Differentially Private User Data Perturbation with Multi-level Privacy Controls

Yilin Shen(✉), Rui Chen, and Hongxia Jin

Samsung Research America, Mountain View, CA 94043, USA
{yilin.shen,rui.chen1,hongxia.jin}@samsung.com

Abstract. Service providers typically collect user data for profiling users in order to provide high-quality services, yet this brings up user privacy concerns. One hand, service providers oftentimes need to analyze multiple user data attributes that usually have different privacy concern levels. On the other hand, users often pose different trusts towards different service providers based on their reputation. However, it is unrealistic to repeatedly ask users to specify privacy levels for each data attribute towards each service provider. To solve this problem, we develop the *first* lightweight and provably framework that not only guarantees differential privacy on both *service provider* and *different data attributes* but also allows configurable *utility functions* based on service needs. Using various large-scale real-world datasets, our solution helps to significantly improve the utility up to 5 times with negligible computational overhead, especially towards numerous low reputed service providers in practice.

Keywords: Differential privacy · Multi-level privacy · Optimization

1 Introduction

The last few decades have witnessed a variety of personalized services to users, such as intelligent assistant, targeted advertising and so on, which has become key business drivers for many companies. As one can understand, such services are based on user's data and oftentimes require substantial user data in order to provide high-quality services. However, consumer fears over privacy continue to escalate due to the release of users' private data. Based on Pew Research [1], 68 % consumers think that current laws are insufficient to protect their privacy and demand tighter privacy laws; and 86 % of Internet users have taken proactive steps to remove or mask their digital footprints. Responding to increasing user privacy concerns, governments in US/EU are increasing regulations and applying/enforcing existing regulations.

More importantly, in order to provide high quality services, service providers usually profile users by analyzing multiple attributes of their private data. Recent research has showed that various attributes of data are often associated with different privacy concerns [13,24,26]. More importantly, Zhang *et al.* [26] revealed that user's perception of privacy concerns will dramatically decrease if providing them fine-grained privacy controls for different attributes of data.

P. Frasconi et al. (Eds.): ECML PKDD 2016, Part II, LNAI 9852, pp. 112–128, 2016.
DOI: 10.1007/978-3-319-46227-1_8

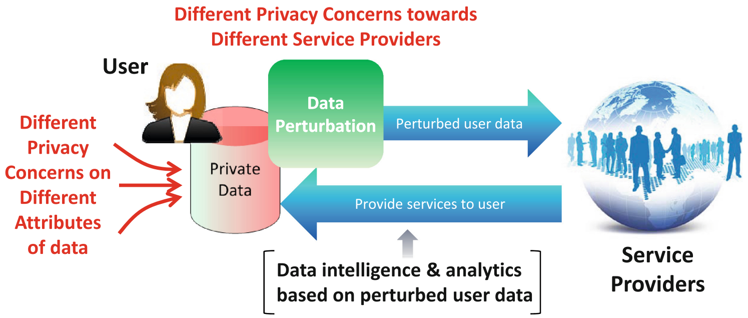

Different Privacy Concerns towards Different Service Providers
User
Different Privacy Concerns on Different Attributes of data
Private Data
Data Perturbation
Perturbed user data
Provide services to user
Data intelligence & analytics based on perturbed user data
Service Providers

Fig. 1. Data Perturbation with Multi-Level Privacy Controls under Untrusted Server

On the other hand, while traditionally users count on service providers to protect their data privacy, recent years have witnessed a variety of privacy breaches through service providers when malicious attackers break into the cloud/server and steal user data. Target, HomeDepot, and Anaheim health insurance companies are among the largest hits. Huge number of sensitive user data is leaked through servers. Additionally, the insiders of service providers are another source of privacy threat. It would be ideal if users do not have to fully trust the service providers to protect their data; and users can impose different privacy concerns based on each service provider's reputation according to recent research [15], i.e., trust Google more than aforementioned intruded service providers.

However, it is unrealistic to repeatedly ask ordinary users to specify privacy levels for each attribute of data every time releasing to different service providers. Therefore, it is critically desirable to develop technologies that not only allow business intelligence but also preserve users' privacy needs toward both different data attributes and different service providers.

In this paper, we aim to develop the *first* lightweight and provably private framework, under *untrusted server* settings, to automate users multi-level privacy controls for releasing the aggregates of attributes associated with their private data to each service provider. As shown in Fig. 1, our adoption of *untrusted server* setting, in which user data is perturbed and anonymized on their private devices before releasing, enjoys a number of benefits as discussed in [23]. In the meanwhile, these protections should be done to still provide different reasonable utilities of perturbed data based on service needs. Our approach is developed to provide a strong and provable privacy guarantee, *differential privacy*, which is the current state-of-the-art paradigm for privacy-preserving data publishing.

Our contributions are summarized as follows:

– We formulate a novel *Multi-Level User Privacy Perturbation (MultiUPP)* problem, which aims to release perturbed aggregates on user data attributes that not only preserves both *differential privacy towards a service provider (overall privacy)* and *differential privacy on each data attribute (per-attribute privacy)*, but also optimizes a specific utility function based on service needs.

- We analyze the lower bound of overall privacy guarantee with optimal utility, as well as the lower bounds of utility loss.
- We propose a novel *Multi-Level Differential Privacy (MultiDP)* mechanism to understand the condition between utility loss and overall and per-attribute privacy preservation. Using MultiDP mechanism, we develop a novel *Differentially Private Multi-Level User Privacy Perturbation (DP-MultiUPP)* framework which allows to plug in different utility objectives. We prove theoretical guarantee on privacy, utility and time complexity.
- We conduct extensive experiments on various large-scale real-world datasets. Our solution is shown to outperform the state-of-the-art approach up to 5 times with negligible computational overhead on both PC and Android smartphones. Particularly, the utility is significantly improved toward low reputed service providers in practice.

The rest of paper is organized as follows: Sect. 2 presents notations, preliminaries and problem definition. Section 3 provides the lower bounds of utility loss and overall privacy budget. Section 4 develops the DP-MultiUPP framework via a novel Multi-Level Differential Privacy Mechanism. Experimental results and related work are presented in Sects. 5 and 6. Finally, Sect. 7 concludes the whole paper and discusses future work.

2 Preliminaries and Problem Definition

In this section, we first introduce notations and restate the definition and existing mechanism of differential privacy. Then, we define a novel *Multi-Level User Privacy Perturbation (MultiUPP)* problem definition, along with its challenges.

2.1 Notations

Let I be the public set/universe of items of size $|I| = n$. A user's raw private data is denoted as a vector $\mathbf{d^r}$ of dimension n. The i^{th} entry in $\mathbf{d^r}$ is either 1 or 0, meaning that item i does or does not belong to user's private/raw history data. Public attribute set is defined as A of size $|A| = m$, in which each item is associated with a subset of attributes represented by a public item-attribute matrix $\mathbf{A}$ of dimension $n \times m$. The entry a_{ij} in $\mathbf{A}$ is the value that item i has for attribute j. For the attributes in A, we define their private aggregate vectors to be $\mathbf{a^r}$ such that $\mathbf{a^r} = \mathbf{A}^T\mathbf{d^r}$. The published perturbed attribute histogram is presented as a vector $\mathbf{a^p}$ (details in utility objectives of problem definition in Sect. 2.3).

This user's multi-level privacy concern on different attributes is denoted as a vector $\mathbf{t} = (t_1, \ldots, t_m)$, in which the j^{th} entry $t_j > 0$ means the privacy budget of attribute j. This user's overall privacy concern towards the service provider is defined as $\epsilon > 0$. A smaller t_j or ϵ means a higher privacy concern (a stronger privacy guarantee) on attribute j or towards the service provider. (Note that all t_j and ϵ correspond to the privacy budget in differential privacy notion, defined in the next subsection.) For reference, we list all notations in Table 1.

Table 1. Notations

Symbol	Description
I	public item set/universe of size $\|I\| = n$
A	public attribute set of size $\|A\| = m$
$\mathbf{A}$	public item-attribute matrix $\mathbf{A} \in \{0,1\}^{n \times m}$
$\mathbf{d^r}$	user private item vector $\mathbf{d^r} \in \{0,1\}^n$
$\mathbf{a^r}$	user private attribute aggregate vector $\mathbf{a^r} \in \mathbb{R}^{*m}$
	($\mathbb{R}^*$: non-negative real numbers)
$\mathbf{a^p}$	user perturbed attribute aggregate vector $\mathbf{a^p} \in \mathbb{R}^m$
$\mathbf{t}$	user per-attribute privacy budget vector $t \in \mathbb{R}^{+m}$ ($\mathbb{R}^+$: positive real numbers)
ϵ	user overall privacy budget towards service provider

2.2 Differential Privacy

Differential privacy [9] is a recent privacy model which provides strong privacy guarantee. Informally, an algorithm $\mathcal{A}$ is differentially private if the output is insensitive to any particular record in the dataset.

Definition 1 (ϵ-Differential Privacy). *Let $\epsilon > 0$ be a small constant. A randomized function $\mathcal{A}$ is ϵ-differentially private if for all data sets D_1 and D_2 differing on at most one element, i.e., $d(D_1, D_2) = 1$, and all $\mathcal{S} \subseteq \mathsf{Range}(\mathcal{A})$,*

$$\mathsf{Pr}[\mathcal{A}(D_1) \in \mathcal{S}] \leq \exp(\epsilon)\mathsf{Pr}[\mathcal{A}(D_2) \in \mathcal{S}] \tag{1}$$

The probability is taken over the coin tosses of $\mathcal{A}$.

The parameter $\epsilon > 0$ is referred to as *privacy budget*, which allows us to control the level of privacy. A smaller ϵ suggests more limit posed on the influence of an individual item, which gives stronger privacy guarantee. Differential privacy enjoys the following important composition property:

Lemma 1 (Composition Property [8]). *If an algorithm $\mathcal{A}$ runs t randomized algorithms $\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_t$, each of which is t_i-differentially private, and applies an arbitrary randomized algorithm ϕ to their results ($\mathcal{A}(D) = \phi(\mathcal{A}_1(D), \ldots, \mathcal{A}_t(D))$), then $\mathcal{A}$ is $\sum_i t_i$-differentially private.*

One of the most widely used mechanisms to achieve ϵ-differential privacy is Laplace mechanism [9] (Theorem 1). Laplace mechanism adds random noises to the numeric output of a query, in which the magnitude of noises follows Laplace distribution with variance $\frac{\Delta f}{\epsilon}$ where Δf represents the global sensitivity of query f (Definition 2).

Definition 2 (Global Sensitivity [9]**).** *For a query $f : \mathcal{D} \rightarrow \mathbb{R}^k$, the global sensitivity Δf of f is as follows:*

$$\Delta f = \max_{d(D_1,D_2)=1} \|f(D_1) - f(D_2)\|_1 \tag{2}$$

for all D_1, D_2 differing in one element, i.e., $d(D_1, D_2) = 1$.

Theorem 1 (Laplace Mechanism [9]**).** *For $f : \mathcal{D} \rightarrow \mathbb{R}^k$, a randomized algorithm $\mathcal{A}_f = f(D) + \mathsf{Lap}^k(\frac{\Delta f}{\epsilon})$ is ϵ-differentially private.*

The Laplace distribution with parameter β, denoted $\mathsf{Lap}(\beta)$, has probability density function $\frac{1}{2\beta}\exp(-\frac{|z|}{\beta})$ and cumulative distribution function $\frac{1}{2}(1 + \mathsf{sgn}(z)(1 - \exp(-\frac{|z|}{\beta})))$.

2.3 MultiUPP Problem Definition

The goal of *Multi-Level User Privacy Perturbation (MultiUPP)* problem is to publish an accurate histogram that summaries the distribution of data attributes, which is sufficient to provide user high-quality services (e.g., personalized advertising, recommendation) in most cases [22]. In the meanwhile, MultiUPP also preserves both *overall privacy toward a specific service provider* and *different privacy needs for different data attributes. More importantly, our MultiUPP problem is considered as a general framework which can be coupled with different utility objectives.* Next, we specify the MultiUPP problem and its associated privacy and utility objectives respectively.

Formal Definition: Given a user's private item vector $\mathbf{d^r}$ associated with public universal item set I and a public item-attribute matrix $\mathbf{A}$; and this user's attribute-based privacy budget vector $\mathbf{t}$ as well as his overall privacy budget ϵ towards a service provider. MultiUPP outputs this user's perturbed attribute aggregates $\mathbf{a^p}$ to satisfy the following privacy and utility objectives:

Privacy Objectives of MultiUPP: We consider two privacy objectives aiming to defend against privacy leakage via public attribute information.

P1. Overall Differential Privacy Objective towards a service provider: satisfy *ϵ-differential privacy* on published histogram on all attributes with the presence or absence of an individual item in I. Each service provider is associated with an overall privacy budget ϵ based on its reputation, i.e., a smaller ϵ for a lower reputed service provider.

P2. Per-attribute Differential Privacy Objectives with Multiple Levels: satisfy *t_j-differential privacy* on published histogram on each attribute j with the presence or absence of an individual item in I. Each attribute j of data is associated with a privacy budget t_j based on each user's privacy concern on this attribute. For example, if a user considers location more private than price (attribute 1 and 2 of an item), this user will set $t_1 < t_2$.

Utility Objectives of MultiUPP: We consider publishing the histogram in which the number of bins equals to the number of attributes and the count in each bin j is perturbed summation of attribute values w.r.t. items in user's history. The published histogram is denoted as $\mathbf{a^p}$ as in Table 1. Following the convention in [25,27], we measure the accuracy (or utility) of a perturbed histogram in terms of the following two utility loss functions between raw and perturbed attribute aggregates $\mathbf{a^r}$ and $\mathbf{a^p}$ (denoted as $\mathcal{U}$):

U1. Expected Mean Absolute Error (MAE): $\mathcal{U}_{\text{MAE}} = \mathsf{E}\left[\frac{1}{m}\|\mathbf{a^p} - \mathbf{a^r}\|_1\right]$

U2. Expected Mean Square Error (MSE): $\mathcal{U}_{\text{MSE}} = \mathsf{E}\left[\frac{1}{m}\|\mathbf{a^p} - \mathbf{a^r}\|_2^2\right]$

In addition, we also consider the following third utility regarding per-attribute utility with multi-level privacy controls, for measuring the utility loss over the best utility on the aggregate of each attribute:

U3. Expected Mean Absolute Error Loss (MAEL): $\mathcal{U}_{\text{MAEL}} = \mathsf{E}\left[\frac{1}{m}\sum_{j=1}^{m} \frac{|a_j^p - a_j^r|}{BU_j}\right] - 1$, where BU_j stands for the best expected utility of attribute j. More specifically, $BU_j = \frac{\Delta f_j}{t_j}$ indicating the expectation of optimal Laplace noise $\mathsf{Lap}\left(\frac{\Delta f_j}{t_j}\right)$ for each query function $f_j : (\mathbb{Z}^+)^n \rightarrow \mathbb{R}$ [12].

Remarks: According to user study results in [24], what users most prefer is to control their different privacy concerns on limited number of relatively coarse-grained data attributes. Thus, we assume that the number of attributes m is bounded by a constant.

Challenges: (1) An item is usually associated with a number of attributes while each attribute has a different privacy concern level. How can we perturb the data to optimize the utility when satisfying all privacy guarantees? (2) When $\epsilon < \sum t_j$, the existing composition approach [8] is no longer feasible. In this case, what are the lower bounds of optimal utilities? What is the lower bound of ϵ with such optimal utilities? (3) When overall privacy budget ϵ is smaller than the above lower bound, how can we optimize the utility loss?

3 Lower Bounds

In this section, we focus on the queries $f_j : (\mathbb{Z}^+)^n \rightarrow \mathbb{R}$ in line with the aggregate (counting) of each attribute in MultiUPP problem definition. We first discuss the lower bound of overall privacy budget ϵ when optimal utilities are achieved, followed by the detailed lower bounds of the utility loss functions (optimal utilities) described in our problem.

3.1 Lower Bound of Overall Privacy Budget ϵ

We first understand the turning point when all utilities for each attribute aggregate are optimized while both overall and per-attribute privacy guarantees are satisfied. That is, we study a lower bound of ϵ on the public domain (item set/universe I), as shown in the following Theorem 2:

Theorem 2 (Lower Bound of ϵ). *For a set of queries $f_1, \ldots, f_m$ in which each $f_j : (\mathbb{Z}^+)^n \to \mathbb{R}$ is associated with its global sensitivity Δf_j and a privacy budget $t_j > 0$. If t_j-differential privacy is satisfied for each query with optimal utility, the overall privacy guarantee ϵ for all queries is lower bounded as follows:*

$$\epsilon \geq \max_{d(D_1,D_2)=1} \Big\{ \sum_{j=1}^{m} \frac{t_j}{\Delta f_j} |f_j(D_1) - f_j(D_2)| \Big\} \tag{3}$$

where $d(D_1, D_2) = 1$ stands for two neighboring datasets D_1, D_2.

Proof. According to the result by Hardt *et al.* [12], the optimal utility for an arbitrary query function f $(\mathbb{Z}^+)^n \to \mathbb{R}$ is $\Omega(\Delta f/\epsilon)$, which can be obtained by Laplace mechanism. Consider two arbitrary neighboring datasets D_1, D_2 ($d(D_1, D_2) = 1$) and any $\mathbf{s} = (s_1, \ldots, s_m) \in Range(\mathcal{A}^N)$ when every j^{th} element is obtained by adding noise $\mathsf{Lap}(\frac{\Delta f_j}{t_j})$ to aggregate of attribute j, in which $\mathcal{A}^N$ is the naive randomized algorithm where each $\mathcal{A}_j^N$ is $\mathsf{Lap}(\frac{\Delta f_j}{t_j})$):

$$\frac{\mathsf{Pr}[\mathcal{A}^N(D_1) = \mathbf{s}]}{\mathsf{Pr}[\mathcal{A}^N(D_2) = \mathbf{s}]} = \prod_{j=1}^{m} \frac{\mathsf{Pr}[\mathcal{A}_j^N(D_1)_j = s_j]}{\mathsf{Pr}[\mathcal{A}_j^N(D_2)_j = s_j]} = \prod_{j=1}^{m} \frac{\exp(-|f_j(D_1) - s_j|\frac{t_j}{\Delta f_j})}{\exp(-|f_j(D_2) - s_j|\frac{t_j}{\Delta f_j})}$$

$$\geq \prod_{j=1}^{m} \exp\Big(- \frac{t_j}{\Delta f_j} |f_j(D_1) - f_j(D_2)| \Big) = \exp\Big(\sum_{j=1}^{m} - \frac{t_j}{\Delta f_j} |f_j(D_1) - f_j(D_2)| \Big)$$

Therefore, proof is complete.

3.2 Lower Bounds of Utility Loss

We next study the lower bounds of optimal utility loss when the privacy objectives are satisfied. These lower bounds will also be used as baseline for experimental evaluation in Sect. 5.

Theorem 3 (Lower Bounds of Utility Loss). *If t_j-differential privacy is satisfied for each query and ϵ satisfies the lower bound in* (3)*, the lower bounds of utilities defined in Sect. 2.3 are as follows:*

$$\mathcal{U}_{MAE} \geq \frac{1}{m} \sum_{j=1}^{m} \frac{C_j}{t_j};\ \mathcal{U}_{MSE} \geq \frac{2}{m} \sum_{j=1}^{m} \frac{C_j^2}{t_j^2};\ \mathcal{U}_{MAEL} \geq 0$$

where $C_j = max_{1 \leq i \leq n}\{a_{ij}\}$.

Proof. As the optimal utility is obtained by Laplace mechanism in our case [12], we prove the above lower bounds based on the properties of Laplace distribution. Let X_j be the random variable following distribution $\mathsf{Lap}(\frac{\Delta f_j}{t_j})$, we have $\mathsf{E}[|X_j|] = \frac{\Delta f_j}{t_j}, \mathsf{Var}[X_j] = 2\Big(\frac{\Delta f_j}{t_j}\Big)^2$. Moreover, $\Delta f_j = C_j = max_{1 \le i \le n}\{a_{ij}\}$ based on the definition of global sensitivity.

$$\mathcal{U}_{MAE} = \mathsf{E}\Big[\frac{1}{m}\|\mathbf{a^p} - \mathbf{a^r}\|_1\Big] \ge \frac{1}{m}\sum_{j=1}^{m}\mathsf{E}[|X_j|] = \frac{1}{m}\sum_{j=1}^{m}\frac{C_j}{t_j}$$

$$\mathcal{U}_{MSE} = \mathsf{E}\Big[\frac{1}{m}\|\mathbf{a^p} - \mathbf{a^r}\|_2^2\Big] \ge \frac{1}{m}\sum_{j=1}^{m}\mathsf{Var}[X_j] = \frac{2}{m}\sum_{j=1}^{m}\frac{C_j^2}{t_j^2}$$

$$\mathcal{U}_{MAEL}(\mathbf{v}) = \mathsf{E}\Big[\frac{1}{m}\sum_{j=1}^{m}\frac{|a_j^p - a_j^r|}{BU_j}\Big] - 1 \ge \frac{1}{m}\sum_{j=1}^{m}\mathsf{E}\Big[\frac{|X_j|}{\frac{\Delta f_j}{t_j}}\Big] - 1 = 0$$

4 DP-MultiUPP Framework

In this section, we develop a novel *Differentially-Private Multi-Level User Privacy Perturbation (DP-MultiUPP)* framework to optimize the utility, especially when the condition (3) does not hold for ϵ. Specifically, we first introduce a novel differential privacy mechanism, called *Multi-Level Differential Privacy (MultiDP) Mechanism*, for trading off the utility loss and privacy guarantees. We then apply MultiDP mechanism to develop the DP-MultiUPP framework, with the provable privacy and utility guarantees and linear time complexity.

4.1 Multi-level Differential Privacy Mechanism

In this subsection, we focus on the case that ϵ is smaller than the lower bound in Theorem 2, i.e., the optimal utility cannot be achieved. In this case, we propose a novel mechanism, called *Multi-Level Differential Privacy (MultiDP) Mechanism*, to optimize the utility loss while preserving both per-attribute t_j-DP and overall ϵ-DP. In this mechanism, our goal is to find the condition for automating per-attribute privacy budgets t'_j (a reflection of the utility loss without violating per-attribute t_j-differential privacy) and overall ϵ-differential privacy guarantee.

As the determination of optimal privacy budgets t'_j is dependent on public domain, we consider the following MultiDP condition:

Definition 3 (MultiDP Condition). *For a set of queries $f_1, \ldots, f_m$ in which each $f_j : (\mathbb{Z}^+)^n \to \mathbb{R}$ is associated with its global sensitivity Δf_j. The set of non-negative numbers $t'_1, \ldots, t'_m$ satisfies MultiDP condition if the following two conditions hold:*

$$0 \le t'_j \le t_j, \forall 1 \le j \le m \tag{4}$$

$$\max_{d(D_1,D_2)=1}\Big\{\sum_{j=1}^{m}\frac{t'_j}{\Delta f_j}|f_j(D_1) - f_j(D_2)|\Big\} \le \epsilon \tag{5}$$

Algorithm 1. DP-MultiUPP Algorithm

```
Input : user private data $\mathbf{d^r}$, public item-attribute matrix $\mathbf{A}$, per-attribute
        privacy budgets $t_j$, overall privacy budget $\epsilon$
Output: perturbed attribute aggregates $\mathbf{a^P}$
$\mathbf{a^r} \leftarrow \mathbf{A}^T \mathbf{d^r}$;
Solve (6) with $\mathbf{v}^T \mathbf{v^r} \geq \mathbf{I}$ using [18];
$\mathbf{v} \leftarrow$ reciprocal of each entry in $\mathbf{v^r}$;
foreach $j = 1, \ldots, m$ do
    $a_j^p = a_j^r + \mathsf{Lap}(v_j)$;
return $\mathbf{a^P}$;
```

Theorem 4 (MultiDP Mechanism). *Given a set of non-negative numbers $t_1, \ldots, t_m$, and $t'_1, \ldots, t'_m$ satisfying MultiDP condition in Definition 3. For a set of queries $f_1, \ldots, f_m$ in which each $f_j : (\mathbb{Z}^+)^n \rightarrow \mathbb{R}$ is associated with its global sensitivity Δf_j, a randomized algorithm $\mathcal{A}^{MultiDP}$ that adds independently generated noise $\mathsf{Lap}\Big(\frac{\Delta f_j}{t'_j}\Big)$ to each query f_j enjoys t_j-differential privacy for each query f_j and overall ϵ-differential privacy for all queries $f_1, \ldots, f_m$.*

Proof. First, it is trivial to prove that $\mathcal{A}^{MultiDP}$ achieves t_j-differential privacy for each query since $t'_j \leq t_j$ always holds for each query j.

Next, we focus on the proof of overall differential privacy for all queries. Let D_1, D_2 be any two neighboring datasets, i.e., $d(D_1, D_2) = 1$. For any $\mathbf{s} = (s_1, \ldots, s_m) \in Range(\mathcal{A}^{MultiDP})$,

$$\frac{\mathsf{Pr}[\mathcal{A}^{MultiDP}(D_1) = \mathbf{s}]}{\mathsf{Pr}[\mathcal{A}^{MultiDP}(D_2) = \mathbf{s}]} \geq \prod_{j=1}^{m} \exp\Big(- \frac{t'_j}{\Delta f_j} |f_j(D_1) - f_j(D_2)| \Big) \geq \exp(-\epsilon)$$

The first step holds due to the independent Laplace noises on each attribute aggregate and triangle inequality; and the last step holds from the MultiDP condition in Definition 3.

The advantage of our proposed lower bound and multi-level mechanism, over the composition approach in [8], is that we take into account the correlation between queries. Therefore, our approach not only provides a much better ϵ lower bound but also helps to dramatically reduce the utility loss.

4.2 DP-MultiUPP Framework

Applying our proposed MultiDP mechanism, DP-MultiUPP framework aims to automate per-attribute privacy budgets $t'_1, \ldots, t'_m$ based on the overall privacy levels/budgets ϵ towards the service provider.

The rest of this subsection consists of notion definition, detailed DP-MultiUPP framework, and theoretical privacy, utility and time complexity analysis.

Notations. We define two notations:

(1) the noise standard deviation reciprocal vector $\mathbf{v^r} = (\frac{t'_1}{\Delta f_1}, \ldots, \frac{t'_m}{\Delta f_m})$, where the j^{th} entry is proportional to the reciprocal of standard deviation of injected Laplace noise on attribute j; and the noise standard deviation vector $\mathbf{v} = (\frac{\Delta f_1}{t'_1}, \ldots, \frac{\Delta f_m}{t'_m})$, where the j^{th} entry is the reciprocal of corresponding j^{th} entry in $\mathbf{v^r}$, i.e., proportional to the standard deviation of injected Laplace noises. The dimension of $\mathbf{v}, \mathbf{v^r}$ is given by the number of attributes m.

(2) the global sensitivity diagonal matrix $\mathbf{GS} = \mathit{diag}(\Delta f_1, \ldots, \Delta f_m)$, where the j^{th} entry is the global sensitivity of query f_j (aggregate of attribute j).

DP-MultiUPP Algorithm. The goal is to achieve optimal noise magnitude $\mathbf{v}$. To do so, we first formulate the mathematical programming as follows:

$$\begin{aligned} &\mathit{minimize}\ \ \mathcal{U}(\mathbf{v}) \\ &\mathit{subject\ to}\ \mathbf{A}\mathbf{v^r} \leq \epsilon \mathbf{1}_n, \mathbf{v^r} \leq \mathbf{GS}^{-1}\mathbf{t}, \mathbf{v^r} \geq \mathbf{0} \end{aligned} \tag{6}$$

where we optimize the utility function $\mathcal{U}$ defined in MultiUPP problem. Specifically, $\mathcal{U}$ takes noise standard deviation vector $\mathbf{v}$ as input, denoted as $\mathcal{U}(\mathbf{v})$. The three constraints imposes the MultiDP condition, which is sufficient to guarantee both t_j-differential privacy and ϵ-differential privacy as shown in MultiDP mechanism. As (6) is not convex in general with an implicit constraint $\mathbf{v}^T\mathbf{v^r} = \mathbf{I}$, we treat $\mathbf{v}, \mathbf{v^r}$ as two vector variables and add one more constraint $\mathbf{v}^T\mathbf{v^r} \geq \mathbf{I}$. The tweaked formulation has convex property. Algorithm 1 describes the DP-MultiUPP algorithm.

Formulation of Various Utilities for DP-MultiUPP Algorithm: Consider random variables $X_j \sim \mathsf{Lap}(\frac{\Delta f_j}{t'_j})$ on each attribute j. We specify utility functions $\mathcal{U}(\mathbf{v})$ for three utility objectives discussed in Sect. 2.3.

- $\mathcal{U}_{MAE}$: *Expected Mean Absolute Error.*

$$\mathcal{U}_{MAE}(\mathbf{v}) \propto \mathsf{E}\Big[\|\mathbf{a^p} - \mathbf{a^r}\|_1\Big] = \sum_{j=1}^{m} \mathsf{E}[|X_j|] = \sum_{j=1}^{m} v_j = \|\mathbf{v}\|_1$$

- $\mathcal{U}_{MSE}$: *Expected Mean Square Error.*

$$\mathcal{U}_{MSE}(\mathbf{v}) \propto \mathsf{E}\Big[\|\mathbf{a^p} - \mathbf{a^r}\|_2^2\Big] = \sum_{j=1}^{m} \mathsf{Var}[X_j] \propto \sum_{j=1}^{m} v_j^2 = \|\mathbf{v}\|_2^2$$

- $\mathcal{U}_{MAEL}$: *Expected Mean Absolute Error Loss.*

$$\mathcal{U}_{MAEL}(\mathbf{v}) \propto \mathsf{E}\Big[\sum_{j=1}^{m} \frac{|a_j^p - a_j^r|}{BU_j}\Big] - 1 \simeq \sum_{j=1}^{m} \frac{v_j}{BU_j} = \mathbf{BU^r}\mathbf{v}^T - 1$$

where $\mathbf{BU^r} = (\frac{t_1}{\Delta f_1}, \ldots, \frac{t_m}{\Delta f_m})$ stands for the reciprocal of standard deviation of injected noise with respect to each given privacy budget t_j. That is, $BU_j^r = \frac{1}{BU_j}$.

Theoretical Analysis. We provide privacy and utility analysis, as well as time complexity analysis.

Privacy analysis: *DP-MultiUPP framework enjoys t_j-differential privacy for each attribute aggregate and overall ϵ-differential privacy for all attribute aggregates.* The proof follows directly from the multi-level differential privacy mechanism proposed in Sect. 4.1.

Utility analysis: *DP-MultiUPP framework ensures the utilities upper bounded by the following:*

$$\mathcal{U}_{MAE} \leq \frac{1}{m}\sum_{j=1}^{m}\frac{C_j}{t'_j};\ \mathcal{U}_{MSE} \geq \frac{2}{m}\sum_{j=1}^{m}\frac{C_j^2}{t'^2_j};\ \mathcal{U}_{MAEL} \geq \frac{t_j}{t'_j} - 1$$

where $t'_j = t_j\epsilon / \max_{d(D_1,D_2)=1}\left\{\sum_{j=1}^{m}\frac{t_j}{\Delta f_j}|f_j(D_1) - f_j(D_2)|\right\}$ *when* ϵ *is smaller than lower bound in* (3). This is because the equal loss of each attribute leads to feasible solution regardless of the selected utility function. In the experiment, we treat this as baseline and show that the performance of our DP-MultiPP framework is much better in practice. In addition, when overall privacy budget ϵ is larger than lower bound in Theorem 2, DP-MultiUPP automatically achieves the lower bounds of utility losses in Theorem 3.

Time complexity analysis: *DP-MultiUPP framework has $O(n)$ time complexity.* This is exactly obtained from the analysis in [18] since the number of attributes is assumed to be bounded by a constant in this paper. Also, steps 3 and 4–5 both take $O(m)$ time.

5 Experimental Evaluation

In this section, we evaluate the performance of our proposed DP-MultiUPP framework. We conduct our experiments extensively on a variety of real-world datasets. We first use different metrics to measure the performance of the utility of all perturbed attribute aggregates as well as each attribute aggregate. Then, we report the scalability of DP-MultiUPP framework on both personal computer with 1.9 GHz CPU and 8 GB RAM, and Android Phone Galaxy S5.

5.1 Datasets, Settings, Metrics and Competitors

Datasets: We use three real world datasets.

MovieLens[1]*:* a movie rating dataset collected by the GroupLens Research Project at the University of Minnesota through the website movielens.umn.edu during the 7-month period from September 19th, 1997 through April 22nd, 1998. The number of attributes is 19. We use the MovieLens-1M, with 1,000,209 ratings from 6,040 users on 3,883 movies.

[1] http://grouplens.org/datasets/movielens.

Yelp[2]*:* a business rating data provided by RecSys Challenge 2013, in which Yelp reviews, businesses and users are collected at Phoenix, AZ metropolitan area. The number of attributes is 21. We use all reviews in training dataset, with 229,907 reviews from 43,873 users on 11,537 businesses.

MSNBC[3]*:* an anonymous web dataset collected by the UCI Machine Learning Repository through the msnbc.com domain during a 24-hour period on September 28, 1999. We consider types of websites as their attributes and the number of attributes is 17. We use the whole dataset, with 4,698,794 reviews from 989,818 users on these 17 attributes of websites.

Settings: We consider a fixed sum of per-attribute privacy budgets, i.e., $\sum t_j = 1$, and randomly select a privacy budget t_j for each attribute to satisfy this summation. We test different overall privacy budget ϵ from 0.05 to 0.4. We run each experiment 10 times and report the average result.

We test our proposed DP-MultiUPP framework by incorporating it with different utility functions in Sect. 2.3, denoted as DP-MultiUPP (MAE), DP-MultiUPP (MSE) and DP-MultiUPP (MAEL).

Metrics: We measure the performance of our DP-MultiUPP framework on utilities of both all attribute aggregates and each attribute aggregate, referred to as *Overall Utilities* and *Per-attribute Utilities.*

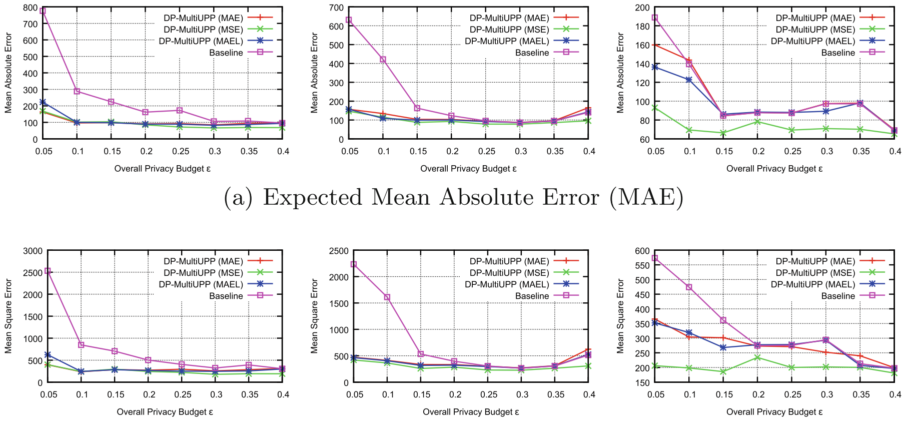
Overall Utilities. We use the expected Mean Absolute Error (MAE) and the expected Mean Square Error (MSE) in Sect. 2.3.

Per-attribute Utilities. We first use expected the Mean Absolute Error Loss (MAEL) in Sect. 2.3. In addition, we also consider another metric, KL-Divergence on injected per-attribute noise variance over optimal per-attribute noise variance, to measure the difference between the variance of injected Laplace noise using the optimized t'_j and that using a given t_j. Specifically, it can be written as $D_{KL} = \sum_{j=1}^{m} \frac{(\Delta f_j/t_j)^2}{\sum_j (\Delta f_j/t_j)^2} \log\left(\frac{\frac{(\Delta f_j/t_j)^2}{\sum_j (\Delta f_j/t_j)^2}}{\frac{(\Delta f_j/t'_j)^2}{\sum_j (\Delta f_j/t'_j)^2}}\right)$.

Competitors: We consider a baseline algorithm based on the state-of-the-art composition algorithm in Lemma 1 and our proposed lower bound of ϵ in Theorem 2. In detail, this baseline algorithm first scans all items and determines if the overall privacy budget ϵ is smaller than its lower bound given by per-attribute privacy budgets t_j. In this case, the utility obtained by this baseline approach is exactly the lower bound of utility loss in Sect. 3.2. If not, we simply inject $\mathsf{Lap}(\frac{\Delta f_j}{t_j})$ noises to the aggregate of each attribute j. Otherwise, we adjust each per-attribute privacy budget t_j to $t'_j = t_j/r$ where ratio $r = \max_{d(D_1,D_2)=1} \left\{ \sum_{j=1}^{m} \frac{t_j}{\Delta f_j} |f_j(D_1) - f_j(D_2)| \right\} / \epsilon$. Then, we inject $\mathsf{Lap}(\frac{\Delta f_j}{t'_j})$ into each attribute aggregate and it is not hard to see that this also satisfies overall ϵ-differential privacy.

[2] https://www.kaggle.com/c/yelp-recsys-2013/data.

[3] https://archive.ics.uci.edu/ml/datasets/MSNBC.com+Anonymous+Web+Data.



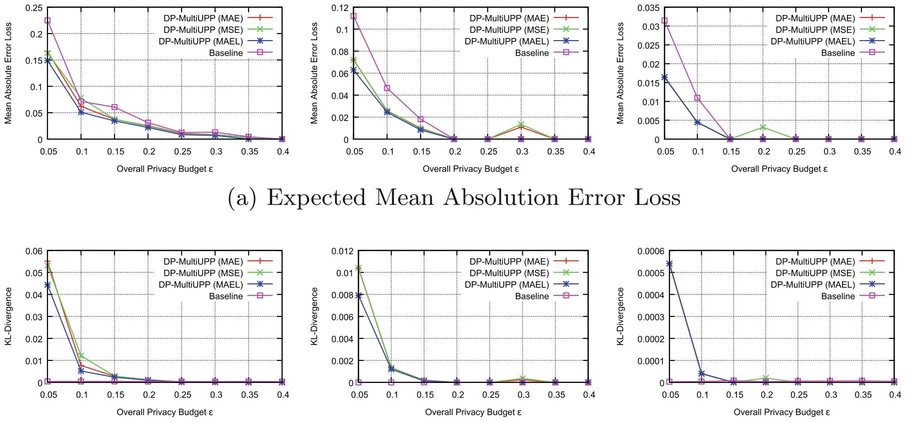
(a) Expected Mean Absolute Error (MAE)

(b) Expected Mean Square Error (MSE)

Fig. 2. Overall Utility Results (Left to Right: MovieLens, Yelp, MSNBC)

5.2 Utility Results

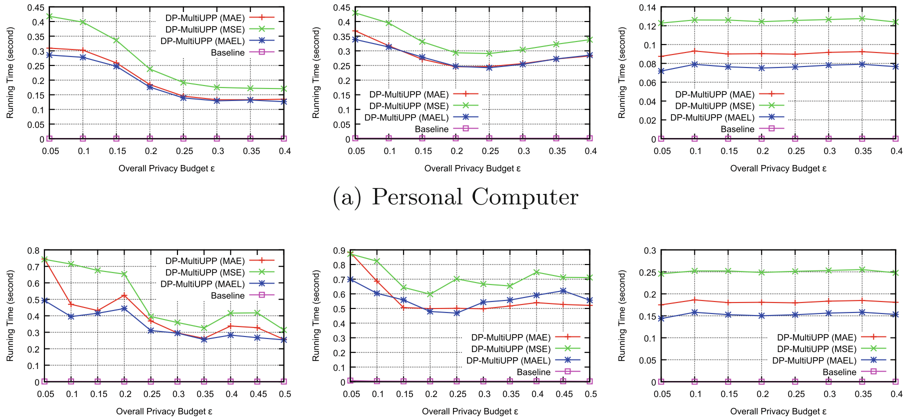
Overall Utility Results: Figure 2 reports the performance of DP-MultiUPP on overall utility results. As one can see, DP-MultiUPP consistently outperforms baseline algorithm regardless of its associated utility function. When ϵ is small ($\epsilon = 0.05$), DP-MultiUPP improves the performance up to 5 times out of the baseline approach. When ϵ is larger than the lower bound in Theorem 2, DP-MultiUPP continuously returns the optimal utility automatically due to its optimized utility objective.



(a) Expected Mean Absolution Error Loss

(b) KL-Divergence

Fig. 3. Per-attribute Utility Results (Left to Right: MovieLens, Yelp, MSNBC)



(a) Personal Computer

(b) Android Phone Galaxy S5

Fig. 4. Running Time of DP-MultiUPP (Left to Right: MovieLens, Yelp, MSNBC)

It is interesting to see that DP-MultiUPP with MSE utility function most of the time has best performance, especially in MSNBC dataset. This is because the variance of the injected noises can better capture all these utility losses. This provides us with an insight regarding how to select a better utility function.

More importantly, the smaller the overall privacy budget ϵ (w.r.t. lower reputed services) is, the bigger advantage DP-MultiUPP has over the baseline algorithm. This makes DP-MultiUPP very practically useful since users need stronger privacy guarantee especially for numerous low reputed service providers.

Per-attribute Utility Results: Figure 3 reports the performance of DP-MultiUPP on per-attribute utility results. Figure 3(a) shows DP-MultiUPP (MAEL) again improves the utility up to twice than using the baseline algorithm. As one can see in Fig. 3(b), the KL-Divergence on injected per-attribute noise variance over optimal per-attribute noise variance remains small in all datasets. This is because the optimization of (6) evenly increases privacy levels for each attribute while preserving the overall privacy level. Thus, user's preferred privacy levels for each attribute are very well maintained.

Scalability: Figure 4 reports the averaged running time of all algorithms on different datasets on both personal computer and Android Phone Galaxy S5. As one can see, our DP-MultiUPP framework takes at most 0.5 s and 1 s on PC and Android smartphone respectively and the running time almost remains invariant with different overall privacy budgets. Overall, thanks to the linear time complexity, DP-MultiUPP is very scalable on different client devices.

5.3 Case Study: Personalized Recommendation

We conduct an additional case study of personalized recommendation using perturbed data obtained by our approach on MovieLens dataset, through collaborative filtering (SGD algorithm) in GraphLab[4]. In this case, we first sanitize perturbed data $\mathbf{d}^{\mathbf{p}}$ based on the perturbed attribute aggregates using the following mathematical programming: $\min \frac{1}{2}\|\mathbf{A}^T\mathbf{d}^{\mathbf{p}} - \mathbf{a}^{\mathbf{p}}\|^2$ s.t. $\mathbf{d}^{\mathbf{p}} \in \{0,1\}^n$. Using $\epsilon = 0.1$, the MAE loss between the recommendation results using user private/raw and perturbed data against ground truth is shown only up to 8 %.

6 Related Work

Privacy Protection under Untrusted Server Settings: A traditional class of approaches preserve privacy based on cryptography under untrusted server setting [2,5,19]. Another orthogonal class of privacy protection approaches is based on injecting noises. Polat *et al.* [20] developed randomized mechanisms to perturb the data before releasing to untrusted service providers. However, their method does not have provable privacy guarantees and was later identified to suffer from inference attacks. A recent work by Shen *et al.* [23] introduced a differential private data perturbation method on user's client. Although this approach has formal privacy and utility guarantee, it can only take one privacy budget and treat every type of data with the same privacy concern.

Differential Privacy: Differential privacy [7,9] has become the de facto standard for privacy preserving data analytics. Dwork *et al.* [9] established the guideline to guarantee differential privacy for individual aggregate queries by calibrating the Laplace noise to each query regarding the global sensitivity. Various works have adopted this definition for publishing histograms [25], search logs [14], mining data streams [6], and record linkage [4]. Later on, a noise mitigation mechanism was proposed by Machanavajjhala *et al.* [17].

Histogram Release via Differential Privacy: The most basic approach is to add noises of full contingency table of the whole dataset that suffers from exponential computational and space complexity. An improvement of this basic approach was proposed by Dwork *et al.* [9] to add independently generated Laplace noise to each k-way marginal table. Later on, Barak *et al.* [3] proposed the approach to add noises in the Fourier domain and improve the expected squared error by 2^k. Li *et al.* [16] proposed the matrix mechanism for counting queries. However, it still suffers from high computational complexity. In addition to these approaches, there exist many other approaches such as [10,11,21]. Unfortunately, none of these approaches provides an option for multi-level privacy concern configuration.

Multi-level Differential Privacy Preservation: The state-of-the-art method is the composition approach in [9] which preserves both per-attribute and overall differential privacy. However, it does not analyze when the achievement of all privacy guarantees is feasible, and does not provide a utility optimization mechanism when it is infeasible to achieve all privacy guarantees.

[4] http://select.cs.cmu.edu/code/graphlab/pmf.html.

7 Conclusion and Future Work

In this paper, we develop the first lightweight framework via differential privacy to automate multi-level privacy controls for releasing different attributes of data to service providers of different reputations. We theoretically analyze privacy, utility and time complexity. The experimental results show that our approach outperforms state-of-the-art approach up to 5 times with high scalability on both personal computer and smartphone. Particularly, our framework shows significant advantage for stronger privacy guarantee towards numerous low reputed service providers, making it very practically useful.

In the future work, we intend to extend our approach into more practical scenarios: (1) we will conduct more thorough experiments on personalized recommendation case study; (2) when the correlation among user private data attributes and the correlation among public attributes are similar, we will define a new privacy notion and mechanism to tackle the decreased privacy guarantees; (3) we will design a streaming mulit-level privacy preserving data publishing approach to tackle continuously generated user private data.

References

1. Pew research report. http://www.pewinternet.org/2013/09/05/anonymity-privacy-and-security-online-2/
2. Armknecht, F., Strufe, T.: An efficient distributed privacy-preserving recommendation system. In: Ad Hoc Networking Workshop, pp. 65–70, June 2011
3. Barak, B., Chaudhuri, K., Dwork, C., Kale, S., McSherry, F., Talwar, K.: Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In: PODS, pp. 273–282. New York, NY, USA (2007)
4. Bonomi, L., Xiong, L., Lu, J.J.: LinkIT: privacy preserving record linkage and integration via transformations. In: SIGMOD, pp. 1029–1032 (2013)
5. Canny, J.: Collaborative filtering with privacy. In: IEEE Symposium on Security and Privacy, pp. 45–57 (2002)
6. Chan, T.-H.H., Li, M., Shi, E., Xu, W.: Differentially private continual monitoring of heavy hitters from distributed streams. In: Cristofaro, E., Murdoch, S.J. (eds.) PETS 2014. LNCS, vol. 8555, pp. 140–159. Springer, Heidelberg (2012). doi:10.1007/978-3-642-31680-7_8
7. Dwork, C.: Differential privacy: a survey of results. In: Jain, R., Jain, S., Stephan, F. (eds.) TAMC 2015. LNCS, vol. 9076, pp. 1–19. Springer, Heidelberg (2008). doi:10.1007/978-3-540-79228-4_1
8. Dwork, C., Lei, J.: Differential privacy and robust statistics. In: STOC, pp. 371–380. ACM, New York (2009)
9. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Kushilevitz, E., Malkin, T. (eds.) TCC 2016. LNCS, vol. 9563, pp. 265–284. Springer, Heidelberg (2006). doi:10.1007/11681878_14
10. Gupta, A., Hardt, M., Roth, A., Ullman, J.: Privately releasing conjunctions and the statistical query barrier. In: STOC, pp. 803–812. New York, NY, USA (2011)
11. Hardt, M., Ligett, K., Mcsherry, F.: A simple and practical algorithm for differentially private data release. In: NIPS, pp. 2339–2347 (2012)

12. Hardt, M., Talwar, K.: On the geometry of differential privacy. In: STOC, pp. 705–714. ACM, New York (2010)
13. Jeckmans, A.J.P., Beye, M.R.T., Erkin, Z., Hartel, P.H., Lagendijk, R.L., Tang, Q.: Privacy in recommender systems. In: Ramzan, N., van Zwol, R., Lee, J.-S., Clüver, K., Hua, X.-S. (eds.) Social Media Retrieval. Computer Communications and Networks, pp. 263–281. Springer, London (2013)
14. Korolova, A., Kenthapadi, K., Mishra, N., Ntoulas, A.: Releasing search queries and clicks privately. In: WWW, pp. 171–180 (2009)
15. Leon, P.G., Ur, B., Wang, Y., Sleeper, M., Balebako, R., Shay, R., Bauer, L., Christodorescu, M., Cranor, L.F.: What matters to users?: factors that affect users' willingness to share information with online advertisers. In: SOUPS, pp. 7:1–7:12. ACM, New York (2013)
16. Li, C., Hay, M., Rastogi, V., Miklau, G., McGregor, A.: Optimizing linear counting queries under differential privacy. In: PODS, pp. 123–134 (2010)
17. Machanavajjhala, A., Kifer, D., Abowd, J., Gehrke, J., Vilhuber, L.: Privacy: theory meets practice on the map. In: ICDE, pp. 277–286 (2008)
18. Megiddo, N.: Linear programming in linear time when the dimension is fixed. J. ACM **31**(1), 114–127 (1984). http://doi.acm.org/10.1145/2422.322418
19. Nikolaenko, V., Ioannidis, S., Weinsberg, U., Joye, M., Taft, N., Boneh, D.: Privacy-preserving matrix factorization. In: CCS, pp. 801–812. New York (2013)
20. Polat, H., Du, W.: Privacy-preserving collaborative filtering using randomized perturbation techniques. In: ICDM, pp. 625–628 (2003)
21. Qardaji, W., Yang, W., Li, N.: PriView: practical differentially private release of marginal contingency tables. In: SIGMOD, pp. 1435–1446. New York (2014)
22. Rajaraman, A., Ullman, J.D.: Mining of Massive Datasets. Cambridge University Press, New York (2011)
23. Shen, Y., Jin, H.: Privacy-preserving personalized recommendation: an instance-based approach via differential privacy. In: ICDM, pp. 540–549 (2014)
24. Tsai, J.Y., Egelman, S., Cranor, L., Acquisti, A.: The effect of online privacy information on purchasing behavior: an experimental study. Inf. Syst. Res. **22**(2), 254–268 (2011)
25. Xu, J., Zhang, Z., Xiao, X., Yang, Y., Yu, G.: Differentially private histogram publication. In: ICDE, pp. 32–43 (2012)
26. Zhang, B., Wang, N., Jin, H.: Privacy concerns in online recommender systems: influences of control and user data input. In: SOUPS, pp. 159–173 (2014)
27. Zhang, X., Chen, R., Xu, J., Meng, X., Xie, Y.: Towards accurate histogram publication under differential privacy. In: SDM, pp. 587–595 (2014)