An Efficient Two-Layer Mechanism for Privacy-Preserving Truth Discovery*

Yaliang Li [1], Chenglin Miao [2], Lu Su [2], Jing Gao [2], Qi Li [3], Bolin Ding [4], Zhan Qin [5], and Kui Ren [2]
[1] Tencent Medical AI Lab, Palo Alto, CA USA
[2] State University of New York at Buffalo, Buffalo, NY USA
[3] University of Illinois at Urbana Champaign, Urbana, IL USA
[4] Alibaba Group, Bellevue, WA USA
[5] University of Texas at San Antonio, San Antonio, TX USA
[1] yaliangli@tencent.com, [2] {cmiao, lusu, jing, kuiren}@buffalo.edu
[3] qili5@illinois.edu, [4] bolin.ding@alibaba-inc.com, [5] zhan.qin@utsa.edu

ABSTRACT

Soliciting answers from online users is an efficient and effective solution to many challenging tasks. Due to the variety in the quality of users, it is important to infer their ability to provide correct answers during aggregation. Therefore, truth discovery methods can be used to automatically capture the user quality and aggregate user-contributed answers via a weighted combination. Despite the fact that truth discovery is an effective tool for answer aggregation, existing work falls short of the protection towards the privacy of participating users. To fill this gap, we propose perturbation-based mechanisms that provide users with privacy guarantees and maintain the accuracy of aggregated answers. We first present a one-layer mechanism, in which all the users adopt the same probability to perturb their answers. Aggregation is then conducted on perturbed answers but the aggregation accuracy could drop accordingly. To improve the utility, a two-layer mechanism is proposed where users are allowed to sample their own probabilities from a hyper distribution. We theoretically compare the one-layer and two-layer mechanisms, and prove that they provide the same privacy guarantee while the two-layer mechanism delivers better utility. This advantage is brought by the fact that the two-layer mechanism can utilize the estimated user quality information from truth discovery to reduce the accuracy loss caused by perturbation, which is confirmed by experimental results on real-world datasets. Experimental results also demonstrate the effectiveness of the proposed two-layer mechanism in privacy protection with tolerable accuracy loss in aggregation.

CCS CONCEPTS

• **Information systems** → **Data mining**; • **Security and privacy** → **Privacy protections**;

KEYWORDS

Truth discovery; differential privacy; two-layer mechanism

ACM Reference Format:
Yaliang Li, Chenglin Miao, Lu Su, Jing Gao, Qi Li, Bolin Ding, Zhan Qin, and Kui Ren. 2018. An Efficient Two-Layer Mechanism for Privacy-Preserving Truth Discovery. In *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3219819.3219998

1 INTRODUCTION

Nowadays, crowdsourcing gains an increasing popularity as it can be adopted to solve many challenging question answering tasks. For example, crowds of users can help search engines to answer the question on whether a website is relevant to a search query [36]; patients who are taking new drugs can answer the question on whether a specific drug has a certain side-effect [25]; and students involved in massive open online courses can help instructors answer the question on which grade the other students should earn during peer grading [2]. In these and many more applications, crowds of users can contribute their efforts to answer questions of interest, which largely reduces the financial cost and benefits various application domains.

However, the information quality of the crowdsourced answers varies significantly among different users. Some users may have sufficient domain knowledge and thus can provide accurate and meaningful answers, while others may submit biased or wrong answers. The situation becomes even worse when some users disperse deceptive answers driven by the financial incentives. The diversity of users motivates an important task for crowdsourced question answering: how to aggregate the noisy candidate answers from crowds of users to infer accurate answers?

A straightforward approach to aggregate the crowdsourced answers is to conduct majority voting in which the answer that has the highest number of occurrences will be selected as the final answer. Unfortunately, this naive approach assumes that all the users are equally reliable, so it cannot distinguish high-quality users from low-quality ones.

In order to provide more accurate answers, it is necessary to capture the variety of user quality and incorporate such quality information into aggregation. However, prior knowledge of user quality is not available, and it is a challenge to estimate user quality without groundtruth information. To tackle this challenge, truth

*This work was done when the first author was at State University of New York at Buffalo, and the first two authors contributed equally to this work.

KDD '18, August 19–23, 2018, London, United Kingdom

ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3219998

discovery [8, 19, 20, 32, 35] emerges as a hot topic due to its ability to estimate user quality without any supervision. As both accurate answers and user quality are unknown, truth discovery approaches adopt the following two principles: If a candidate answer is supported by many high-quality users, it is more likely to be the true answer; Meanwhile, if a user provides many accurate answers, he will be assigned a high weight. These two principles rely on each other, and they are tightly coupled together. Based on these principles, various truth discovery methods have been proposed for different scenarios, and the existing work [18, 22] has demonstrated the advantages of adopting truth discovery to aggregate crowdsourced answers.

Privacy Concerns. However, one missing part in the aforementioned existing work is the protection of user privacy: individual users may have privacy concern when sharing their own answers with others. For example, individual users can report the relevance between a search query and a webpage, but the answers may leak their personal preferences. Patients' reactions to drugs are valuable for physicians to discover drugs' side-effects, but contain sensitive information that patients might not want to share. Peer grading serves as a useful tool for massive open online courses, but individual students' judgment towards others' work should be protected. Without a convincing privacy-preserving mechanism, users may not be willing to contribute their sensitive information for question answering tasks, or even worse, they may provide untruthful information to protect their privacy, which degrades the performance of crowdsourced question answering.

One possible solution to tackle this challenge is to adopt encryption or secure multi-party computation techniques to protect the privacy of users [16, 23, 24, 34]. Unfortunately, these techniques require expensive computation resources and intensive communications among users. Therefore, due to the large scale of users in most crowdsourcing applications, encryption or secure multi-party computation techniques are not suitable for privacy-preserving crowdsourced question answering.

Privacy-Preserving Mechanisms. In the light of this challenge, we start by presenting a simple yet efficient mechanism to protect the user privacy using perturbation technique. This mechanism allows users to randomly perturb their candidate answers with a pre-defined probability, and then the perturbed answers are submitted for weighted aggregation. To provide users with a strong privacy guarantee, it is required to set the pre-defined perturbation probability to be a large value, and thus the accuracy of the aggregated answers may not be satisfactory.

In order to guarantee both aggregation accuracy and user privacy, we propose another privacy-preserving mechanism. In this mechanism, users are allowed to independently sample their private probabilities from a hyper distribution, and then perturb their candidate answers according to the sampled probabilities. As the perturbation is controlled by two layers of distributions, we refer to this proposed mechanism as *two-layer mechanism*, and accordingly the aforementioned simple mechanism as *one-layer mechanism*.

The major difference between these two privacy-preserving mechanisms is that the one-layer mechanism forces all the users to adopt the same probability to perturb their candidate answers, while the two-layer mechanism allows users to sample their own probabilities. Recall that estimating user quality is a key factor in truth discovery. Thus the personalized noise introduced by the two-layer mechanism can be largely reduced by the user quality estimation component in truth discovery. In this way, the proposed two-layer mechanism can wisely take advantage of truth discovery, and they are coupled together to ensure that the aggregation accuracy only drops slightly even when strong privacy is guaranteed.

To theoretically compare the one-layer and two-layer mechanisms, we first quantify the user privacy based on local differential privacy definition [9, 11, 15], and prove that these two mechanisms can provide the same level of strong privacy guarantee. We then compute the accuracy loss under the privacy-preserving mechanisms, and show that the two-layer mechanism can give better utility (less accuracy loss) than the one-layer mechanism when they provide users with the same level of privacy protection.

We further confirm the theoretical analysis by conducting experiments on two real-world crowdsourced datasets. The experimental results demonstrate that with the same privacy guarantee, the proposed two-layer mechanism delivers better utility than the one-layer mechanism due to the fact that the proposed two-layer mechanism can fully utilize the benefits of user quality estimation in truth discovery. We also demonstrate that the proposed two-layer mechanism is a general framework and performs well under various scenarios.

Contributions. To summarize, the following contributions are made in this paper:

- Motivated by the strong need to protect user privacy, we propose a two-layer mechanism that is tightly combined with truth discovery. The proposed privacy-preserving method can make users feel comfortable to share their sensitive answers with others and thus enables more real-world applications.
- After formally defining the privacy and utility, we theoretically compare the two privacy-preserving mechanisms, and prove that the two-layer mechanism can give better utility compared to the one-layer mechanism when they provide the same level of privacy guarantee.
- Experiments on two real-world crowdsourced datasets are conducted to confirm the theoretical analysis, and the results clearly demonstrate the advantages of the proposed two-layer mechanism.

The rest of the paper is organized as follows. We review the related work in Section 2 and then formally define our task in Section 3. Preliminaries on crowdsourced question answering are introduced in Section 4. We present the proposed privacy-preserving mechanisms in Section 5 and then theoretically analyze them in Section 6. The experiments on real-world datasets are summarized in Section 7. We conclude the paper in Section 8.

2 RELATED WORK

Crowdsourced Data Aggregation. Recent years have witnessed the growing popularity of crowdsourcing in question answering [2, 25, 36], and thus many efforts have been attracted and contributed. One important component of crowdsourced question answering is to wisely aggregate the candidate answers from users. As the

quality of users varies significantly, existing work develops various methods to capture the user quality [7, 28, 36]. Among them, truth discovery [8, 13, 19–21, 32, 35] is a category of algorithms that can automatically estimate user weights from the data, and incorporate such weights into aggregation to derive more accurate answers. However, none of the above work considers the privacy concern of users. Recently some mechanisms [16, 23, 24, 34] are proposed to protect user privacy in truth discovery based on encryption or secure multi-party computation techniques. Compared to them, the proposed mechanism in this paper is the first perturbation-based solution to protect user privacy when applying truth discovery to aggregate answers from crowds of users, which is much more efficient.

Privacy-Preserving Data Aggregation. Nowadays, the problem of privacy-preserving data aggregation has been widely studied, and various techniques can be applied: (1) *Randomized response* [1, 5, 31] is a survey technique that can provide privacy protection for individual users. When a user participates in a survey, his true answers can be protected via the added randomness in the data collection process. Compared to randomized response, the proposed two-layer mechanism provides a personalized flipping probability and enables better privacy-utility trade-off. The difference between the proposed two-layer mechanism and randomized response work is discussed in detail in Section 5.2. (2) *Differential privacy mechanisms* [10, 11, 17] have also been applied to protect the sensitive information from users when publishing statistical information about the data. (3) Further, *encryption* [3, 26] and *secure multi-party computation* [27] techniques provide secure protocols that enable the computation on sensitive data while the privacy can be guaranteed.

The existing work of privacy-preserving data aggregation [4, 6, 33] focuses on tasks such as the computation of statistics [12, 27], or user location privacy protection [14, 29]. However, these work treats all the users equally, and their tasks are different from ours. As mentioned before, an important component of truth discovery is to estimate user weights from the data and conduct weighted aggregation. Thus these privacy-preserving data aggregation methods cannot be easily applied to privacy-preserving crowdsourced question answering in which the unique characteristics of user weights should be taken into account.

3 TASK DEFINITION

We start by formally defining the task. Conceptually, two parties, server and user, are involved in the crowdsourced question answering. The server, who conducts data collection, is interested in a set of questions where each of them is associated with a finite number of possible answers. The users, who represent the individual participants, provide their own answers to these questions. After collecting the candidate answers from users, the server aggregates these candidate answers to derive final answers.

The main privacy concern of users is that the submitted answers may contain their sensitive information, and thus users are not willing to leak these answers to any other parties. This prevents users from sharing their own answers with the server. The server, who is assumed to be untrusted, may try to infer additional knowledge about users from their submitted answers. This unfaithful behavior of server can be driven by financial incentives or other benefits.

Motivated by the strong need to provide users with privacy protection, we aim to design privacy-preserving mechanisms for crowdsourced question answering. The developed mechanisms may enable more people to share their data, which will further unleash the power of crowdsourcing in question answering.

Formally speaking, our task can be formulated as follows:

Definition 3.1. Suppose there is a set of questions $\mathcal{Q}$, and the candidate answers (categorical data) are collected from a set of users $\mathcal{U}$. Let x_q^u represent the candidate answer of user u for the q-th question. The goal of our task is to get accurate final answers by jointly conducting user weight estimation and weighted aggregation on the user-provided answers. Meanwhile, the users' privacy should be protected so that the probability of inferring users' true answers based on the user-provided answers is low.

4 PRELIMINARY: TRUTH DISCOVERY

Crowdsourcing provides an efficient way to answer questions of interest by utilizing the wisdom of crowds. The questions will be distributed to multiple users, and multiple candidate answers can be collected for each question. Thus an important component of crowdsourced question answering is how to aggregate multiple answers for each question to get an accurate final answer. Formally speaking, for each question $q \in \mathcal{Q}$, we collect a set of candidate answers $\{x_q^u\}_{u \in \mathcal{U}}$ from the user set $\mathcal{U}$, and the goal is to aggregate these candidate answers to derive an accurate final answer x_q^*.

Majority Voting. A straightforward way is to conduct majority voting, that is, the candidate answer that has the most occurrences among all possible answers will be chosen as the aggregated result. Mathematically, the aggregated answer x_q^* is calculated as follows:

$$x_q^* = \arg\max_{x \in \mathcal{X}} \sum_{u \in \mathcal{U}} \mathbb{1}(x, x_q^u), \tag{1}$$

where $\mathcal{X}$ is the set of all possible answers, and $\mathbb{1}(\cdot, \cdot)$ is an indicator function.

The main drawback of this aggregation strategy is that it treats all the users equally. In practice, the information quality varies significantly among different users. The aggregated answers can be greatly improved by distinguishing high-quality users from the others and relying on these identified high-quality users.

Truth Discovery. However, the challenge is that the user quality is usually unknown *a priori* in practice. To tackle this challenge, truth discovery [8, 19, 20, 32] emerges as a hot topic due to its ability to automatically estimate user quality from data in the form of user weights. Truth discovery has been successfully applied in crowdsourced question answering, and the existing work [18, 22] has demonstrated the advantages of truth discovery on this task.

Although different truth discovery methods have been proposed to deal with various scenarios, they follow the same general principle: the candidate answers from high-quality users will be counted more in the aggregation, and the users who provide accurate answers more often will be assigned higher weights. Following this principle, the process of answer aggregation and weight estimation are tightly coupled. Truth discovery methods start with a uniform

initialization of user weights, and then iteratively conduct the following steps until convergence:

- Answer Aggregation: In this step, the user weights w_u are assumed to be known. The aggregated answer for each question x_q^* is calculated based on the following weighted voting:

$$x_q^* = \arg\max_{x \in \mathcal{X}} \sum_{u \in \mathcal{U}} w_u \cdot \mathbb{1}(x, x_q^u). \quad (2)$$

- Weight Estimation: In this step, the aggregated answers are fixed. For each user, his weight is estimated based on the accuracy of his provided answers, comparing with the current aggregated answers. That is,

$$w_u = g(p_u) = g(\frac{\sum_{q \in \mathcal{Q}} \mathbb{1}(x_q^*, x_q^u)}{|\mathcal{Q}|}), \quad (3)$$

where p_u is the probability that user u provides correct answers, and $g(\cdot)$ is a monotonically increasing function. This user weight estimation formula follows the idea that if a user provides correct answers more often, he will be assigned a higher weight.

5 PRIVACY-PRESERVING MECHANISMS

Compared to the simple majority voting, truth discovery methods estimate user weights and incorporate such weights into aggregation, and thus the final answers are more accurate. However, the existing work on crowdsourced answer aggregation either fails to consider the user privacy issue or introduces expensive computational cost. Users' contributions are valuable for the question answering tasks, but the candidate answers from users may contain their sensitive information, and thus users have privacy concern to share such personal information with the server. Motivated by the strong need to protect user privacy, we present two privacy-preserving truth discovery mechanisms for crowdsourced question answering. The goals of these mechanisms are two-fold: providing users with privacy guarantees and achieving accurate final answers.

5.1 One-Layer Mechanism

To protect the privacy of users, the one-layer mechanism adopts perturbation technique. More specifically, users can perturb their original answers to other possible ones, and then submit the perturbed answers to the server. As the server does not know original answers of users, the privacy of individual users can be guaranteed.

Mathematically, the perturbation method can be defined as:

Definition 5.1. The answer perturbation method $\mathcal{M}$ is a function with domain $\mathcal{X}$, and its range is the same with the domain. Let p^f be the probability to perturb the original answer x. $\forall x, \hat{x} \in \mathcal{X}$, $\mathcal{M}(x) = \hat{x}$ with probability $1 - p^f$ if $x = \hat{x}$, and with probability $\frac{p^f}{s-1}$ if $x \neq \hat{x}$, where s is the size of the range.

The one-layer mechanism uses the above answer perturbation method to allow users to perturb their answers with the same pre-defined probability p^f. After receiving perturbed answers from users, the server aggregates these answers by applying truth discovery. The general flow of one-layer mechanism is summarized in Algorithm 1. In addition, we illustrate the concrete procedure by Example 1.

Algorithm 1 One-layer Mechanism for Privacy-Preserving Crowdsourced Question Answering

Input: Question set $\mathcal{Q}$, and the set of users $\mathcal{U}$
Output: Aggregated answers $\{\hat{x}_q^*\}_{q \in \mathcal{Q}}$

1: Server distributes the question set to each user;
2: Users prepare his/her candidate answers;
3: Users perturb their candidate answers $\{x_q^u\}$ according to the perturbation method (Definition 5.1) with the pre-defined p^f, and submit the perturbed answers $\{\hat{x}_q^u\}$ to the server;
4: Server applies truth discovery to get aggregated answers $\{\hat{x}_q^*\}_{q \in \mathcal{Q}}$.
5: **return** Aggregated answers $\{\hat{x}_q^*\}_{q \in \mathcal{Q}}$.

Example 1: Consider a question with two possible answers $\{Y, N\}$, i.e., $s = 2$. A particular user's answer to this question is Y. Following the one-layer mechanism, this user will flip his answer with pre-defined probability $p^f = 0.4$ before submitting his answer to the server. Let's assume that the perturbed answer of this particular user to the considered question is N, and then this user will submit the perturbed answer N to the server. However, from the perspective of the server, it can only see the submitted answer from this user (i.e., N), and it does not know his original answer (i.e., Y). Thus the privacy of this user is protected.

5.2 Two-Layer Mechanism

The above one-layer mechanism provides users with privacy protection. However, in order to provide a strong privacy guarantee, the pre-defined perturbation probability p^f needs to be set as a large value. In this case, all the users perturb their answers with the same large probability. Thus the accuracy of the aggregated answers can decrease dramatically and the utility may not be satisfied. This motivates us to improve the one-layer mechanism so that guarantees of both privacy and utility can be provided.

In truth discovery, user weights are automatically estimated, and such weights are incorporated into the aggregation. To improve the utility of the privacy-persevering mechanism, we fully utilize this unique property of truth discovery, and propose a two-layer mechanism (Algorithm 2). We also provide an example (i.e., Example 2) to further illustrate the two-layer mechanism.

Example 2: Consider the scenario in Example 1. Following the two-layer mechanism, that particular user will also perturb his answer before submitting to the server. The difference here is that the user needs to sample his own flipping probability p_u^f from a hyper distribution, instead of using the pre-defined flipping probability p^f for all the users.

Benefit of The Two-Layer Mechanism. Compared to the one-layer mechanism, the main difference is that in the two-layer mechanism, each user samples his own private probability to perturb his answers. This novel design of two-layer perturbation fully explores the property of truth discovery, which makes it possible to achieve high accuracy even when the added noise is large. Truth discovery is a weighted aggregation in which the weight of each user is dynamically adjusted based on their information quality. In this

Algorithm 2 Two-layer Mechanism for Privacy-Preserving Crowdsourced Question Answering

Input: Question set $\mathcal{Q}$, and the set of users $\mathcal{U}$
Output: Aggregated answers $\{\hat{x}_q^*\}_{q\in\mathcal{Q}}$

1: Server distributes the question set to each user;
2: Users prepare his/her candidate answers;
3: Each user samples a private probability p_u^f from a pre-defined hyper distribution f;
4: Users perturb their candidate answers $\{x_q^u\}$ according to the perturbation method (Definition 5.1) with their own probabilities p_u^f's, and submit the perturbed answers $\{\hat{x}_q^u\}$ to the server;
5: Server applies truth discovery to get aggregated answers $\{\hat{x}_q^*\}_{q\in\mathcal{Q}}$.
6: **return** Aggregated answers $\{\hat{x}_q^*\}_{q\in\mathcal{Q}}$.

way, the effect of added noise can be absorbed in the weight and thus will not affect the final aggregation much. In the following section, rigorous analysis and proof show how accurate aggregation results are obtained by the proposed two-layer mechanism even if significant noise is added to the data.

Comparison with Randomized Response. Randomized response is a well-known privacy protection mechanism which also allows users to randomly flip their answers to sensitive survey questions. However, the proposed two-layer mechanism differs from randomized response in terms of the general goal: Randomized response is adopted to compute some statistics of the data [1, 5, 31], while our goal is to find the correct answers by jointly conducting user weight estimation and weighted aggregation. Among the related work about randomized response, the FRAPP framework [1] also provides a way to randomize the perturbation probabilities. However, in FRAPP framework, the utility is degraded by such randomization, as the aggregation component does not consider the diversity of quality among users. While in the proposed two-layer mechanism, the utility can be improved by the personalized flipping probability, and such benefit is brought by the weighted combination component in truth discovery. The estimated weights can reduce the effect of perturbation, and this leads to unique theoretical analysis that we will present in next section.

6 THEORETICAL ANALYSIS

In this section, we theoretically compare the one-layer and two-layer mechanisms from the perspectives of privacy and utility. It is proven that the proposed two-layer mechanism can provide the same privacy guarantee as the one-layer mechanism while the utility can be significantly improved.

6.1 Privacy Analysis

We start by formally defining the user privacy. Differential privacy [10, 11, 17] is widely adopted to quantify the privacy. However, it assumes that the server is trustable, which is different from our problem setting. Recently, local differential privacy [9, 11, 15] is proposed to deal with the scenario where users do not trust the server. Thus we adopt this privacy definition:

Definition 6.1 (ϵ-Local Differential Privacy). A randomized algorithm $\mathcal{M}$ is ϵ-locally differentially private if for all x_1 and x_2 in $\mathcal{X}$ that are different, and all $\mathcal{S} \subseteq \mathcal{X}$,

$$P\{\mathcal{M}(x_1) \in \mathcal{S}\} \leq e^{\epsilon} \times P\{\mathcal{M}(x_2) \in \mathcal{S}\}. \tag{4}$$

Intuitively, the local differential privacy quantifies the probability that two different values x_1 and x_2 can be perturbed to the same range. We hope that the server cannot distinguish the perturbed values of two different original values. Note that local differential privacy can be regarded as a special case of traditional differential privacy where each dataset only contains one tuple. Thus, for the same privacy parameter ϵ, local differential privacy provides a stronger privacy guarantee than traditional differential privacy.

Next we analyze and compare the privacy guarantees provided by the one-layer and two-layer mechanisms in terms of the above privacy definition.

6.1.1 One-Layer Mechanism. The following theorem states that the one-layer mechanism satisfies ϵ-local differential privacy:

THEOREM 6.2. *When all the users perturb their candidate answers with probability p^f, the one-layer mechanism is $(\ln \frac{(1-p^f)(s-1)}{p^f})$-locally differentially private, where s is the number of possible answers.*

PROOF. According to the privacy definition 6.1, we can calculate the probability ratio $\frac{P\{\mathcal{M}(x_1)\in\mathcal{S}\}}{P\{\mathcal{M}(x_2)\in\mathcal{S}\}}$ and find its maximum. The probability ratio is maximized when we have two different inputs and the output range is identical to one of them. Mathematically, when $x_1 \neq x_2$ and $\mathcal{S} = x_1$, the probability ratio achieves its maximum. According to the perturbation method in Definition 5.1, we have:

$$\frac{P\{\mathcal{M}(x_1) \in \mathcal{S}\}}{P\{\mathcal{M}(x_2) \in \mathcal{S}\}} \leq \frac{P(\mathcal{M}(x_1) = x_1)}{P(\mathcal{M}(x_2) = x_1)} = \frac{1-p^f}{\frac{p^f}{s-1}} = e^{\epsilon}. \tag{5}$$

Thus we get $\epsilon = \ln \frac{(1-p^f)(s-1)}{p^f}$. □

As all the users adopt the same probability to perturb their answers in the one-layer mechanism, the above privacy analysis is applicable to all the users.

6.1.2 Two-Layer Mechanism. For the two-layer mechanism, each user samples his own probability p_u^f to perturb his original answers. The server does not know the sampled probability p_u^f, and the prior knowledge the server has is the hyper distribution f. Here we adopt the widely used uniform distribution $U(a, b)$ as a specific instantiation of hyper distribution f, as the uniform distribution can provide stronger privacy protection than any other distribution. Based on these assumptions, we derive the following theorem:

THEOREM 6.3. *When users sample private perturbation probability p_u^f from uniform distribution $U(a, b)$, the proposed two-layer mechanism is $(\ln \frac{(2-a-b)(s-1)}{a+b})$-locally differentially private.*

The proof of this theorem is similar to the proof of Theorem 6.2, and we omit it due to space limitation. This theorem shows that

when a is fixed, more privacy can be preserved as we increase the value of b. In the two-layer mechanism, all the users follow the same procedure to sample their private perturbation probability independently, so the above analysis holds for all the users.

6.1.3 Comparison. Here, we compare the privacy guarantees provided by one-layer and two-layer mechanisms.

THEOREM 6.4. *The proposed two-layer mechanism provides the same privacy guarantee as the one-layer mechanism when* $\frac{a+b}{2} = p^f$.

This theorem proves that the two-layer mechanism can provide the same level of privacy guarantee to users compared to the one-layer mechanism. In the following, we analyze these mechanisms from the utility perspective, and show that the two-layer mechanism can provide better utility than the one-layer mechanism.

6.2 Utility Analysis

For utility analysis, we adopt the error rate change as the metric. That is, we compare the error rate of aggregated answers derived from the original answers and perturbed answers. The smaller error rate change, the better utility.

6.2.1 Error Bound Before Perturbation. We first quantify the error rate of the aggregated answers derived from original crowdsourced answers.

According to truth discovery, the aggregated answers for binary-choice questions are obtained by weighted voting $x_q^* = H(\{x_q^u\}, \{w_u\}) \stackrel{\text{def}}{=} sign(\sum_{u\in\mathcal{U}} x_q^u \cdot w_u)$, where w_u is the estimated weight for user u. This equation holds as x_q^u is either $+1$ or -1 and thus the sign of the weighted sum determines the final aggregated answer x_q^*. Let p_u be the probability of user u providing correct answers, then the estimated weight w_u is a monotonically increasing function of p_u: $w_u = g(p_u)$.

Assume that we have a set of users $\mathcal{U}$. Let t_q be the true answer for question $q \in Q$. Then the expectation of the error rate based on the candidate answers from users is:

$$\begin{aligned}
Error(H) &= \frac{1}{|Q|}\sum_{q\in Q}\mathbb{I}(t_q, x_q^*) \\
&= \frac{1}{|Q|}\sum_{q\in Q}\mathbb{I}(t_q, sign(\sum_{u\in\mathcal{U}} x_q^u \cdot w_u)) \\
&\le \frac{1}{|Q|}\sum_{q\in Q}\exp\{-t_q \cdot \sum_{u\in\mathcal{U}} x_q^u \cdot w_u\} \\
&= \frac{1}{|Q|}\sum_{q\in Q}\prod_{u\in\mathcal{U}}\exp\{-t_q \cdot x_q^u \cdot w_u\} \\
&= \frac{1}{|Q|}\sum_{q\in Q}\prod_{u\in\mathcal{U}}\left(p_u \cdot e^{-1} + (1-p_u)\cdot e\right)^{w_u} \\
&= \prod_{u\in\mathcal{U}}\left(p_u \cdot e^{-1} + (1-p_u)\cdot e\right)^{w_u} \\
&= \prod_{u\in\mathcal{U}}\left(p_u \cdot e^{-1} + (1-p_u)\cdot e\right)^{g(p_u)}. \qquad (6)
\end{aligned}$$

If we adopt the monotonically increasing function $g(p_u) = \log\frac{p_u}{1-p_u}$ to calculate user weight, then we have:

$$Error(H) \le \prod_{u\in\mathcal{U}}\left(p_u \cdot e^{-1} + (1-p_u)\cdot e\right)^{\log\frac{p_u}{1-p_u}}. \qquad (7)$$

This equation bounds the error rate of the aggregated results $x_q^* = H(\{x_q^u\}, \{w_u\})$.

Let's consider two users. User 1 provides correct answers with probability p_1 where $p_1 \in [0, 0.5]$, and user 2 provides correct answers with probability p_2 where $p_2 = 1 - p_1$. If all the answers from user 2 are flipped to the opposite ones, user 2 will also have the probability p_1 to provide correct answers. Therefore $x_q^2 = -x_q^1$. According to the above weight calculation function, the weight of user 2 is $w_2 = \log\frac{p_2}{1-p_2} = \log\frac{1-p_1}{p_1} = -\log\frac{p_1}{1-p_1} = -w_1$. When we aggregate the answers from all the users, for question q, the weighted vote from user 2 is $x_q^2 \cdot w_2 = (-x_q^1)\cdot(-w_1) = x_q^1 \cdot w_1$, which equals to the weighted vote from user 1. This indicates that user 2 is equivalent to user 1. Thus without loss of generality, we assume that for each user u, $p_u \in [0, 0.5]$.

6.2.2 Error Bound after Perturbation. Consider a user who provides correct information with probability p_u. If this user perturbs his binary answers with probability p_u^f, then his probability to provide correct information after perturbation is:

$$\hat{p}_u = p_u \cdot (1-p_u^f) + (1-p_u)\cdot p_u^f = p_u + p_u^f \cdot (1 - 2\cdot p_u). \qquad (8)$$

Assume that a set of users $\mathcal{U}$ is involved, and each of them provides correct answers with probability p_u. In order to protect the privacy of users, they will perturb their answers based on the proposed mechanisms. According to Eq. (7) and (8), the error rate of the aggregated results derived from the perturbed data will be:

$$Error(H_{perturbed}) \le \prod_{u\in\mathcal{U}}\left(\hat{p}_u \cdot e^{-1} + (1-\hat{p}_u)\cdot e\right)^{\log\frac{\hat{p}_u}{1-\hat{p}_u}}. \qquad (9)$$

Let's denote $\left(p_u \cdot e^{-1} + (1-p_u)\cdot e\right)^{\log\frac{p_u}{1-p_u}}$ as function $G(p_u)$. In order to compare $Error(H)$ and $Error(H_{perturbed})$, we use $G'(p_u) = 2\cdot p_u$ to approximate function $G(p_u)$ for the purpose of simplification as these two functions are very close to each other.

Plugging this approximation into Eq. (9), we can simplify $Error(H_{perturbed})$ as:

$$\begin{aligned}
Error(H_{perturbed}) &\lessapprox \prod_{u\in\mathcal{U}} G'(\hat{p}_u) = \prod_{u\in\mathcal{U}} 2\cdot\hat{p}_u \\
&= \prod_{u\in\mathcal{U}} 2\cdot(p_u + p_u^f\cdot(1-2\cdot p_u)) \\
&= (\prod_{u\in\mathcal{U}} 2\cdot p_u) + \Delta \\
&= Error(H) + \Delta, \qquad (10)
\end{aligned}$$

where Δ denotes the error rate change.

6.2.3 Important Users. From Eq. (7), we can see that the users will not equally affect the error bound. If a user's p_u is close to 0.5, he will not contribute too much to lower the error bounds $Error(H)$ and $Error(H_{perturbed})$. This motivates us to focus on important users who will affect the error bound significantly.

In order to define the important users, we first define the metric to calculate the importance score of user u: $Im(u) = \frac{Error(H_{-u})}{Error(H)}$, where $Error(H_{-u})$ denotes the error rate of aggregated answers without considering the information from user u. That is, the importance score is the ratio between error rates of aggregated answers with and without a particular user's information. If a user is important, his answers will greatly reduce the error rate of aggregated results, and thus the important score $Im(u)$ will be large. On the other hand, if the important score of a user is close to 1, it is indicated that this user makes a negligible contribution to the aggregation. According to Eq. (7), $Im(u) = \frac{1}{G(p_u)}$.

Based on the important score, we define the set of important users as $\mathcal{U}_{important} = \{u : Im(u) \geq c \cdot \max_{u' \in \mathcal{U}} Im(u')\}$, where $c \in (0, 1]$ is a threshold to determine whether a user is important or not. When c is set to be 1, $\mathcal{U}_{important}$ only includes the best user(s). When c is close to 1, $\mathcal{U}_{important}$ will also include the users who have importance scores close to the best user(s).

Let's replace the user set $\mathcal{U}$ in Eq. (10) with $\mathcal{U}_{important}$:

$$\begin{aligned} Error(H_{perturbed}) &\leq \prod_{u \in \mathcal{U}_{important}} G(\hat{p}_u) \approx \prod_{u \in \mathcal{U}_{important}} G'(\hat{p}_u) \\ &= \prod_{u \in \mathcal{U}_{important}} 2 \cdot (p_u + p_u^f \cdot (1 - 2 \cdot p_u)) \\ &= (\prod_{u \in \mathcal{U}_{important}} 2 \cdot p_u) + \Delta = Error(H) + \Delta, \end{aligned}$$

where Δ is:

$$\begin{aligned} \Delta = & \prod_{u \in \mathcal{U}_{important}} p_u^f \cdot (1 - 2 \cdot p_u)) \\ & + \sum_{u \in \mathcal{U}_{important}} \left(p_u \prod_{u' \neq u} p_{u'}^f \cdot (1 - 2 \cdot p_{u'}) \right) \\ & + \cdots \end{aligned} \tag{11}$$

For important users, their corresponding p_u is close to 0 (note we have demonstrated that users with $p_u = 0$ are equivalent to users with $p_u = 1$), so the error rate change Δ is dominated by the term $\prod_{u \in \mathcal{U}_{important}} p_u^f$.

One-Layer Mechanism. In the proposed one-layer mechanism, all the users are forced to adopt the same probability to perturb their answers, i.e., $p_u^f = p^f$. Thus the difference of $Error(H)$ and $Error(H_{perturbed})$ can be calculated as $\Delta = \prod_{u \in \mathcal{U}_{important}} p^f$.

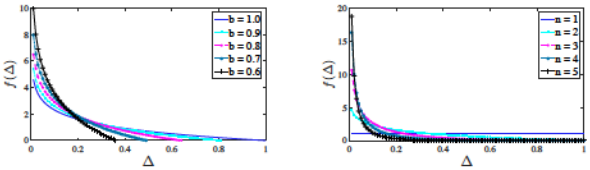
Recall that in the one-layer mechanism, the parameter p^f is pre-defined by the server. When p^f is small (weak privacy), the error rate change of the one-layer mechanism will be small (good utility). However, to provide users with strong privacy guarantees, p^f needs to be set as a large value. As a result, the error rate change Δ increases quickly as we increase the parameter p^f. This will be confirmed by the experiments on various datasets in Section 7.

Two-Layer Mechanism. In the proposed two-layer mechanism, $\{p_u^f\}$ are independently sampled from an identical uniform distribution $U(a, b)$, and thus Δ is the product of n independent uniform random variables. When a is set to be 0, the probability density function (PDF) of Δ is:

$$f(\Delta) = \begin{cases} \frac{1}{b^n \cdot (n-1)!} (\log \frac{b^n}{\Delta})^{n-1}, & \Delta \in [0, b^n] \\ 0, & otherwise, \end{cases}$$

where b is the maximum probability that can be sampled from the hyper distribution, and n is the number of important users.

In Figures 1, we plot the PDFs by varying parameters b and n.



(a) n = 2 with different b's (b) b = 1 with different n's

Figure 1: PDF of $f(\Delta)$

Based on the above analysis, the following claims can be inferred for the proposed two-layer mechanism, and they will be experimentally confirmed in Section 7.

- The error rate change Δ of the two-layer mechanism follows a long-tail distribution: the probability of observing a small Δ is high, while a large Δ occurs with very low probability. In other words, good utility can be guaranteed with high probability.
- The above analysis reveals the relationship between the utility and the hyper distribution $f = U(a, b)$. Let's assume that a is fixed. Figure 1a shows that when the parameter b decreases, the probability of the error rate change (Δ) being small will increase, and thus better utility can be guaranteed.
- The above analysis also indicates the relationship between the utility and important users: more important users can lead to better utility. In Figure 1b, the probability of Δ being small increases as more important users are available (large n).

6.3 Summarization

In section 6.1, we first conduct privacy analysis and Theorem 6.4 states that the one-layer and two-layer mechanisms can provide the same privacy guarantee if $p^f = \frac{a+b}{2}$. Then in the following section 6.2, we show that when they provide the same privacy guarantee, their utility (error rate change Δ) is quite different. For the one-layer mechanism, the error rate change is $\Delta = (p^f)^n = (\frac{b}{2})^n$ (a is set to be 0), while for the two-layer mechanism, the error rate change follows a long-tail distribution in which Δ is small with high probability. This difference is caused by the fact that users sample their own probabilities to perturb their answers. Such diversity in the flipping probabilities is the key to better utility. Among the users, there are some users who have relatively good quality, and their answers can guide the weight estimation even when b is quite large. In contrast, in the one-layer mechanism, all the users are forced to perturb their answers with the same probability p^f, and the benefit of user weight estimation is limited. Thus the two-layer mechanism has a higher probability to achieve better utility compared to one-layer mechanism.

7 EXPERIMENTS

In the previous section, we theoretically compare the one-layer mechanism and the two-layer mechanism in terms of both privacy and utility. In this section, we conduct experiments on two real-world datasets to confirm the theoretical analysis.

7.1 Experiment Setup

7.1.1 Datasets. To validate the advantages of the proposed privacy-preserving mechanism, we adopt the following real-world crowdsourced datasets.

Peer Grading Dataset. This dataset is collected from a graduate level course, which involves 72 students (users). In this course, all the students are divided into 26 groups and each group gives a presentation for their course project. During each group's presentation, the other students fill an evaluation form for this group based on the guidelines from the instructor to evaluate whether the performance of this group is satisfactory or not. The instructor also provides her grades for each group, which can be regarded as groundtruth. Although peer grading is an effective way to assess students' course projects, the graders (students) may have privacy concern to share their evaluations for other students. Thus this dataset is a perfect testbed to evaluate the proposed privacy-preserving mechanisms. Some statistics of this dataset are summarized in Table 1.

Duchenne Smile Dataset. This dataset is also collected from a real-world application, in which each question is to judge whether the smiling face in an image is Duchenne or Non-Duchenne. The authors in [30] also obtain the labels (candidate answers) from workers (users) on Amazon Mechanical Turk. Part of groundtruth labels are provided by certified experts in the Facial Action Coding System. Statistics about this dataset can also be found in Table 1.

Table 1: Statistics of Real-world Datasets

	Peer Grading Dataset	Duchenne Smile Dataset
#question	26	2134
#users	72	64
#answers	360	17729
#groundtruth	26	159

7.1.2 Performance Measure. To quantify the privacy, we adopt the ϵ local differential privacy defined in Definition 6.1. As mentioned before, local differential privacy can be treated as a special case of the traditional differential privacy when the considered dataset contains only one tuple. The smaller ϵ indicates stronger privacy.

For utility measure, we adopt the metric of Error Rate Change, that is, we compare the error rate of the aggregated answers derived from users' original answers and the perturbed answers. Small error rate change indicates that the perturbation has little effect on the performance, and thus good utility is achieved.

7.1.3 Compared Methods. We evaluate the one-layer and two-layer mechanisms to confirm the theoretical analysis. Besides, we also replace the truth discovery method with majority voting to illustrate the benefits of considering user weights in aggregation. We denote Majority Voting and Truth Discovery as MV and TD respectively. Thus "MV with One-layer", "TD with one-layer" and "MV with Two-layer" are baseline methods. Note that although randomized response has a different goal with the proposed two-layer mechanism, "MV with One-layer" and "MV with Two-layer" can be considered as the adapted versions of general *randomized response* and *FRAPP method* respectively.

7.1.4 Environment. All the methods are implemented on the same platform (MATLAB), and run on the same machine with 8G RAM, Intel Core i5 processor. As the perturbation is random, each performance result reported below is the mean of 100 independent trials to reduce the effect of randomness.

7.2 Performance Comparison

We first evaluate the performance of the privacy-preserving mechanisms on Peer Grading dataset and Duchenne Smile dataset. By varying the privacy parameter ϵ, different levels of perturbation are performed according to the one-layer and two-layer mechanisms. The utility, i.e., Error Rate Change, under various scenarios is reported in Tables 2 and 3. Note that $\epsilon = 0$ is a special case enabled by the unique characteristics of categorical data. Consider a question with two possible answers. When users flip their answers with probability 0.5, the answers from all the users have the same probability distribution, i.e., users will provide any possible answer with probability 0.5. In this case, the noise (answer perturbation) is so large that users' original answers have no influence on the probability distribution of perturbed answers. According to the definition of differential privacy, these users become indistinguishable, i.e., $\epsilon = 0$. For the general cases, Theorems 6.2 and 6.3 demonstrate that ϵ can be 0 with certain parameter settings.

Table 2: Performance Comparison on Peer Grading Dataset

	One-layer Mechanism		Two-layer Mechanism	
ϵ	MV	TD	MV	TD
1.0	0.1019	0.0850	0.0885	0.0619
0.9	0.1135	0.0954	0.1112	0.0800
0.8	0.1300	0.1127	0.1212	0.0827
0.7	0.1404	0.1135	0.1265	0.0923
0.6	0.1615	0.1408	0.1554	0.1138
0.5	0.1669	0.1546	0.1631	0.1262
0.4	0.1788	0.1604	0.1696	0.1354
0.3	0.1842	0.1658	0.1785	0.1415
0.2	0.2042	0.1823	0.1938	0.1581
0.1	0.2269	0.2088	0.2165	0.1723
0.01	0.2373	0.2135	0.2327	0.1865
0.001	0.2485	0.2196	0.2377	0.1958
0.0	0.2504	0.2212	0.2419	0.1992

From Tables 2 and 3, we can observe that the ranges of Error Rate Change on two real-world datasets are slightly different. This phenomenon is caused by the quality of the original (clean) datasets. Without any perturbation, the accuracy of aggregated answers is 0.73 and 0.76 for Peer Grading dataset and Duchenne Smile dataset respectively. In the case of $\epsilon = 0$, all the users randomly flip their answers, and the accuracy of the aggregated answers will be around

Table 3: Performance Comparison on Duchenne Smile Dataset

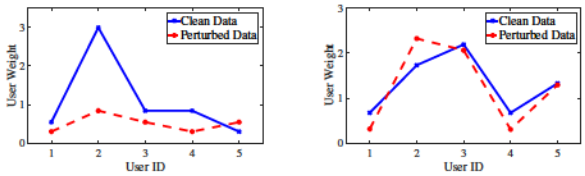
	One-layer Mechanism		Two-layer Mechanism	
ϵ	MV	TD	MV	TD
1.0	0.1228	0.0735	0.1240	0.0560
0.9	0.1296	0.0870	0.1337	0.0629
0.8	0.1438	0.1104	0.1447	0.0709
0.7	0.1459	0.1157	0.1564	0.0821
0.6	0.1699	0.1321	0.1701	0.0906
0.5	0.1784	0.1562	0.1738	0.0962
0.4	0.1917	0.1769	0.1903	0.1081
0.3	0.2126	0.1930	0.2133	0.1271
0.2	0.2218	0.2016	0.2160	0.1379
0.1	0.2335	0.2086	0.2248	0.1511
0.01	0.2406	0.2150	0.2362	0.1742
0.001	0.2414	0.2178	0.2426	0.1774
0.0	0.2516	0.2277	0.2447	0.1793

0.5. Thus the error rate change under $\epsilon = 0$ will be around 0.23 and 0.26 for Peer Grading and Duchenne Smile datasets respectively, which is consistent with our experimental results. Note that the quality control of the original crowdsourced dataset is not within the scope of the proposed privacy-preserving mechanism. Our focus is on the performance change before and after perturbation, which indicates the utility of the proposed approach.

We further analyze the experimental results from the following four aspects:

(1) Comparison of MV under the one-layer and two-layer mechanisms: In the one-layer and two-layer mechanisms, users either adopt the same probability or sample their own probabilities to perturb their candidate answers. Although the ways of choosing probability are different, to provide the same level of privacy guarantee, the same level of noise is required to be injected. From both Tables 2 and 3, we can observe the similar performance of majority voting under one-layer and two-layer mechanisms, which confirms that the same level of noise is added by the one-layer and two-layer mechanisms.

(2) Comparison of MV and TD under the two-layer mechanism: Under the two-layer mechanism, TD gives better performance than MV. The reason is that truth discovery estimates user weights and incorporates such weights into aggregation, while majority voting does not consider the variety in user quality. To demonstrate the user weight estimation, we plot the estimated weights for some randomly selected users in Figure 2. The blue lines show the estimated user weights based on users' original answers, and the red dot lines illustrate the estimated user weights based on users' perturbed answers. Comparing the blue lines and red lines, we can observe that the weight of a user will be reduced if he adopts a big probability to perturb his answers (for example, the perturbation probability of the 2-nd user in Figure 2a is 0.46). On the other hand, if a user adopts a small probability to perturb his answers, his weight will keep the same, or be adjusted slightly higher (for example, the perturbation probability of the 2-nd user in Figure 2b is 0.01). This is because that the estimated user weights indicate the relative quality of users, and when the quality of other users decreases, users who do not change their quality might be assigned higher weights. As truth discovery method can automatically adjust the estimated user weights, the effect of the perturbation can be partly absorbed and thus the performance change will be smaller than majority voting method.



(a) Peer Grading Dataset (b) Duchenne Smile Dataset

Figure 2: Estimated User Weights

(3) Comparison of TD under the one-layer and two-layer mechanisms: Both Tables 2 and 3 demonstrate that TD under two-layer mechanism gives better performance than the one-layer mechanism. The reason is illustrated in the previous section: in the two-layer mechanism, users sample their own probabilities to perturb their grades or candidate answers. Some important users may sample small probabilities p_u^f and their quality is still high. Thus these important users will be assigned high weights and lead to small error rate change (good utility). In contrast, in the one-layer mechanism, all the users adopt the same probability to perturb their answers. To guarantee strong privacy, the probability p^f is required to be large, and thus the quality of all the users dramatically decreases. In this case (small ϵ), the user weight estimation does not make a big difference, and thus the performance of the truth discovery with one-layer mechanism is close to the majority voting method.

(4) Utility-Privacy trade-off: The trade-off between utility and privacy can be observed from the performance of either one-layer mechanism or the two-layer mechanism in Tables 2 and 3. To clearly show the trade-off, we also plot the utility w.r.t. the privacy on both Peer Grading and Duchenne Smile datasets in Figures 3 and 4. We can observe that to provide strong privacy (small ϵ), more perturbation should be performed and thus the utility is sacrificed. To keep good utility, the provided privacy guarantee cannot be too strong. However, comparing with other methods, truth discovery with the two-layer mechanism (blue dot line in Figures 3 and 4) can tolerate more perturbation.

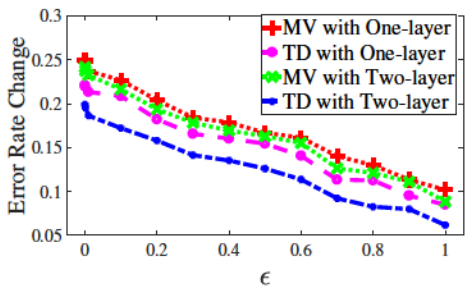


Figure 3: Utility-Privacy Trade-off on Peer Grading Dataset

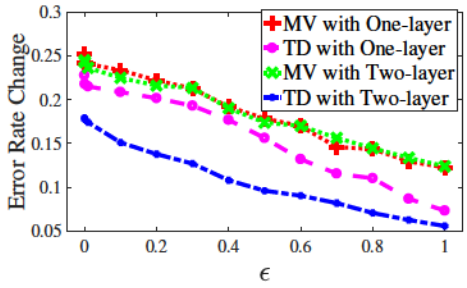


Figure 4: Utility-Privacy Trade-off on Duchenne Smile Dataset

8 CONCLUSIONS

Crowdsourcing has been successfully applied to solve many challenging question answering tasks. However, individual users may have the privacy concern when sharing their sensitive answers. Motivated by this strong need, we propose efficient and effective two-layer mechanism for crowdsourced question answering, which allows users to randomly perturb their answers and then conduct truth discovery on the perturbed answers. Theoretical analysis proves that the two-layer mechanism provides the same level of privacy guarantee as the one-layer mechanism. Furthermore, we theoretically show that good utility can be guaranteed by the two-layer mechanism even with strong privacy constraints. This benefit is brought by the fact that the two-layer mechanism fully utilizes the properties of truth discovery which automatically estimates user quality to derive aggregated answers. The advantage of the proposed two-layer mechanism is confirmed by the experimental results on two real-world datasets. With our developed privacy-preserving mechanism, we can greatly broaden the application domain of truth discovery and enable tasks that would otherwise be infeasible due to privacy concerns.

9 ACKNOWLEDGMENTS

This work was sponsored in part by US National Science Foundation under grant IIS-1553411, CNS-1742845, CNS-1652503 and CNS-1737590. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agency.

REFERENCES

[1] Shipra Agrawal, Jayant R Haritsa, and B Aditya Prakash. 2009. FRAPP: a framework for high-accuracy privacy-preserving mining. *Data Mining and Knowledge Discovery* 18, 1 (2009), 101–139.
[2] Yoram Bachrach, Tom Minka, John Guiver, and Thore Graepel. 2012. How to Grade a Test without Knowing the Answers – A Bayesian Graphical Model for Adaptive Crowdsourcing and Aptitude Testing. In *Proc. of ICML*. 255–262.
[3] Anirban Basu, Jaideep Vaidya, Juan Camilo Corena, Shinsaku Kiyomoto, Stephen Marsh, Guibing Guo, Jie Zhang, and Yutaka Miyake. 2014. Opinions of people: factoring in privacy and trust. *ACM SIGAPP Applied Computing Review* 14, 3 (2014), 7–21.
[4] Elisa Bertino, Beng Chin Ooi, Yanjiang Yang, and Robert H Deng. 2005. Privacy and ownership preserving of outsourced medical data. In *Proc. of ICDE*. 521–532.
[5] Arijit Chaudhuri and Rahul Mukerjee. 1988. *Randomized response: Theory and techniques*. Marcel Dekker New York.
[6] Chris Clifton, Murat Kantarcioglu, AnHai Doan, Gunther Schadow, Jaideep Vaidya, Ahmed Elmagarmid, and Dan Suciu. 2004. Privacy-preserving data integration and sharing. In *Proc. of ACM SIGMOD workshop*. 19–26.
[7] Alexander Philip Dawid and Allan M Skene. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied statistics* (1979), 20–28.
[8] Xin Luna Dong, Laure Berti-Equille, and Divesh Srivastava. 2009. Integrating Conflicting Data: The Role of Source Dependence. *PVLDB* 2, 1 (2009), 550–561.
[9] John C Duchi, Michael I Jordan, and Martin J Wainwright. 2013. Local privacy and statistical minimax rates. In *Proc. of FOCS*. 429–438.
[10] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography*. 265–284.
[11] Cynthia Dwork and Aaron Roth. 2014. The algorithmic foundations of differential privacy. (2014).
[12] Úlfar Erlingsson, Aleksandra Korolova, and Vasyl Pihur. 2014. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. In *Proc. of CCS*. 1054–1067.
[13] Alban Galland, Serge Abiteboul, Amélie Marian, and Pierre Senellart. 2010. Corroborating Information from Disagreeing Views. In *Proc. of WSDM*. 131–140.
[14] Haibo Hu, Jianliang Xu, Sai Tung On, Jing Du, and Joseph Kee-Yin Ng. 2010. Privacy-aware location data publishing. *ACM Transactions on Database Systems (TODS)* 35, 3 (2010), 18.
[15] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. 2014. Extremal Mechanisms for Local Differential Privacy. In *NIPS*. 2879–2887.
[16] Hiroshi Kajino, Hiromi Arai, and Hisashi Kashima. 2014. Preserving worker privacy in crowdsourcing. *Data Mining and Knowledge Discovery* 28, 5-6 (2014), 1314–1335.
[17] Daniel Kifer and Ashwin Machanavajjhala. 2012. A rigorous and customizable framework for privacy. In *Proc. of PODS*. 77–88.
[18] Qi Li, Yaliang Li, Jing Gao, Lu Su, Bo Zhao, Demirbas Murat, Wei Fan, and Jiawei Han. 2015. A Confidence-Aware Approach for Truth Discovery on Long-Tail Data. *PVLDB* 8, 4 (2015), 425–436.
[19] Qi Li, Yaliang Li, Jing Gao, Bo Zhao, Wei Fan, and Jiawei Han. 2014. Resolving Conflicts in Heterogeneous Data by Truth Discovery and Source Reliability Estimation. In *Proc. of SIGMOD*. 1187–1198.
[20] Xian Li, Xin Luna Dong, Kenneth B. Lyons, Weiyi Meng, and Divesh Srivastava. 2012. Truth Finding on the Deep Web: Is the Problem Solved? *PVLDB* 6, 2 (2012), 97–108.
[21] Yaliang Li, Jing Gao, Chuishi Meng, Qi Li, Lu Su, Bo Zhao, Wei Fan, and Jiawei Han. 2015. A Survey on Truth Discovery. *ACM SIGKDD Explorations Newsletter* 17, 2 (2015), 1–16.
[22] Fenglong Ma, Yaliang Li, Qi Li, Minghui Qiu, Jing Gao, Shi Zhi, Lu Su, Bo Zhao, Heng Ji, and Jiawei Han. 2015. FaitCrowd: Fine Grained Truth Discovery for Crowdsourced Data Aggregation. In *Proc. of KDD*. 745–754.
[23] Chenglin Miao, Wenjun Jiang, Lu Su, Yaliang Li, Suxin Guo, Zhan Qin, Houping Xiao, Jing Gao, and Kui Ren. 2015. Cloud-enabled privacy-preserving truth discovery in crowd sensing systems. In *Proc. of SenSys*. 183–196.
[24] Chenglin Miao, Lu Su, Wenjun Jiang, Yaliang Li, and Miaomiao Tian. 2017. A Lightweight Privacy-Preserving Truth Discovery Framework for Mobile Crowd Sensing Systems. In *Proc. of INFOCOM*. 1539–1547.
[25] Liam O' Neill, Franklin Dexter, and Nan Zhang. 2016. The Risks to Patient Privacy from Publishing Data from Clinical Anesthesia Studies. *Anesthesia & Analgesia* 122, 6 (2016), 2017–2027.
[26] Vibhor Rastogi and Suman Nath. 2010. Differentially private aggregation of distributed time-series with transformation and encryption. In *Proc. of SIGMOD*. 735–746.
[27] Elaine Shi, T-H Hubert Chan, Eleanor G Rieffel, Richard Chow, and Dawn Song. 2011. Privacy-Preserving Aggregation of Time-Series Data.. In *Proc. of NDSS*.
[28] R. Snow, B. O'Connor, D. Jurafsky, and A. Ng. 2008. Cheap and Fast - But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks. In *Proc. of EMNLP'08*. 254–263.
[29] Hien To, Gabriel Ghinita, and Cyrus Shahabi. 2014. A Framework for Protecting Worker Location Privacy in Spatial Crowdsourcing. *PVLDB* 7, 10 (2014), 919–930.
[30] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, and J. Movellan. 2009. Whose Vote Should Count More: Optimal Integration of Labelers of Unknown Expertise. In *NIPS*. 2035–2043.
[31] Xiaokui Xiao, Yufei Tao, and Minghua Chen. 2009. Optimal random perturbation at multiple privacy levels. *PVLDB* 2, 1 (2009), 814–825.
[32] Xiaoxin Yin, Jiawei Han, and Philip S. Yu. 2007. Truth discovery with multiple conflicting information providers on the web. In *Proc. of KDD*. 1048–1052.
[33] Ye Zhang, Wai-Kit Wong, Siu-Ming Yiu, Nikos Mamoulis, and David W Cheung. 2013. Lightweight privacy-preserving peer-to-peer data integration. In *PVLDB*, Vol. 6. 157–168.
[34] Yifeng Zheng, Huayi Duan, Xingliang Yuan, and Cong Wang. 2017. Privacy-Aware and Efficient Mobile Crowdsensing with Truth Discovery. *IEEE Transactions on Dependable and Secure Computing* (2017).
[35] Yudian Zheng, Guoliang Li, Yuanbing Li, Caihua Shan, and Reynold Cheng. 2017. Truth inference in crowdsourcing: is the problem solved? *PVLDB* 10, 5 (2017), 541–552.
[36] D. Zhou, J. C. Platt, S. Basu, and Y. Mao. 2012. Learning from the Wisdom of Crowds by Minimax Entropy. In *NIPS*. 2204–2212.