarak 数据集中，BGR 的 ASR 始终维持在 1.0 的饱和状态，而 DSR、BDR 和 CNR 的 ASR 则在一定范围内上下波动，未出现显著的上升或下降趋势。

上述现象表明，在流式数据采集协议中，隐私预算的大小无法对攻击的最终成功率产生决定性影响。其原因在于，流式协议的脆弱性主要来源于底层 HG 的容量限制与指数式衰减逻辑，而非单纯的频率估计偏差。恶意用户绕过 LDP 扰动集中发送目标项，使目标项的累计速率超过指数式衰减速率，此时，ε 仅决定良性用户数据的噪声分布，无法阻止目标项的稳定驻留。

尽管定理 1 指出，较大的 ε 会降低触发衰减的概率，从而在理论上可能延长目标项的驻留时间，但实验中的波动现象表明，在实际的流式协议中，HG 的替换逻辑以及协议自身的设计特点占据主导地位。特别地，以 retail 数据集为例，当 ε 为 0.1～0.5 时，较宽的误差带表明部分协议的实验结果呈现出较大的 SEM，反映了投毒攻击的强度与指数式衰减的强度相近并可能抵消。在这种临界状态下，目标项的计数累积与指数式衰减的速率趋于一致，使系统对随机事件变得极度敏感。此时，LDP 机制引入的扰动与 HG 的指数式衰减的影响被显著放大，例如，数据项在随机化过程中的具体结果，或针对最弱项的计数判断是否执行衰减操作，此类微观的区别决定目标项最终是否被驱逐，使最终的统计结果截然不同，导致多次独立实验的结果差异极大，从而在统计上表现为较大的 SEM。因此，底层的存储机制使 ε 带来的扰动对实际结果的影响大幅减弱，导致其对鲁棒性的影响呈现出非单调的波动趋势。

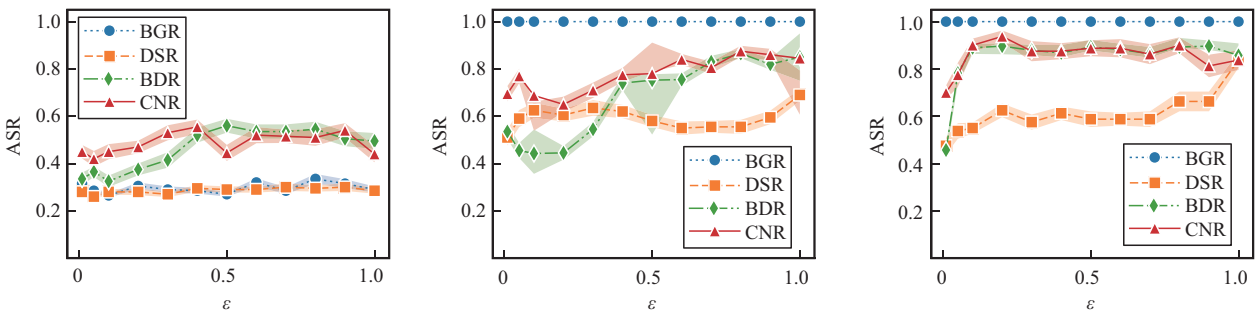
4.4　内存容量的影响

图 4 展示了 ASR 随 HG 大小 K 的变化情况。与“更大的内存通常能提升系统性能”的直觉相反，实验观察到，随着 K 从 20 增加到 200，各协议的 ASR 普遍增加或保持在较高水平。例如，在 Retail 数据集中，随着 K 的增加，DSR 的 ASR 从约 0.60 显著上升至接近 1.0；BGR 则在多个数据集中始终保持在 1.0 的饱和状态。这一结果表明，内存容量增加反而会降低算法鲁棒性，挑战了追求更大内存的传统设计理念。

这种 K 与 ASR 之间的正相关关系验证了定理 1 的理论分析。ASR 的上升主要归因于空间压缩程度对 HG 内指数式衰减机制的影响。当 K 较小时，缓存空间有限，导致新到数据项未命中缓存的概率较高，系统维持较高的固有衰减速率，目标项的累积计数速率难以超过该衰减速率，因而极易被驱逐，只有真实的高频项才能稳定驻留；而攻击者注入的目标项由于初始频率值较低，会频繁触发指数式衰减机制，最终被驱逐。然而，随着 K 的增加，HG 空间增大，缓存未命中的概率降低，触发指数式衰减机制的频率也相应减少，使目标项能够持续累积计数，在 HG 中驻留，并最终进入 Top-k 列表，即扩大的存储容量反而缓解了内部的驱逐压力，导致协议鲁棒性下降。

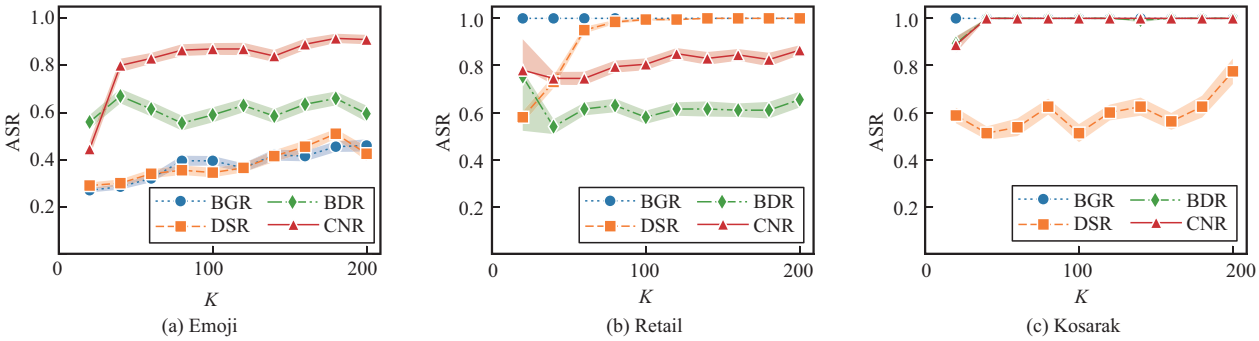
4.5　目标数量的影响

图 5 展示了 ASR 随目标数量 m 的变化情况。可



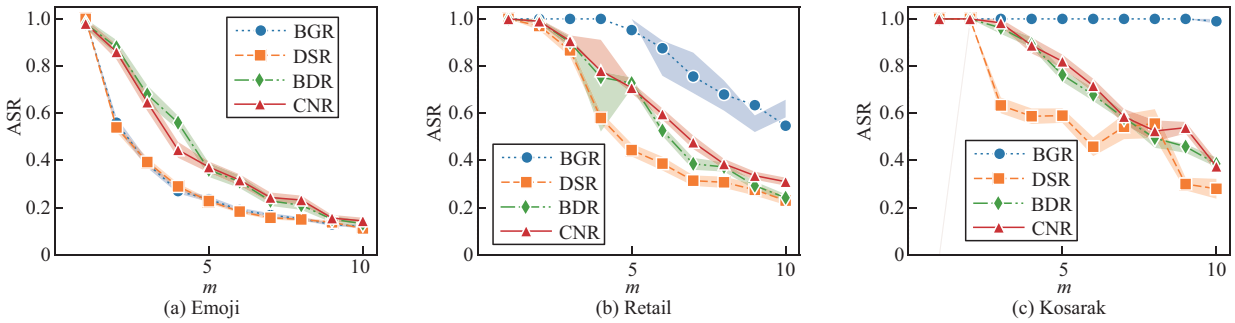
(a) Emoji　(b) Retail　(c) Kosarak

图 3　ASR 随隐私预算 ε 的变化情况



(a) Emoji　(b) Retail　(c) Kosarak

图 4　ASR 随 HG 大小 K 的变化情况



(a) Emoji　(b) Retail　(c) Kosarak

图 5　ASR 随目标数量 m 的变化情况

以观察到，在所有协议中，m 与 ASR 之间均存在显著的负相关关系。当攻击者仅专注于单个目标（m=1）时，大多数协议的 ASR 接近 1.0，表明协议对集中式单目标攻击较为脆弱。然而，随着 m 从 1 增加到 10，ASR 迅速下降。例如，在 Emoji 数据集中，当 m 增加到 6 时，DSR 的 ASR 从接近 1.0 降至 0.2 以下。这一趋势表明，在不增加总恶意用户比例的前提下，攻击目标数量的增加会显著降低攻击的成功率。

这一负相关关系验证了定理 1 的理论分析。这种 ASR 的下降是由攻击者资源的“稀释效应”造成的。由于总投毒比例 β 是固定的，当攻击者试图同时提升多个目标项的排名时，分配给每个目标项的投毒总量会成比例下降（约为 $\frac{1}{m}$）。这种注入频率的降低使单个目标项的计数累积速率大幅减缓，难以抵消因未命中缓存而触发的衰减。因此，这些目标项在 HG 中的计数极易被清零并被驱逐出缓存，从而无法保留在最终的 Top-k 列表中。

5　基于分布一致性的攻击检测

本节提出针对上述投毒攻击的检测方法。在每一时刻 HG 更新之后，检测此时的 HG 是否受到投毒攻击影响。该过程基于数据自身的自洽性实现，不需要依赖外部的信息。

5.1　检测算法

在诚实执行算法的场景下，服务器观测到的数据分布应当严格符合 LDP 机制预期的统计规律。然而，投毒攻击者通常绕过随机扰动直接提交伪造数值，这种行为破坏了数据生成的内在一致性，其结构特征在正常的随机过程中出现的概率极低。基于此，本文提出一种基于分布一致性的检测机制，核心思想在于验证观测到的 HG 是否可能由诚实的 LDP 过程产生。逻辑上，若观测到的 HG 是诚实的，则基于其恢复出的全域分布重新合成 HG 分布，应当产生具有相似统计特性的结果。为量化这种相似性的正常范围，需要引入二次模拟作为参照。

形式化地，设 t 时刻服务器存储的结构为 HG_t，首先恢复出全域直方图 $\boldsymbol{r}_t$，并基于此进行 m 次基准分布合成与重建，生成第一代分布集合 $R_t^{(2)}$。随后，对 $R_t^{(2)}$ 重复该过程，导出第二代分布集合 $R_t^{(3)}$。由于 $R_t^{(2)}$ 与 $R_t^{(3)}$ 均是在完全受控且诚实的条件下生成的，二者之间的差异仅反映了 LDP 噪声和 HG 存储带来的误差，作为系统的基准噪声水平。

检测基于分布间的 L_1 距离实现。设基准组 g_{ben}

为$R_t^{(2)}$与$R_t^{(3)}$间的成对距离，检测组g_{det}为r_t与$R_t^{(2)}$间的距离。若HG_t被污染，其特征无法通过基准分布合成复现，致使g_{det}显著偏离g_{ben}。检测工作流如图6所示，详细步骤见算法1。

算法1　t时刻下基于分布一致性的检测

输入　t时刻的HG_t，服务器端总接收流量N_t，大小为d的数据域Ω，隐私预算ε，显著性水平p_{th}，Simulate(·)所使用的LDP协议Π

输出　二元检测结果$D_t \in \{0,1\}$

1) 初始化两个空列表$R_t^{(2)} \leftarrow [\,]$，$R_t^{(3)} \leftarrow [\,]$

2) $r_t \leftarrow \text{Reconstruct}(\text{HG}_t, N_t, \varepsilon, \Omega)$

3) for $i \in [m]$ do

4)　　$\text{HG}_t^{(i,2)} \leftarrow \text{Simulate}(\Pi, r_t, N_t, \varepsilon, \Omega)$

5)　　$r_t^{(i,2)} \leftarrow \text{Reconstruct}(\text{HG}_t^{(i,2)}, N_t, \varepsilon, \Omega)$

6)　　$\text{HG}_t^{(i,3)} \leftarrow \text{Simulate}(\Pi, r_t^{(i,2)}, N_t, \varepsilon, \Omega)$

7)　　$r_t^{(i,3)} \leftarrow \text{Reconstruct}(\text{HG}_t^{(i,3)}, N_t, \varepsilon, \Omega)$

8)　　$R_t^{(2)}.\text{append}(r_t^{(i,2)})$，$R_t^{(3)}.\text{append}(r_t^{(i,3)})$

9) end for

10) 基准组$g_{\text{ben}} \leftarrow [\,L_1(R_t^{(2)}[i], R_t^{(3)}[i])\,]_{i=1}^{m}$

11) 待检测组$g_{\text{det}} \leftarrow [\,L_1(r_t, R_t^{(2)}[i])\,]_{i=1}^{m}$

12) $p \leftarrow \text{Two\_Sample\_KS\_Test}(g_{\text{det}}, g_{\text{ben}})$[29]

13) if $p < p_{\text{th}}$ then

14)　　return $D_t \leftarrow 1$

15) else

16)　　return $D_t \leftarrow 0$

17) end if

5.2　全域频数直方图重建与基准分布合成

（1）全域频数直方图重建

重建过程旨在从受限的HG结构中恢复全域频数分布。输入包括HeavyGuardian实例HG_t、服务器端接收到的总数据量N_t、大小为d的数据域Ω、隐私预算ε以及LDP协议Π。输出为一个非负直方图$\boldsymbol{r}_t \in \mathbb{R} \geqslant 0^d$，其全域求和结果需严格等于$N_t$。

使用去偏形状和原始总量构建头部。从HG_t中提取前k个高频项组成候选集合head，并获取其去偏估计值C_{shape}。为消除LDP引入的噪声带来的负数估计，对head中的估计值进行非负截断，即$C_{\text{shape}}[x] = \max\{0, C_{\text{shape}}[x]\}$。随后，计算头部去偏估计的总和$S_{\text{shape}} = \sum_{x \in \text{head}} C_{\text{shape}}[x]$。结合$\text{HG}_t$中记录的原始未去偏计数之和$S_{\text{raw}}$，确定头部的实际分配总量$S_{\text{head}} = \min\{S_{\text{raw}}, N_t\}$。基于此，对集合head中的每个元素按去偏估计的比例分配频数：$\boldsymbol{r}_t[x] = \dfrac{S_{\text{head}} \cdot C_{\text{shape}}[x]}{S_{\text{shape}}}$。

根据头部的分布形状拟合Zipf参数。为了推断未被HG记录的低频项分布规律，首先将head集合中的项按去偏估计值$C_{\text{shape}}[x]$降序排列，建立秩1到k的映射关系。然后，基于这些高频项的秩及其在S_{shape}中占据的相对比例，并结合全域大小d构建理论的Zipf分布模型。通过搜索算法寻找一个最优的偏度参数α，使理论模型下Top-k项目的概率质量与实际观测到的头部分布最接近，从而最小化拟合误差。该参数α反映了整体数据分布的倾斜程度，为后续尾部数据的频数分配提供依据。

按Zipf权重将剩余总量分配给尾部。计算尾部待分配的总质量$S_{\text{tail}} = N_t - S_{\text{head}}$，以确保重构直方图的总量与实际接收流量$N_t$严格匹配。提取尾部项集合tail = $\Omega \setminus$ head，并将其按数据域Ω的自然顺序排列为$x'_{k+1}, \cdots, x'_d$。利用拟合得到的偏度参数

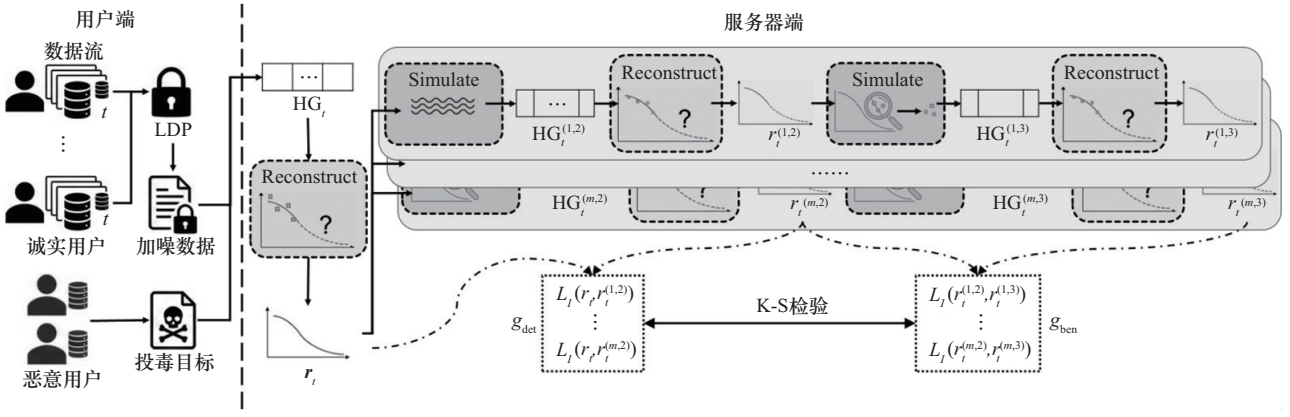


图6　检测工作流

α，计算尾部权重的归一化因子$Z=\sum_{j=k+1}^{d} j^{-\alpha}$。最终，对每个尾部项秩$i\in\{k+1,\cdots,d\}$，根据其对应的Zipf权重分配频数：$\boldsymbol{r}_t[x'_i]=S_{\text{tail}}\cdot i^{-\alpha}/Z$。

（2）基准分布合成

该合成过程旨在基于重构出的数据分布，复现HG实例，以验证观测数据的统计合理性。形式化地，该过程接收重构的全域直方图$\boldsymbol{r}_t$、总数据量N_t以及LDP协议Π作为输入，并输出一个合成的HG实例HG_t^{syn}。

生成HG_t^{syn}的具体步骤如下。首先，将归一化的直方图$\frac{\boldsymbol{r}_t}{N_t}$视为全域的概率质量函数，并从中独立抽取$N_t$个样本，形成模拟的原始数据流。随后，针对每一个样本，应用LDP协议Π定义的用户端随机化机制进行扰动，扰动后的数据被依次插入一个初始化为空的HG结构中。最终得到的HG_t^{syn}反映了所有用户均诚实参与下的理想情况。

5.3　检测效果评估

本节将探索不同系统参数配置对检测效果的影响。具体而言，分别评估了投毒比例β、隐私预算ε、HG容量K以及投毒目标数量m变化时，检测成功率的变化趋势。

（1）实验设置

使用与4.1节相同的数据集和参数设置，其中用于判定分布一致性的假设检验显著性水平统一设定为$p_{\text{th}}=0.05$，该数值是统计假设检验中广泛采用的经验阈值[30]，旨在将检验过程产生误判的概率上限严格控制在5%以内。检测成功率定义为正确识别出攻击是否存在的比例。对于每种配置，进行20次独立实验，其中包括10次无攻击的场景和10次存在攻击的场景。

目前尚无针对面向数据流场景的高频项检测任务的投毒攻击检测方案，本文适配了基于高频项集挖掘的检测方法[5]作为对比方案。为适配本文场景，将同一时刻所有用户提交的项目集合视为一个“事务”，以统计各个项集共同出现的次数；利用切比雪夫不等式计算出正常情况下项集共现次数的阈值，若挖掘出的项集共现次数超过该阈值，则判定此时刻存在投毒攻击。

对于4个LDP协议中的任一协议，令p表示在该协议的随机化机制下，用户如实报告真实值的概率。在无攻击假设下，N个用户对该值的报告次数服从二项分布$\text{Binomial}(N,p)$。给定允许的假阳率η，检测阈值为置信区间上界，表示为：

$$T=N\cdot p+\sqrt{\frac{N\cdot p\cdot(1-p)}{\eta}}$$

若服务端统计的原始高频项最大计数值超过T，则判定存在投毒攻击。对于4个协议的p，具体计算式分别为：

$$P_{\text{BGR}}=\frac{\mathrm{e}^{\varepsilon}}{\mathrm{e}^{\varepsilon}+d-1}$$

$$P_{\text{DSR}}=\frac{\mathrm{e}^{\varepsilon}}{\mathrm{e}^{\varepsilon}+K}$$

$$P_{\text{BDR}}=p_1\cdot(p_2-q_2)$$

$$P_{\text{CNR}}=p_1\cdot(p_2-q_2)$$

$$p_1=\frac{\mathrm{e}^{\varepsilon\alpha}}{\mathrm{e}^{\varepsilon\alpha}+1}$$

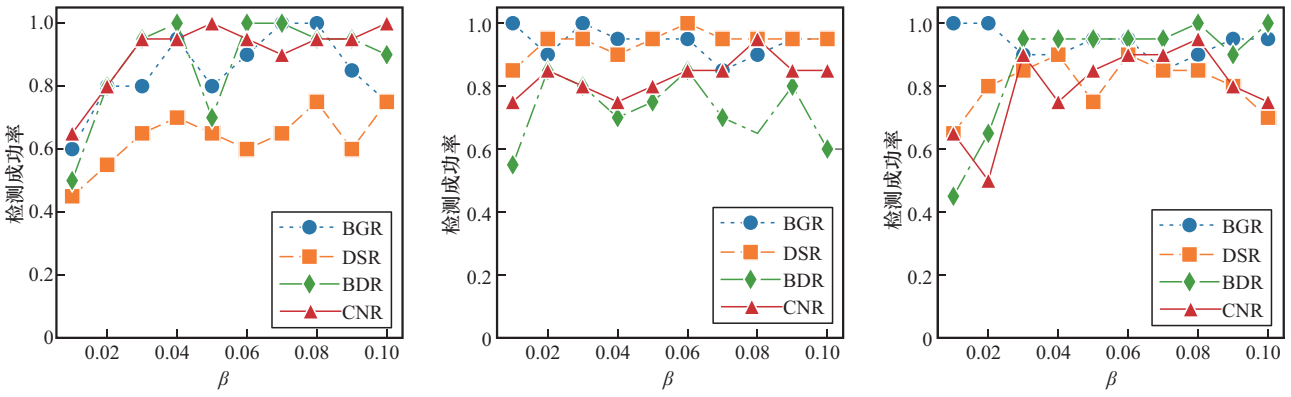
$$p_2=\frac{\mathrm{e}^{\varepsilon(1-\alpha)}}{\mathrm{e}^{\varepsilon(1-\alpha)}+K-1}$$

$$q_2=\frac{1}{\mathrm{e}^{\varepsilon(1-\alpha)}+K-1}$$

其中，ε为隐私预算，d为数据域大小，α为第一阶段隐私预算分割比例，K为HG的容量。

（2）投毒比例对检测成功率的影响

图7展示了检测成功率与投毒比例β的关系。随着β的增加，注入的虚假数据量增长，产生了偏离预期LDP噪声分布的更强信号。在不同协议的横向对比中，检测方案在BGR上大多表现出整体最高的检测成功率，而DSR、BDR和CNR则相对较难被检测。这种差异主要源于各协议在本地扰动时采用的数据域大小，BGR在完整的原始数据域上执行随机化，良性数据被扰动后分散在整个大域中，而投毒数据集中于少数目标项，使攻击信号非常显著。相比之下，DSR、BDR和CNR采用了域收缩或隐私预算分割机制，将大部分扰动限制在高频项域内，使良性数据的扰动结果也高度集中在少数项上，因此投毒数据所表现出的集中特征，更易与优化协议下良性数据的集中分布相混淆，从而大幅降低检测的成功率。CNR利用服务器端候选列表来过滤噪声并保留真实的高频项，这种优化使良



(a) Emoji　(b) Retail　(c) Kosarak

图 7　检测成功率与投毒比例 β 的关系

性分布在统计上平滑且准确，使攻击流量能够作为有效数据混入。

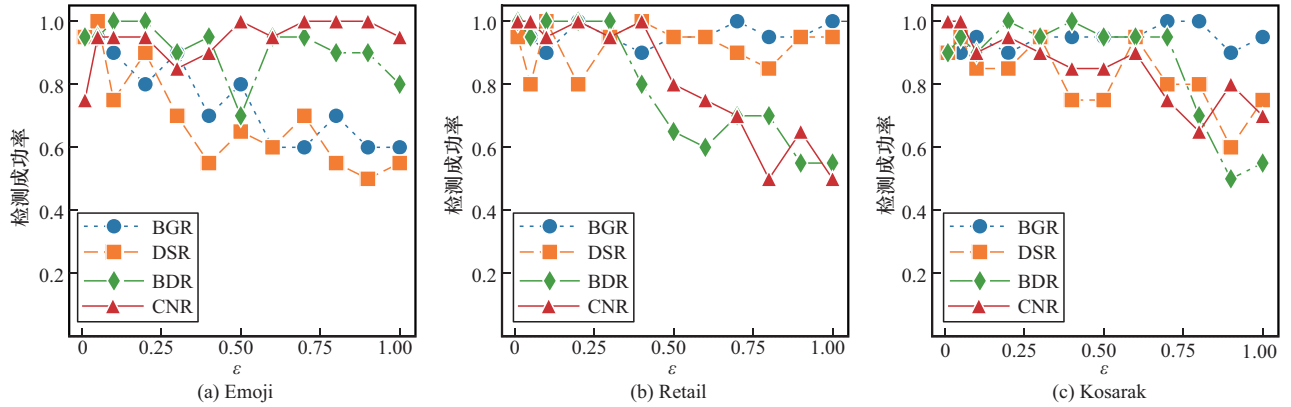
表 1 展示了在 Kosarak 数据集上，不同投毒比例下，本文方案与对比方案对 4 个协议的检测成功率对比结果。当投毒比例较低（如 0.01～0.06）时，对比方案在 DSR、BDR 和 CNR 上的检测成功率普遍停留在 0.50 左右。当投毒比例较低时，有毒数据的规模较小，在单轮统计周期内，毒数据的触发频率无法达到对比方案设定的检测阈值，导致基于频率的检测器完全失效，将所有攻击样本均判定为无攻击（即假阴性率为 100%）。此时，系统仅能正确识别无攻击样本，因而检测成功率退化为 0.50。只有当投毒比例较高（大于或等于 0.08）时，对比方案检测成功率才出现显著提升。相比之下，本文方案即使在低投毒比例下，依然能够在多数情况下维持 0.80 甚至 0.90 以上的检测成功率。这表明，本文方案对小规模投毒攻击具有更高的敏感性，能够有效弥补对比方案在低投毒比例下检测能力不足的问题。

表 1　不同投毒比例下的检测成功率对比

投毒比例	BGR		DSR		BDR		CNR	
	对比方案	本文方案	对比方案	本文方案	对比方案	本文方案	对比方案	本文方案
0.01	0.70	1.00	0.50	0.65	0.55	0.45	0.50	0.65
0.02	0.90	1.00	0.50	0.80	0.50	0.65	0.50	0.50
0.03	0.95	0.90	0.50	0.85	0.50	0.95	0.50	0.90
0.04	0.95	0.90	0.50	0.90	0.50	0.95	0.50	0.75
0.05	0.95	0.95	0.50	0.75	0.50	0.95	0.50	0.85
0.06	0.90	0.95	0.85	0.90	0.50	0.95	0.50	0.90
0.07	0.90	0.85	1.00	0.85	0.50	0.95	0.50	0.90
0.08	0.85	0.90	1.00	0.85	1.00	1.00	1.00	0.95
0.09	0.90	0.95	1.00	0.80	1.00	0.90	1.00	0.80
0.10	0.95	0.95	1.00	0.70	1.00	1.00	1.00	0.75

（3）隐私预算对检测成功率的影响

图 8 展示了检测成功率与隐私预算 ε 的关系。本文观察到随着隐私预算 ε 增加，检测成功率呈现总体下降趋势。这是因为在 ε 较小的高噪声区域，



(a) Emoji　(b) Retail　(c) Kosarak

图 8　检测成功率与隐私预算 ε 的关系

良性数据发生扰动的概率高导致整体分布较为分散，而攻击数据未经过扰动导致整体分布更集中，与良性数据的分布存在明显偏差，使攻击更容易被检测。随着ε增加，良性数据中的噪声减少，由原来分散的分布向真实数据的集中分布收敛，与攻击数据的分布更相似，导致检测难度更大，从而使检测成功率呈总体下降趋势。

BGR 的检测成功率总体保持相对较高，其他协议的检测成功率随着ε增大出现显著下降。这是因为BGR 使用了巨大的原始数据域，即使在较大的隐私预算下整体扰动概率降低，良性数据在全域的分布也十分分散，与投毒数据的分布存在明显差距。相反，DSR、BDR 和 CNR 将扰动限制在缩减的高频项域内，ε的增大会使良性数据进一步向局部的少数项集中，良性数据与投毒数据分布几乎难以区分，导致检测器的性能迅速下降。

表 2 展示了在 Kosarak 数据集上，不同隐私预算下本文方案与对比方案对 4 个协议的检测成功率对比。对比方案仅对 BGR 协议能够有效识别攻击，而在 DSR、BDR 和 CNR 协议中，无论隐私预算如何变化，其检测成功率始终停留在 0.50，基本无法检测出攻击。相比之下，本文方案在各种隐私预算下均表现出较强的鲁棒性，特别是在 DSR、BDR 和 CNR 协议中，本文方案显著优于对比方案，在多数隐私预算（0.1～0.7）下能够将检测成功率维持在 0.75 甚至 0.90 以上。尽管在较高的隐私预算（如 0.9 和 1.0）下，本文方案在部分协议（如 BDR 和 CNR）中的检测成功率出现了一定程度的下降，但整体性能依然大幅领先于对比方案。这表明本文方案能够在不同的隐私保护强度下，稳定且有效地检测出投毒攻击。

表 2　不同隐私预算下的检测成功率对比

隐私预算	BGR		DSR		BDR		CNR	
	对比方案	本文方案	对比方案	本文方案	对比方案	本文方案	对比方案	本文方案
0.1	1.00	0.95	0.50	0.85	0.55	0.90	0.50	0.90
0.2	1.00	0.90	0.50	0.85	0.50	1.00	0.50	0.95
0.3	1.00	0.95	0.50	0.95	0.50	0.95	0.50	0.90
0.4	1.00	0.95	0.50	0.75	0.50	1.00	0.50	0.85
0.5	1.00	0.95	0.50	0.75	0.50	0.95	0.50	0.85
0.6	1.00	0.95	0.50	0.95	0.50	0.95	0.50	0.90
0.7	1.00	1.00	0.50	0.80	0.50	0.95	0.50	0.75
0.8	1.00	1.00	0.50	0.80	0.50	0.75	0.50	0.65
0.9	1.00	0.90	0.50	0.60	0.50	0.55	0.50	0.80
1.0	1.00	0.95	0.50	0.75	0.50	0.55	0.50	0.70

（4）内存容量对检测成功率的影响

图 9 展示了检测成功率与 HG 大小 K 的关系。与“HG 越大，服务器获取信息越多，则恢复的分布越准确，检测的成功率应越高”的直觉不同，随着 K 从 20 增大到 200，检测成功率整体下降，这是由“稀释”效应引起的。具体而言，检测机制依赖于重建全局分布以验证一致性，在攻击数据的分布中，投毒的存在使 HG 中数据频率增加，挤占了非 HG 的尾部数据的频率，总体分布上呈现出头部分布的频率高、尾部分布的频率低的现象。而较大的 K 使头部数据的频率增加，尾部数据频率降低，让尾部数据的分布接近攻击数据中尾部数据的分布。这种全局稳定性掩盖了由固定数量的投毒项引入的局部失真，降低了检测器的灵敏度。

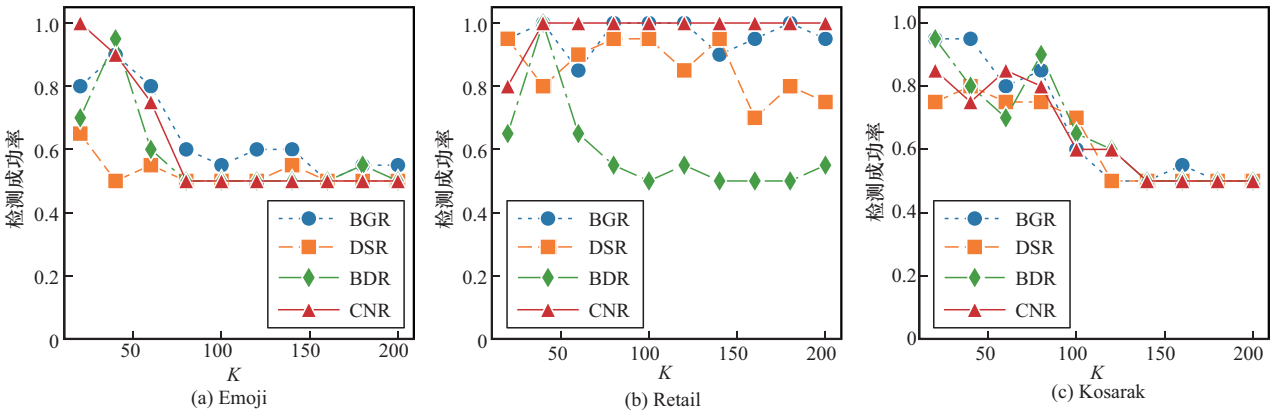


图 9　检测成功率与 HG 大小 K 的关系

表 3 展示了在 Kosarak 数据集上，不同内存容量下本文方案与对比方案对 4 个协议的检测成功率对比。对比方案在内存容量为 20 时，4 个协议的检测成功率均停留在 0.50，无法有效识别攻击；仅在内存容量为 40～100（或 120）的特定范围内，对部分协议能够有效识别攻击；当内存容量大于等于 140 时，其检测成功率再次回落至 0.50，基本无法检测出攻击。这是由于较大内存容量使检测阈值被压缩得极低，检测器会过度敏感，将大量正常的数据波动也误判为投毒攻击（即假阳性为 100%），导致检测成功率为 0.50。相比之下，本文方案在内存容量较低（20～80）时表现出较强的检测能力，特别是在 BGR、BDR 和 CNR 协议中，检测成功率大部分能够达到 0.80 甚至 0.95 以上。在内存容量为 20 时，本文方案在 BGR 和 BDR 中的检测成功率均达到 0.95，显著优于对比方案。随着内存容量的增加，本文方案的检测成功率逐渐下降，在内存容量为 100～120 时已出现明显下滑，当内存容量大于或等于 140 时，检测成功率同样停留在 0.50 左右。这表明本文方案在内存资源受限的条件下具有更好的检测能力，能够有效弥补对比方案在受限内存下检测能力不足的问题。

表 3　不同内存容量下的检测成功率对比

内存容量	BGR		DSR		BDR		CNR	
	对比方案	本文方案	对比方案	本文方案	对比方案	本文方案	对比方案	本文方案
20	0.50	0.95	0.50	0.75	0.50	0.95	0.50	0.85
40	1.00	0.95	1.00	0.80	1.00	0.80	1.00	0.75
60	1.00	0.80	1.00	0.75	1.00	0.70	0.50	0.85
80	1.00	0.85	1.00	0.75	1.00	0.90	1.00	0.80
100	1.00	0.60	0.50	0.7	1.00	0.65	0.50	0.60
120	0.50	0.50	0.50	0.50	1.00	0.60	0.50	0.60
140	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
160	0.50	0.55	0.50	0.50	0.50	0.50	0.50	0.50
180	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
200	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50

（5）目标数量对检测成功率的影响

图 10 揭示了检测成功率与目标集合大小 m 的关系。这种现象归因于攻击能量的分散。由于总投毒比例 β 是固定的，增加 m 使攻击者将虚假频率计数分布在更多的项上，从而降低了单个异常的幅度。与 BGR 相比，这种效应在 DSR 和 CNR 中尤为显著。在 BGR 中，背景噪声非常高，以至于任何系统性的注入模式仍然明显。然而，对于保持更平滑分布的 DSR 和 CNR，将攻击分散在多个目标项（即 $m > 1$）上模仿了数据的自然重尾特征。通过模拟一组中等频率的项而不是单个离群值，攻击者有效地利用了算法保留尾部信息的能力，从而成功逃避了检测。

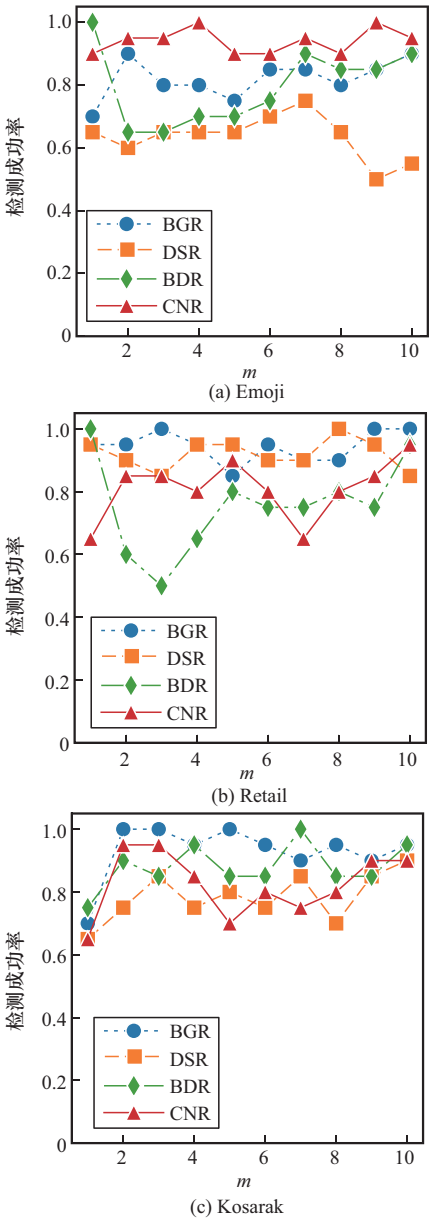


图 10　检测成功率与目标集合大小 m 的关系

表 4 展示了在 Kosarak 数据集上，不同目标数量下本文方案与对比方案对 4 个协议的检测成功率

对比。对比方案仅对BGR协议能够有效识别攻击，而在DSR、BDR和CNR中，无论目标数量如何变化，其检测成功率始终停留在0.50，基本无法检测出攻击。相比之下，本文方案在各种目标数量下均表现出较强的鲁棒性，特别是在DSR、BDR和CNR协议中，本文方案显著优于对比方案，在多数目标数量（2～10）下能够将检测成功率大部分维持在0.75甚至0.85以上。在BDR协议中，本文方案的检测成功率尤为突出，在目标数量为2～10时基本稳定在0.85以上，最高达到1.00。在BGR协议中，本文方案仅在目标数量为1时检测成功率略低（为0.70），但在目标数量大于1后，检测成功率迅速提升至0.90以上，整体表现接近对比方案。这表明本文方案能够在不同的目标数量条件下，稳定且有效地检测出投毒攻击。

表4 不同目标数量下的检测成功率对比

目标数量	BGR		DSR		BDR		CNR	
	对比方案	本文方案	对比方案	本文方案	对比方案	本文方案	对比方案	本文方案
1	1.00	0.70	0.50	0.65	0.50	0.75	0.50	0.65
2	1.00	1.00	0.50	0.75	0.50	0.90	0.50	0.95
3	1.00	1.00	0.50	0.85	0.50	0.85	0.50	0.95
4	0.95	0.95	0.50	0.75	0.50	0.95	0.50	0.85
5	0.95	1.00	0.50	0.80	0.50	0.85	0.50	0.70
6	1.00	0.95	0.50	0.75	0.50	0.85	0.50	0.80
7	0.95	0.90	0.50	0.85	0.50	1.00	0.50	0.75
8	1.00	0.95	0.50	0.70	0.50	0.85	0.50	0.80
9	1.00	0.90	0.50	0.85	0.50	0.85	0.50	0.90
10	1.00	0.95	0.50	0.90	0.50	0.95	0.50	0.90

6 结束语

本文针对LDP在实际应用中面临的数据投毒攻击威胁，针对现有研究多关注静态数据协议，对内存受限场景下采用数据压缩技术的动态流式数据采集分析协议鲁棒性分析不足的问题，以BGR、DSR、BDR、CNR这4种LDP流式高频项识别协议为研究对象，提出攻击驱动的鲁棒性评估框架，并在此框架下对上述协议进行鲁棒性评估。理论和实验发现DSR协议鲁棒性最高，同时，发现存储空间大小对协议鲁棒性有深远的影响。本文进一步提出一种不依赖具体协议设计的基于分布一致性的攻击检测方案，通过数据频率重建技术解决内存受限的情况下全域频率信息缺失的问题，实验表明该方案在隐私预算较小或投毒比例较大时检测效果显著。为设计更具鲁棒性的LDP流式数据采集分析协议和有效防御机制提供了重要理论依据与实践参考。

参考文献：

[1] Erlingsson Ú, Pihur V, Korolova A. RAPPOR: randomized aggregatable privacy-preserving ordinal response[C]//Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2014: 1054-1067.

[2] Apple Differential Privacy Team. Learning with privacy at scale[R]. (2017-12-06) [2026-06-03].

[3] Ding B L, Kulkarni J, Yekhanin S. Collecting telemetry data privately[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. New York: ACM Press, 2017: 3574-3583.

[4] Cheu A, Smith A, Ullman J. Manipulation attacks in local differential privacy[C]//Proceedings of the 2021 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2021: 883-900.

[5] Cao X, Jia J, Gong N Z. Data poisoning attacks to local differential privacy protocols[C]//Proceedings of the 30th USENIX Conference on Security Symposium. Berkeley: USENIX Association, 2021: 947–964.

[6] Li X G, Li N H, Sun W H, et al. Fine-grained poisoning attack to local differential privacy protocols for mean and variance estimation[C]//Proceedings of the 32nd USENIX Conference on Security Symposium. New York: ACM Press, 2023: 1739-1756.

[7] Li X G, Li Z T, Li N H, et al. On the robustness of LDP protocols for numerical attributes under data poisoning attacks[C]//Proceedings 2025 Network and Distributed System Security Symposium. Piscataway: IEEE Press, 2025: 1-15.

[8] Wu Y, Cao X, Jia J, et al. Poisoning attacks to local differential privacy protocols for key-value data[C]//Proceedings of the 31st USENIX Conference on Security Symposium. Berkeley: USENIX Association, 2022: 535-552.

[9] Li X C, Liu W R, Lou J, et al. Local differentially private heavy hitter detection in data streams with bounded memory[J]. Proceedings of the ACM on Management of Data, 2024, 2(1): 1-27.

[10] Li W H. Pandora: an efficient and rapid solution for persistence-based tasks in high-speed data streams[J]. Proceedings of the ACM on Management of Data, 2025, 3(1): 1-26.

[11] Bao E, Yang Y, Xiao X K, et al. CGM: an enhanced mechanism for streaming data collection with local differential privacy[J]. Proceedings of the VLDB Endowment, 2021, 14(11): 2258-2270.

[12] Joseph M, Roth A, Ullman J, et al. Local differential privacy for evolving data[C]//Proceedings of the 32nd International Conference on Neural Information Processing Systems. New York: ACM Press, 2018: 2381-2390.

[13] Ren X B, Shi L, Yu W R, et al. LDP-IDS: local differential privacy for infinite data streams[C]//Proceedings of the 2022 International Conference on Management of Data. New York: ACM Press, 2022: 1064-1077.

[14] Wang T H, Chen J Q, Zhang Z K, et al. Continuous release of data streams under both centralized and local differential privacy[C]//Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2021: 1237-1253.

[15] Sánchez-Macián A, Martínez J, Reviriego P, et al. On the privacy of the count-min sketch: extracting the top-K elements[J]. IEEE Transactions on Emerging Topics in Computing, 2024, 12(4): 1056-1065.

[16] Wang X J, Mo L, Guo L K, et al. Online streaming sampling publication method over sliding windows with differential privacy[J]. IEEE Transactions on Dependable and Secure Computing, 2025, 22(6): 6896-6912.

[17] Li J Z, Li Z K, Xu Y F, et al. WavingSketch: an unbiased and generic

sketch for finding top-k items in data streams[C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM Press, 2020: 1574-1584.
[18] Yang T, Jiang J, Liu P, et al. Elastic sketch: adaptive and fast network-wide measurements[C]//Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication. New York: ACM Press, 2018: 561-575.
[19] Metwally A, Agrawal D, Abbadi A E. Efficient computation of frequent and top-k elements in data streams[M]. Berlin: Springer, 2004.
[20] Li X Y, Ren X B, Yang S S, et al. Fine-grained manipulation attacks to local differential privacy protocols for data streams[J]. IEEE Transactions on Knowledge and Data Engineering, 2026, 38(3): 1768-1782.
[21] Duchi J C, Jordan M I, Wainwright M J. Local privacy and statistical minimax rates[C]//Proceedings of the 2013 IEEE 54th Annual Symposium on Foundations of Computer Science. Piscataway: IEEE Press, 2013: 429-438.
[22] Dwork C, Naor M, Pitassi T, et al. Differential privacy under continual observation[C]//Proceedings of the Forty-Second ACM Symposium on Theory of Computing. New York: ACM Press, 2010: 715-724.
[23] Wang T H, Blocki J, Li N H, et al. Locally differentially private protocols for frequency estimation[C]//Proceedings of the 26th USENIX Conference on Security Symposium. New York: ACM Press, 2017: 729-745.
[24] Feng W Q, Gao J Q, Chen X Q, et al. F3: fast and flexible network telemetry with an FPGA coprocessor[J]. Proceedings of the ACM on Networking, 2024, 2(4): 1-22.
[25] Liang J C, Du Y, Huang H, et al. Memory-efficient and hardware-friendly sketches for hierarchical heavy hitter detection[J]. IEEE Transactions on Network and Service Management, 2026, 23: 582-593.
[26] Jarlow V, Stylianopoulos C, Papatriantafilou M. QPOPSS: query and parallelism optimized space-saving for finding frequent stream elements[J]. Journal of Parallel and Distributed Computing, 2025, 204: 105134.
[27] Ma T Y, Gao G J, Huang H, et al. Scout sketch: finding promising items in data streams[C]//Proceedings of the IEEE INFOCOM 2024 - IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2024: 1561-1570.
[28] Yang T, Gong J Z, Zhang H W, et al. HeavyGuardian: separate and guard hot items in data streams[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM Press, 2018: 2584-2593.
[29] Massey F J J. The Kolmogorov-Smirnov test for goodness of fit[J]. Journal of the American Statistical Association, 1951, 46(253): 68-78.
[30] Wasserman L. All of statistics: a concise course in statistical inference[M]. New York: Springer New York, 2004.

[作者简介]

田月池（1998– ），女，中国科学院信息工程研究所博士生，主要研究方向为隐私计算、隐私保护效果评估。

彭锦钰（2003– ），男，中国科学院信息工程研究所硕士生，主要研究方向为数据安全、隐私计算。

李效光（1995– ），男，博士，西安电子科技大学讲师、硕士生导师，主要研究方向为差分隐私算法设计和安全性分析。

张芊龙（2000– ），男，中国科学院信息工程研究所博士生，主要研究方向为数据安全、隐私计算。

李凤华（1966– ），男，博士，中国科学院信息工程研究所研究员、博士生导师，主要研究方向为网络与系统安全、信息保护、隐私计算。

左金鑫（1992– ），女，博士，中国科学院信息工程研究所副研究员、硕士生导师，主要研究方向为数据安全、隐私计算。

牛犇（1984– ），男，博士，中国科学院信息工程研究所研究员、博士生导师，主要研究方向为数据安全、隐私计算。