Privacy-Preserving Streaming Truth Discovery in Crowdsourcing With Differential Privacy

Dan Wang, *Student Member, IEEE*, Ju Ren, *Member, IEEE*, Zhibo Wang, *Senior Member, IEEE*, Xiaoyi Pang, Yaoxue Zhang, *Senior Member, IEEE*, and Xuemin Shen, *Fellow, IEEE*

Abstract—Differential privacy (DP) has gained popularity in truth discovery recently due to its strong privacy guarantee. However, existing DP mechanisms for streaming data publication are not suitable for truth discovery as they fail to consider the different reliabilities of individuals, while the DP-based approaches for truth discovery are not suitable for streaming data because they ignore the correlations between truths over time. Directly applying these existing methods to streaming crowdsourced data would lead to low accuracy of the discovered truth. To solve this problem, in this paper, we propose an edge computing based privacy-preserving truth discovery mechanism, named PrivSTD, for streaming crowdsourced data to realize high accuracy of discovered truth while protecting the privacy of workers. Specifically, edge servers are introduced between the untrusted cloud server and workers to securely calculate the local truths and workers' reliabilities. A truth-dependent budget recycle mechanism is proposed for each edge server to adaptively determine the perturbed timestamp and allocate the privacy budget according to the changing pattern of local truths. Besides, a reliability-based perturbation mechanism is proposed to reduce the perturbation magnitude on the basis of worker's reliability. We theoretical analyze the data utility and computation cost of PrivSTD, and prove that PrivSTD can satisfy w-event (ϵ, δ)-differential privacy. Extensive experimental results on synthetic and real-world datasets demonstrate that PrivSTD achieves better utility than the state-of-the-art approaches.

Index Terms—Crowdsourcing, truth discovery, streaming data, differential privacy, edge computing

1 Introduction

1.1 Background and Motivation

Crowdsourcing is a technology of outsourcing tasks to a group of public workers. It gains its popularity through providing a flexible, time- and cost-saving way to execute large amounts of tasks that require human intelligence [1], [2]. A typical crowdsourcing application involves three entities: a centralized crowdsourcing platform (or a cloud server, e.g., Amazon mechanical Turk,[1] MicroWorkers,[2] and crowSPRING[3]), the task requesters, and the workers. A task requester that needs information or services releases tasks to a crowdsourcing platform, and workers who accept tasks would provide their answers to the platform in exchange for payments. Then, the task requester chooses truths from the collected answers. The continuously collected answers are called streaming data, which has increasing prominences in a wide range of applications. For example, in traffic monitoring applications, traffic information is reported by multiple users frequently in real time. In healthcare monitoring applications, the health data of patients are continuously recorded. In environmental monitoring, temperatures in different places need to be collected regularly. [3], [4]

Although crowdsourcing provides an easy way to accomplish tasks, it faces the problem of noisy or conflicted answers. An intuitive approach for solving the conflicts is to take the average of numerical data or accept the majority of categorical data as the truth. However, averaging or majority voting assumes all the workers are equally reliable [5], which may not hold in some cases. To ensure the authority of the result, the result must be closer to the information provided by reliable workers. However, a challenge comes out that the worker's reliability is usually unknown in advance.

In order to solve this challenge, a series of truth discovery methods [4], [6], [7], [8], [9] have been proposed to estimate the workers' reliabilities in the form of weights without any supervision. Since both worker's weight and truth are unknown in prior and can only be inferred from answers, the weight estimation and truth finding steps are tightly combined under a simple principle that a worker providing answers closer to the truths will be assigned higher weight, and the answers provided by a worker with a higher weight are closer to the truths.

[1] https://www.mturk.com/
[2] https://www.microworkers.com/
[3] https://www.crowdspring.com/

- *Dan Wang and Ju Ren are with the School of Computer Science and Engineering, Central South University, Changsha 410083, China. E-mail: {wang_dan113, renju}@csu.edu.cn.*
- *Yaoxue Zhang is with the School of Computer Science and Engineering, Central South University, Changsha 410083, China, and also with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China. E-mail: zyx@csu.edu.cn.*
- *Zhibo Wang is with the Institute of Cyber Science and Technology, College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China. E-mail: zhibowang@zju.edu.cn.*
- *Xiaoyi Pang is with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China. E-mail: xypang@whu.edu.cn.*
- *Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo N2L 3G1, Canada. E-mail: sshen@uwaterloo.ca.*

Manuscript received 14 Sept. 2020; revised 1 Feb. 2021; accepted 24 Feb. 2021. Date of publication 1 Mar. 2021; date of current version 31 Aug. 2022. (Corresponding author: Ju Ren.) Digital Object Identifier no. 10.1109/TMC.2021.3062775

While truth discovery provides an effective way for inferring the true answers, it causes privacy concerns of workers, because the centralized platform (the cloud server) can easily obtain private information of workers from the answers. For example, aggregating the medical data from participants helps discover the effects of new drugs, but it may also leak their private health conditions. One single record may contain little personal information, but continuously collecting records and combining with auxiliary information can sometimes identify people uniquely, as mentioned in the field of privacy-preserving data mining [10], [11], [12], [13]. As a result, protecting the answers from each worker in truth discovery is very important, especially in the streaming crowdsourcing applications, which continuously collect sensitive data. However, the streaming data also poses significant challenges in privacy preservation, because the data distribution of the data stream is usually unknown and constantly changes during the whole data collection process.

1.2 Limitations of Prior Arts

In order to protect the privacy of workers, several works explored privacy-preserving truth discovery mechanisms by using encryption techniques [12], [13]. Miao *et al.* [3], [14] performed weighted aggregation on users' encrypted data using homomorphic cryptosystem to protect both users' data and their reliability scores. References [15], [16], [17], [18] further improved the bandwidth and computation performance on individual users. However, these approaches require multiple interactions between servers and users. To simplify the process, Tang *et al.* [19] implemented a non-interactive privacy-preserving truth discovery system with garbled circuit. Zheng *et al.* [20] argued that the inferred truths were private for task requester, thus they proposed an encrypted confidence-aware truth discovery approach to protect the sensory data and reliability degrees of users, as well as the inferred truths of the requester.

Comparing with these encrpytion techniques, differential privacy (DP) [21], [22], [23] is more widely used in data aggregation due to the advantages of low computational cost and provable privacy protection. However, the traditional DP mechanisms are not suitable for truth discovery, because they do not consider the different reliability of workers in truth discovery, consequently injecting the maximum noise to each worker. In order to improve the accuracy of perturbed truths, several works have integrated DP mechanism into the truth discovery algorithm. Sun *et al.* [24] introduced the matrix factorization technology into truth discovery. They added Laplace noise into the loss function of matrix factorization instead of the original answers to reduce the effect of noise on the aggregation. However, this method did not consider different reliabilities of workers. Li *et al.* [25] flipped the candidate answers with a specific probability that was sampled from a hyper distribution to reduce the perturbation magnitude. But this method only aims at binary data, and is not suitable for continuous data that used in our work, since the candidates of continuous data are endless. Then, Li *et al.* [26] proposed a similar method for continuous data. They assumed that the errors of workers follow normal distributions, of which the variances are sampled from an exponential distribution that scaled by a hyper parameter. This hyper parameter controls both the privacy protection degree and the utility of truth discovery. Following the approach in reference [26], Sun *et al.* [27] proposed Paris-TD, a privacy-preserving incentive mechanism for truth discovery and Xu *et al.* [28] introduced a verifiable and privacy-aware truth discovery protocol that enables any entity to verify the correctness of aggregated results returned from the server. Vadavalli *et al.* [29] used random disturbance to protect the privacy information in the healthcare database. But it was based on the assumption that the healthcare database was trusted. While in our work, we think that the data center (cloud server) is untrusted, and the answers must be protected before being transmitted to the cloud server. Moreover, all of these methods were proposed for one-time truth discovery, while the mechanism we proposed is for the truth discovery of streaming data. Simply applying the existing works to the streaming data may incur too much noise, since they did not fully take advantage of the correlations among truths in the data stream.

Motivated by the above-mentioned problems in the truth discovery of streaming data, we aim to design a privacy-preserving truth discovery algorithm with w-event DP for the streaming crowdsourced data, which can simultaneously take into consideration both the correlations among truths over time and the characteristic of workers' reliabilities to achieve high accuracy of discovered streaming truth while protecting the privacy of workers. To realize these objectives, we are still facing several challenges. First, *it is challenging to explore the evolving pattern of truths*. The untrusted cloud server cannot access to the true answers while an individual worker cannot calculate truth over his own answers. Thus, neither the cloud server nor a specific worker can obtain the correlations among truths over time. Second and the most important, since the truths and workers' reliabilities are unknown in prior, *how to reflect the relationships among evolving truths, workers' reliabilities and noise magnitude is very challenging*. Lastly, since the allocated privacy budget to each data point is also changing over time, *how to realize w-event differential privacy for each participant should be carefully investigated*.

1.3 Contributions

To overcome these challenges, we propose an edge computing based **Priv**acy-preserving **S**treaming data **T**ruth **D**iscovery algorithm, called PrivSTD, to achieve accurate truth discovery for streaming data while protecting the privacy of workers against untrusted cloud server. Specifically, multiple edge servers are introduced to help workers securely calculate the local truths and workers' reliabilities. A truth-dependent budget recycle mechanism is then proposed for each edge server to adaptively determine the perturbed timestamp and allocate the privacy budget according to the evolution of local truths. More importantly, a reliability-based perturbation mechanism is designed to efficiently improve the accuracy of estimated truths. In particular, instead of achieving w-event ϵ-differential privacy, we relax the privacy constraint of PrivSTD for higher accuracy, which satisfies w-event (ϵ, δ)-differential privacy.

To summarize, this paper makes the following contributions:

- To the best of our knowledge, this is the first work exploring privacy-preserving truth discovery over streaming data. We propose a novel edge computing based solution to realize accurate truth discovery under strong w-event (ϵ, δ)-differential privacy protection by taking into consideration both the different reliabilities among participants and the correlations between data points over time.

- We introduce edge servers to help workers securely estimate local truths and workers' reliabilities, based on which the truth dependent budget recycle mechanism and reliability-based perturbation mechanism are proposed to adaptively allocate privacy budget and add appropriate noise for each worker based on evolving truths and reliabilities.
- We theoretical analyze the data utility and computation cost of PrivSTD and prove that PrivSTD satisfies w-event (ϵ, δ)-DP. The experimental results on both synthetic datasets and two real-world datasets demonstrate that PrivSTD outperforms the state-of-the-arts.

The remainder of this paper is organized as follows. Section 2 introduces some preliminaries. The overview of PrivSTD is presented in Section 3. We detail the design of PrivSTD in Section 4. Experimental results are provided in Section 5. Finally, Section 6 draws the conclusions.

2 Preliminaries

This section first introduces the system model of traditional privacy-preserving truth discovery, and then describes the definitions of DP and w-event DP as well as the composition characteristic. The important notations frequently used throughout the paper are listed in Table 1.

2.1 Truth Discovery

Suppose there are M crowd participants and N tasks in a crowdsourcing system. Let $X_i^t = \{x_{i,1}^t, x_{i,2}^t, \ldots, x_{i,N}^t\}$ denote the answers of worker i at timestamp t, and $\boldsymbol{X}^t = \{X_1^t, X_2^t, \ldots, X_M^t\}$ denote the answer set of all workers. The goal of truth discovery is to infer the truth set $Z^t = \{z_1^t, z_2^t, \ldots, z_N^t\}$ from $\boldsymbol{X}^t$. Worker's error is the distances between answers and truths that are denoted as $R_i^t = \{r_{i,1}^t, r_{i,2}^t, \ldots, r_{i,N}^t\}$, which are inversely proportional to worker's reliability. It is observed that in truth discovery worker's error follows the normal distribution $N(0, (\sigma_i^t)^2)$ [4], [26]. Thus, we can use the standard error deviation σ_i^t to reflect the worker's reliability. The higher the worker's reliability is, the smaller the error variance will be. The standard error deviation set from all workers is denoted as $\boldsymbol{\sigma}^t = \{\sigma_1^t, \sigma_2^t, \ldots, \sigma_M^t\}$. Since the cloud server is untrusted, the answers should be protected before being sent to the cloud server. Thus, each worker locally perturbs his answer with DP. Let $\hat{X}_i^t = \{\hat{x}_{i,1}^t, \hat{x}_{i,2}^t, \ldots, \hat{x}_{i,N}^t\}$ denote the perturbed answers of worker i, and $\hat{\boldsymbol{X}}^t = \{\hat{X}_1^t, \hat{X}_2^t, \ldots, \hat{X}_M^t\}$ denote the perturbed answer set of all workers. Based on the perturbed answer set $\hat{\boldsymbol{X}}^t$, the cloud server can only infer the perturbed truths, which are denoted as $\hat{Z}^t = \{\hat{z}_1^t, \hat{z}_2^t, \ldots, \hat{z}_N^t\}$.

Truth discovery [7], [30], [31] is an effective tool for estimating truths over answers from workers with different reliabilities. Although existing truth discovery algorithms have different solution processes, they follow the same workflow that iteratively conducts worker reliability estimation and truth computation until convergence. Here, we use a typical truth discovery algorithm, CRH [30], which has been used in a lot of privacy-preserving truth discovery researches [3], [14], [15], [16], [17], [18], [19], [20], [24], [25], [26]. Algorithm 1 describes the pseudo-code of truth discovery.

TABLE 1
Frequently Used Notations

Notation	Description
M, N	the number of workers and tasks, respectively
$x_{i,j}^t$, $\hat{x}_{i,j}^t$	the answer for task j from worker i at timestamp t before and after perturbation, respectively
X_i^t, $\hat{X}_i^t$	the set of all answers from worker i at timestamp t before and after perturbation, respectively
$\boldsymbol{X}^t$, $\hat{\boldsymbol{X}}^t$	the set of all answers from all workers at timestamp t before and after perturbation, respectively
w_i^t, $\hat{w}_i^t$	the estimated weight of worker i at timestamp t before and after perturbation, respectively
z_j^t, $\hat{z}_j^t$	the estimated truth of task j at timestamp t before and after perturbation, respectively
Z^t, $\hat{Z}^t$	the set of all estimated truths at timestamp t before and after perturbation, respectively
σ_i^t	the standard deviation of worker i's answer error at timestamp t
w	the size of sliding window
ϵ	the privacy budget
δ	the probability that a mechanism does not satisfy DP
τ	the domain of tasks' answers

Worker Weight Estimation. CRH initializes the truth with the average value of answers, then estimates the workers' weights $W^t = \{w_1^t, w_2^t, \ldots, w_M^t\}$ based on the current truths. The basic idea is that the smaller the distance between the answers and current truth is, the higher the weight of this worker will be. A worker's weight is formally defined as

$$w_i^t = -\log\left(\frac{\sum_{j=1}^N d(x_{i,j}^t, z_j^t)}{\sum_{i=1}^M \sum_{j=1}^N d(x_{i,j}^t, z_j^t)}\right), \tag{1}$$

where $d(\cdot)$ is the function measuring the distance between answer and truth, $x_{i,j}^t$ is the answer of worker i to task j at timestamp t, and z_j^t is the estimated truth of task j at timestamp t.

Truth Computation. In this step, workers' quality is assumed to be fixed, and the truth can be inferred through the weighted averaging strategy. Then, we have

$$z_j^t = \frac{\sum_{i=1}^M w_i^t \cdot x_{i,j}^t}{\sum_{i=1}^M w_i^t}. \tag{2}$$

This follows the principle that the answer from more reliable worker is closer to the truth. The workers' weights and truths are iteratively estimated until convergence.

Error Variance. After obtaining the truths, the error variances of workers can be estimated from the following equation.

$$\sigma_i^t = \sqrt{\frac{1}{N}\sum_{j=1}^N (d_{i,j}^t - \bar{d}_i^t)^2}. \tag{3}$$

where $d_{i,j}^t = d(x_{i,j}^t, z_j^t)$ is the error of worker i on task j, and

$\bar{d}_i^t = \frac{1}{N}\sum_{j=1}^{N} d(x_{i,j}^t, z_j^t)$ is the average error of worker i.

Algorithm 1. Truth Discovery (TD)

```
Input: Workers' answers $X^t$
Output: Estimated truths $Z^t$
Initialize truths $z_j^t = \frac{1}{M}\sum_{i=1}^{M} x_{i,j}$
while Not convergent do
    Estimate weights with Eq. (1)
    Estimate truths with Eq. (2)
end
```

2.2 Differential Privacy (DP) [32]

Definition 1 (ϵ-Differential Privacy). *If a mechanism $\mathcal{A}$ satisfies ϵ-differential privacy (ϵ-DP), for any output O of $\mathcal{A}$ and two neighbor answer vectors $X_{i,1}$ and $X_{i,2}$ of worker i that differ at one element, we have*

$$\Pr\{\mathcal{A}(X_{i,1}) = O\} \leq e^{\epsilon} \Pr\{\mathcal{A}(X_{i,2}) = O\}, \tag{4}$$

where Pr{·} denotes the probability of an event, and ϵ is the parameter to adjust the privacy protection level. A smaller ε corresponds to stronger privacy protection but more utility loss. A weaker version of ϵ-DP is (ϵ, δ)-DP, which satisfies ϵ-DP with probability at least $1-\delta$.

Among the mechanisms to achieve differential privacy, the widely used one is Laplace mechanism.

Definition 2 (Laplace Mechanism). *Let f: $X_i \to \mathbb{R}^N$, the Laplace mechanism is defined as*

$$\mathcal{A}(X_i) = f(X_i) + < Lap(\Delta_i/\epsilon_i) >^N, \tag{5}$$

where $f(\cdot)$ in truth discovery is the identity query, i.e., $f(X_i) = X_i$. $\Delta_i = \max||f(X_{i,1}) - f(X_{i,2})||_1 = \max||X_{i,1} - X_{i,2}||_1$ is the sensitivity, *which equals to τ (the domain of $|X_i|$), Lap(λ) is the Laplace distribution (centered at 0) with scale λ, where Lap$(x|\lambda) = \frac{1}{2\lambda}\exp(-\frac{|x|}{\lambda})$.*

Note that, in privacy-preserving statistics publishing, $f(\cdot)$ is a statistical query with a sensitivity of 1, which may be much smaller than that in truth discovery. Thus, its perturbation would be less than that in truth discovery under the same privacy protection level, especially when τ is very large.

2.3 w-event Differential Privacy

We set the stream prefix of an infinite series of any worker i at timestamp t as $S_i^t = (X_i^1, X_i^2, \ldots, X_i^t)$ [33]. We say two stream prefixes S_i^t and $S_i^{t'}$ are w-neighboring, if (1) for any $k \in [0, t]$ and $S_i^t[k] \neq S_i^{t'}[k]$, $S_i^t[k]$ and $S_i^{t'}[k]$ are neighboring; (2) for any $k_1 < k_2$, $S_i^t[k_1] \neq S_i^{t'}[k_1]$ and $S_i^t[k_2] \neq S_i^{t'}[k_2]$, there are $k_2 - k_1 + 1 < w$.

Definition 3 (w-event DP [34]). *A mechanism $\mathcal{A}$ satisfies w-event ϵ-DP, if for all sets $\mathbb{S} \subseteq$ Range(A) and all neighboring stream prefixes S_i^t and $S_i^{t'}$ of worker i and all t, it holds that*

$$\Pr\{\mathcal{A}(S_i^t) \in \mathbb{S}\} \leq e^{\epsilon} \Pr\{\mathcal{A}(S_i^{t'}) \in \mathbb{S}\}. \tag{6}$$

Similar to ϵ-DP, w-event ϵ-DP can also be relaxed to w-event (ϵ, δ)-DP if Eq. (6) holds with probability at least $1-\delta$. We simply call both w-event ϵ-DP and (ϵ, δ)-DP as w-event DP in the following. In addition, on the basis of sequential composition theorem [32], the w-event DP has the following property.

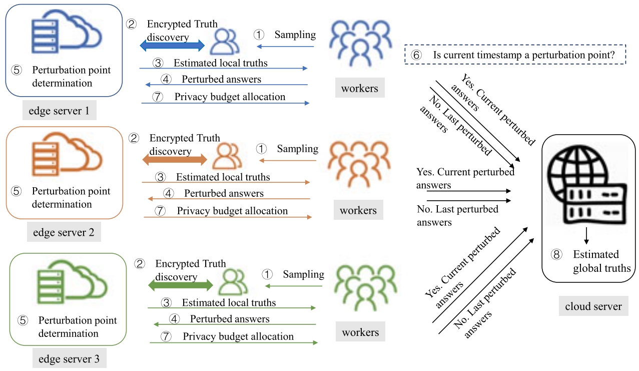

② Encrypted Truth discovery
① Sampling
③ Estimated local truths
④ Perturbed answers
⑦ Privacy budget allocation
⑤ Perturbation point determination
edge server 1
workers
edge server 2
edge server 3
⑥ Is current timestamp a perturbation point?
Yes. Current perturbed answers
No. Last perturbed answers
⑧ Estimated global truths
cloud server

Fig. 1. The framework of PrivSTD.

Theorem 1. *Suppose a mechanism $\mathbf{\mathcal{A}}$, which takes the set of stream prefix S_i^t of any worker i as input, can be decomposed into t mechanisms $A^1, A^2, \ldots, A^t$, and each A^t provides $(\epsilon_i^t, \delta_i^t)$-DP. Then* **A** *satisfies w-event DP if*

$$\sum_{k=\tau-w+1}^{\tau} \epsilon_i^k = \epsilon, \quad \sum_{k=\tau-w+1}^{\tau} \delta_i^k = \delta. \tag{7}$$

This theorem enables the w-event DP scheme to view ϵ, δ as the total available privacy budget in any sliding window of size w of any worker and reasonably allocate them to every timestamp.

3 Overview of PrivSTD

To protect the answers in streaming truth discovery, each worker perturbs their answers at the local. However, sine they know neither the changes of truths over time nor the effects of his/her answers on the truths before uploading the perturbed answers, they may introduce excessive noise to satisfy the overstrict DP guarantee, which would result in low accuracy of aggregated truth.

To improve the accuracy, we propose a novel edge computing based streaming data truth discovery algorithm for crowdsourcing system, called PrivSTD. It adaptively determines the perturbed timestamps according to the evolving pattern of truths and dynamically perturbs the chosen points on the basis of workers' reliabilities to realize high accuracy of discovered truths while protecting the privacy of workers. To achieve this, we introduce edge servers to help workers securely estimate local truths and workers' reliabilities. Fig. 1 shows the framework of PrivSTD, which mainly consists of three kinds of entities: worker, edge server and the cloud server.

Worker. As the participants in crowdsourcing tasks, workers will accept tasks and upload their answers to the cloud server. We assume that each worker honestly answers the tasks. To realize high accuracy of discovered truths while protecting the privacy, they securely interact with the edge server to determine the perturbation point and privacy budget. If current timestamp is determined as a perturbed point, they dynamically perturb the answers on the basis of their reliabilities and upload the perturbed answers to the cloud server, otherwise they upload the previous perturbed answers as replacements.

Edge Server. We consider that the edge server is honest-but-curious, and cannot collude with each other or the cloud. Its role is to determine the perturbation point and allocate privacy budget for workers. To realize this goal, it first samples a fraction of tasks from each worker to securely estimate local truths, based on which it adaptively determines the perturbation point and allocates the privacy budget of next timestamp for all the interacted workers according to the evolving pattern of truths.

Cloud Server. The cloud server is the crowdsourcing platform, which gathers perturbed answers from workers to calculate global truths. We assume that the cloud server is untrusted and cannot collude with edge servers and workers.

The detail interactions among workers, edge servers, and cloud server are described as follows:

① - ②: At each timestamp, the edge server first randomly samples a subset of tasks from each worker to estimate the local truths. To protect the privacy of workers, the local truths are calculated on the sampled answers through encrypted truth discovery, where workers' answers and weights are encrypted. The edge server can only obtain the plaintext of estimated local truth without knowing the workers' answers and weights. The specific process of encrypted truth discovery will be introduced in Section 4.1. Note that, it can also be used to protect the privacy of all answers. However, the encryption techniques (e.g., homomorphic encryption [3], garbled circuit [19]) have high computational overhead and poor scalability (The computation complexity will be analyzed in Section 4.3). Thus, it is only used on a fraction of tasks of each worker to estimate the local truths, based on which an elaborate DP algorithm is further proposed to protect the privacy of all the users in the truth discovery.

③ - ④: Once receiving the estimated local truths, the workers estimate their reliabilities, and perturb their answers with reliability-based perturbation mechanism (Section 4.2). The perturbed answers are then sent to the edge servers.

⑤ - ⑥ : Each edge server calculates the perturbed local truth based on the perturbed answers, and then determines the perturbation point according to the perturbed local truth and estimated local truths. If current timestamp is determined as a perturbed point, the workers interacted with this edge server upload the perturbed answers to the cloud server. Otherwise, they upload the previous perturbed answers as replacements. Note that, different edge servers may have different perturbation points.

⑦ - ⑧ : After that, the edge server allocates the privacy budgets of next timestamp for workers with the privacy budget allocation mechanism, which needs to satisfy the w-event DP. At last, the cloud server aggregates the perturbed answers to estimate global truths.

Note that, the cloud server can also find the global truths through aggregating the estimated or perturbed local truths from the edge server, but this method has two limitations: (1) the global truths calculated on the estimated local truths may have information loss because most workers' data is not counted; and (2) the global truths calculated on the perturbed local truths may have a greater error variance than that on the perturbed answers, as the former experiences twice truth discovery. Taking into account the two limitations, the cloud server aggregates the perturbed answers, instead of the local truths, to estimate global truths.

4 Design of PrivSTD

In this section, we first detail the design of PrivSTD and then theoretically analyze the privacy guarantee, expected error on truths, and the computation cost of PrivSTD. The object is to design a privacy-preserving streaming truth discovery framework with high accuracy while achieving w-event DP guarantee. The key idea is to leverage *the budget recycle mechanism* to explore the evolving pattern of truths to adaptively determine the perturbed timestamps and allocate privacy budget, and to utilize *the reliability-based perturbation mechanism* to perturb the perturbation points to minimize the total perturbation noise. The detailed procedure of PrivSTD is shown in Algorithm 2.

Algorithm 2. PrivSTD Algorithm at Timestamp t

```
Input: Current answer set $X^t$, the last perturbed local truth set $\hat{Z}_\pi^{t-1}$, the state set $S^t = \{s_\pi^t\}_{\pi=1}^{\Pi}$, the recycled number set $K^t = \{k_\pi^t\}_{\pi=1}^{\Pi}$, the privacy parameters $(\epsilon, \delta)$, the sliding window size $w$, the domain of answers $\tau$.
Output: The global perturbed truth set $\hat{Z}^t$, next state set $S^{t+1}$, next recycled number set $K^{t+1}$.
for edge server $\pi \in \Pi$ do
  if $s_\pi^t \neq canceled$ then
    $\epsilon_\pi^t = k_\pi^t \epsilon / w, \delta_\pi^t = k_\pi^t \delta / w, where\ k_\pi^t \in [1, w]$
    Generate the sampled answers set $\{x_{i,j}\}_{i \in M, j \in K}$
    // Secure Local Truth Estimation
    $\tilde{Z}_\pi^t, \boldsymbol{\sigma}_\pi^t = SLTE(U_s, \{x_{i,j}\}_{i \in M, j \in K})$
    for worker $i \in \pi$ do
      //Reliability-based perturbation
      $\Delta_i^t = \Phi^{-1}\left(\frac{2-\delta_\pi^t}{2}\right)\sqrt{2}\sigma_i^t, \hat{X}_i^t = X_i^t + Lap\left(\frac{min(\Delta_i^t, \tau)}{\epsilon_\pi^t}\right)$
    end
    $\hat{Z}_\pi^t = TD(\{\hat{X}_i^t\}_{i \in \pi})$
    // Perturbed point determination
    $E_a = |\hat{Z}_\pi^{t-1} - \tilde{Z}_\pi^t|, E_p = |\hat{Z}_\pi^t - \tilde{Z}_\pi^t|$
    if $E_a \geq E_p$ then
      Workers upload $\hat{X}_\pi^t$    // $t$ is a perturbed point
    end
    else
      Workers upload $\hat{X}_\pi^t = \hat{Z}_\pi^{t-1}$    // $t$ is an approximation point
    end
  end
  else
    Upload $\hat{Z}_\pi^t = \hat{Z}_\pi^{t-1}$
  end
  //Privacy budget allocation
  $S_\pi^{t+1}, K_\pi^{t+1} = PBA(S_\pi^t, K_\pi^t, w)$
end
for the cloud server do
  $\hat{Z}^t = TD(\{\hat{X}_i^t\}_{i \in M})$    // Global truth estimation
end
```

4.1 Budget Recycle Mechanism

In the streaming data truth discovery, we intend to use the w-event DP to provide privacy guarantee for streaming data. w-event DP can be realized by combining several DP mechanisms in many ways as long as they satisfy the composition property in Theorem 1. The most widely used combination method is the sampling mechanism [33], [34], [35], which reduces the overall perturbation through only perturbing answers at the sampled timestamps rather than at every timestamp.

Inspired by the sampling method in BA mechanism [34], we propose the budget recycle mechanism to sample a part of timestamps to conduct perturbation according to the evolving pattern of truths and allocate appropriate privacy budgets to workers, to reduce the amount of noise added to worker' submitted data, while simultaneously satisfying w-event DP. We call the sampled timestamps as perturbed points. To explore the evolving pattern of truths, we introduce edge servers to help workers to securely estimate local truths. Each edge server independently performs the budget recycle mechanism to sample the perturbed points and allocate appropriate privacy budgets to workers. The procedure of the budget recycle mechanism can be decomposed into three steps: secure local truth estimation, perturbed point determination and privacy budget allocation, which correspond to ①-②, ⑤-⑥, and ⑦, respectively.

4.1.1 Secure Local Truth Estimation

The local truths are estimated using the homomorphic Paillier encryption, which has been used in lots of works [3], [14], [17], [20]. Here we briefly introduce the process of encrypted truth discovery. More details can be found in reference [3].

We assume all the keys are given by a completely trusted key management center. Initially the key management center gives two large prime numbers p and q, and then calculates $n = pq$, $g = n + 1$, $\lambda = lcm(p-1, q-1)$. (g, n) are public keys that are distributed to the sample workers, and λ is private key that is sent to the cloud server. Each worker randomly samples a subset of tasks for local truth estimation. The cloud server then initializes the truths for the tasks.

Step 1: Secure Weight Update. Once receiving truths from the cloud server, each worker i calculates the distances between his/her sampled answers and truths to obtain $d_i^t = \sum_{j=1}^{K} d(x_{i,j}^t - z_j^t)$ (for convenience, we omit the superscript t that stands for timestamp in the following), where K is the number of sample tasks, and then selects a random value $r_i \in \mathbb{Z}_{n^2}$ to encrypt d_i as

$$E(d_i) = g^{d_i} {r_i}^n \mod n^2. \tag{8}$$

After receiving the ciphertexts from all sample workers, according to the homomorphic property, the edge server sums all the ciphertexts as

$$\begin{aligned} E_{sum} &= E\left(\sum_{i=1}^{M} d_i\right) = \prod_{i=1}^{M} E(d_i) \\ &= g^{\sum_{i=1}^{M} d_i} \left(\prod_{i=1}^{M} r_i\right)^n \mod n^2, \end{aligned} \tag{9}$$

Then it delivers the ciphertexts to the cloud server. After that, the cloud server executes the decryption function to get the plaintext of distance summations as

$$sum_d = \frac{L(E_{sum}^{\lambda} \mod n^2)}{L(g^{\lambda} \mod n^2)} \mod n, \tag{10}$$

where $L(x) = (x-1)/n$ is a function.

The proofs of Eqs. (9) and (10) can be found in reference [36]. sum_d is then sent to the workers for weight update according to Eq. (1) and further for computing weighted data for different tasks.

Step 2: Secure Truth Estimation. The workers' weights and the weighted data are also encrypted with Eq. (8), and the ciphertexts are then sent to the edge server for aggregation calculated by Eq. (9). After that, the edge server transmits the aggregation to the cloud server for decryption by Eq. (10). With the plaintexts of summations of weights and weighted data, the cloud server can easily estimated truths according to Eq. (2) and broadcasts them to the workers.

Step 1 and Step 2 are iteratively performed until convergence or a given number of iterations is reached. The protocol of secure local truth estimation is shown in Algorithm 3. During the iteration, the weights are computed on the worker side, and the answers and weights are protect by encryption. The untrusted edge server and cloud serve can only obtain the aggregation. Thus, workers' privacy will not be disclosed.

Algorithm 3. Secure Local Truth Estimation (SLTE)

```
Input: Two large prime numbers $p$ and $q$, $n = pq$, $\lambda = lcm(p-1, q-1)$, a random value $h$, Sampled workers $U_s$ and corresponding answers $\{x_{i,j}\}_{i \in M, j \in K}$
Output: Estimated local truths $\tilde{Z}^t$, standard error deviation set $\boldsymbol{\sigma}^t$.
The cloud server initializes truths $Z^t$
while Not convergent do
    //Secure weight update
    Each worker calculates $d_i^t = \sum_{j=1}^{K} d(x_{i,j}^t - z_j^t)$
    Each worker encrypts $d_i^t$ with Eq. (8)
    The edge server sums encrypted $d_i$ with Eq. (9)
    The cloud server decrypts the summations with Eq. (10)
    Each worker estimates weight $w_i^t$ with Eq. (1) and calculates weighted data $\{w_i^t \cdot x_{i,j}^t\}_{j=1}^{K}$
    //Secure truth estimation
    Each worker encrypts weight and weighted data with Eq. (8)
    The edge server respectively sums encrypted weight and weighted data with Eq. (9)
    The cloud server decrypts the summations of encrypted weight and weighted data with Eq. (10)
    The cloud server estimates truths $\tilde{Z}^t$ with Eq. (2)
end
Each worker estimates error deviation $\sigma_i^t$ with Eq. (3)
Return local truths $\tilde{Z}^t$, $\boldsymbol{\sigma}^t$
```

4.1.2 Perturbed Point Determination

The perturbed point is determined by each edge server independently, thus the workers at different edge server may have different perturbed points. For the πth edge server, we define the approximation error as $E_a = |\hat{Z}_{\pi}^{t-1} - \tilde{Z}_{\pi}^{t}|$, and the perturbation error as $E_p = |\hat{Z}_{\pi}^{t} - \tilde{Z}_{\pi}^{t}|$, where $\hat{Z}_{\pi}^{t-1}$ is the last perturbed local truths, $\hat{Z}_{\pi}^{t}$ is the current perturbed local truths calculated on perturbed answers $\hat{X}_{\pi}^{t}$, and $\tilde{Z}_{\pi}^{t}$ is the crypto-estimated local truths. We assume the number of workers is large enough and $\tilde{Z}_{\pi}^{t}$ is the unbiased estimation of local truth Z_{π}^{t}. If the approximation error E_a is no less than the perturbation error E_p (i.e., $E_a \geq E_p$), it means the perturbed local truths (as well as the perturbed global truths) that calculated on $\hat{X}_{\pi}^{t-1}$ will has a larger error than that calculated on $\hat{X}_{\pi}^{t}$. To reduce the error, we set timestamp t as a perturbed point and use $\hat{X}_{\pi}^{t}$ to compute the global truths. Otherwise, the timestamp t is an approximation point and we calculate the global truths with $\hat{X}_{\pi}^{t-1}$. Briefly, the current timestamp is determined as a perturbed point only when Eq. (11) is satisfied.

$$E_a \geq E_p. \tag{11}$$

Note that, the decision process of a perturbed point will not reveal private answers, because this decision determined by approximation error and perturbation error is computed on truths, which are insensitive. Although truths have complicated connections with private answers, an adversary still has difficulty in establishing specific relationship between the decision and the private answer due to the reliability-based perturbation and weighted aggregation. While in BA mechanism [34], the decision of a perturbed point will naturally leak some private information of the input data, because such decision is directly affected by the approximation error, which is computed on sensitive data. Thus BA must split a partial privacy budget to perturb the approximation error. Under the same total privacy budget, BA would incur more noise than PrivSTD, as the split privacy budget in BA is smaller than that in PrivSTD.

At each timpstamp t, Eq. (11) should be judged separately on each edge server. Based on the results, we divide the workers into two categories. The workers in the first category belong to the edge servers that satisfy Eq. (11), then the timestamp t is a perturbed point for workers in the first category. The remaining workers fall into the second category, for which the timestamp t is a non-perturbed point. Consequently, at each timpstamp t, the cloud server receives current perturbed answers from workers in the first category and the last perturbed answers from workers in the second category. With these perturbed answers, the cloud server further implements truth discovery algorithm in Algorithm 1 to obtain the global truths.

4.1.3 Privacy Budget Allocation

The privacy budget allocation is performed on each edge server under the privacy constraint of w-event DP. Initially, we uniformly allocates the privacy budget to each timestamp, where any worker interacting with edge server π at any timestamp t has the privacy budgets $\epsilon_{\pi}^{t} = k_{\pi}^{t}\epsilon/w$ and $\delta_{\pi}^{t} = k_{\pi}^{t}\delta/w$, where $k_{\pi}^{t} = 1$. However, workers only consume their privacy budgets at perturbed timestamps. The privacy budgets at non-perturbed timestamps can be recycled for the next perturbed timestamp. Thus, after the perturbed point determination, the privacy budget of the next timestamp of each worker should be adaptively adjusted according to the result of Eq. (11).

There are four cases to calculate the privacy budget of the next timestamp as shown in Algorithm 4. Take the edge server π as a example. When its state is *not canceled*, (1) if timestamp t is an approximation point, then $k_{\pi}^{t+1} = \min\{k_{\pi}^{t}+1, w\}$. (2) If timestamp t is a perturbed timestamp and $k_{\pi}^{t} > 1$, which means that the edge server π has recycled the privacy budgets at previous $k_{\pi}^{t}-1$ timestamps, its state would be set as *canceled* until $k_{\pi}^{t}-1$ timestamps later in order to satisfy the privacy constraint of w-event DP that the summary of privacy budgets within any w windows cannot exceed ϵ and δ (Recall the Theorem 1). When the state of edge server π is *canceled*, it first needs to determine whether its recycle number $k_{\pi}^{t} > 1$ or not. (3) If $k_{\pi}^{t} > 1$, its state is still *canceled* and k_{π}^{t} is reduced by 1. (4) If k_{π}^{t} is 1, its state would be set as *not canceled*.

Algorithm 4. Privacy Budget Allocation (PBA)

```
Input: the state set $S^t = \{s_{\pi}^{t}\}_{\pi=1}^{\Pi}$, the recycled number set
       $K^t = \{k_{i}^{t}\}_{\pi=1}^{\Pi}$, the sliding window size $w$
Output: Next state set $S^{t+1}$, next recycled number set $K^{t+1}$.
for each edge server $\pi$ do
    if $s_{\pi}^{t} \neq canceled$ then
        if $t$ is an approximation point then
            $k_{\pi}^{t+1} = \min\{k_{\pi}^{t}+1, w\}$   // Case 1
        end
        else
            if $k_{\pi}^{t} > 1$ then
                $s_{\pi}^{t+1} = canceled$   // Case 2
            end
        end
    end
    else
        if $k_{\pi}^{t} > 1$ then
            $k_{\pi}^{t+1} = k_{\pi}^{t} - 1$   // Case 3
        end
        if $k_{\pi}^{t} == 1$ then
            $s_{\pi}^{t+1} \neq canceled$   // Case 4
        end
    end
end
Return $S^{t+1}$, $K^{t+1}$
```

4.2 Reliability-Based Perturbation Mechanism

Although the budget recycle mechanism has a similar sampling method with BA mechanism, they still has a big difference in perturbation mechanism. BA uses the traditional Laplace mechanism mentioned in Definition 2 to protect the privacy of workers. It injects Laplace noise $Lap(\Delta_{i}^{t}/\epsilon_{i}^{t})$ to the query function $f(X_{i}^{t})$ of the ith worker at timestamp t, where the query function in DP is the identity query, i.e., $f(X_{i}^{t}) = X_{i}^{t}$, and Δ_{i}^{t} is the sensitivity of query function $f(\cdot)$, which is defined as the domain of X_{i}^{t}. For all workers, the sensitivity is the maximum domain of answers and may be very large for the continuous data. In order to reduce the perturbation noise, we propose the reliability-based perturbation mechanism to perturb the answers of each worker through dynamically adjusting the query function sensitivity of each worker according to their reliability.

The sensitivity of the identity query function $f(\cdot)$ of the ith worker at timestamp t is simply named as the sensitivity of worker i at timestamp t, computed by

$$\begin{aligned} \Delta_{i}^{t} &= max|X_{i,1}^{t} - X_{i,2}^{t}| \\ &= max|(X_{i,1}^{t} - Z^{t}) - (X_{i,2}^{t} - Z^{t})| \\ &= max|R_{i,1}^{t} - R_{i,2}^{t}|, \end{aligned} \tag{12}$$

where $X_{i,1}^t$ and $X_{i,2}^t$ are two neighboring datasets that differ at one element. Z^t is the truth set, and $R_{i,1}^t$ and $R_{i,2}^t$ are the error set corresponding to $X_{i,1}^t$ and $X_{i,2}^t$.

As we mentioned in Section 2.1, in truth discovery, the ith worker's errors R_i^t at timestamp t typically follow a normal distribution $N(0, (\sigma_i^t)^2)$, where a smaller variance $(\sigma_i^t)^2$ represents a higher degree of reliability. Since both $R_{i,1}^t$ and $R_{i,2}^t$ follow the normal distribution $N(0, (\sigma_i^t)^2)$, we have $R_{i,1}^t - R_{i,2}^t = X_{i,1}^t - X_{i,2}^t \sim N(0, 2(\sigma_i^t)^2)$. According to the property of normal distribution, there are

$$\Pr(|X_{i,1}^t - X_{i,2}^t| < K\sqrt{2}\sigma_i^t) = 2\Phi(K) - 1, \tag{13}$$

where $\Phi(\cdot)$ is the distribution function of standard normal distribution, and K is a variable that controls the range of sensitivity $\Delta_i^t = max|X_{i,1}^t - X_{i,2}^t|$. This equation means that at timestamp t, worker i has the maximum sensitivity $K\sqrt{2}\sigma_i^t$ with the probability of $2\Phi(K) - 1$.

In order to reduce the magnitude of Laplace noise, the sensitivity $K\sqrt{2}\sigma_i^t$ should be as small as possible, but smaller K corresponds to weaker privacy protection due to the smaller probability $2\Phi(K) - 1$. To achieve the trade-off between privacy and utility, we use privacy budget δ_i^t, where $\delta_i^t \in [0, 1]$, to control K. Specifically, we set $2\Phi(K) - 1 = 1 - \delta_i^t$, so $K = \Phi^{-1}(\frac{2-\delta_i^t}{2})$. As a result, with the probability of $1 - \delta_i^t$ the worker i has the maximum sensitivity

$$\Delta_i^t = \Phi^{-1}\left(\frac{2 - \delta_i^t}{2}\right)\sqrt{2}\sigma_i^t. \tag{14}$$

Δ_i^t is usually much smaller than the domain of $|X_i^t|$, which is denoted as τ. If Δ_i^t is higher than $domain(|X_i^t|)$, we choose τ as the ith worker's sensitivity. With the sensitivity Δ_i^t and privacy budget ϵ_i^t, the edge server then injects Laplace noise to the answers of corresponding worker i, which is given by

$$\hat{X}_i^t = X_i^t + Lap\left(\frac{min(\Delta_i^t, \tau)}{\epsilon_i^t}\right). \tag{15}$$

This method actually relaxes the privacy constraint of ϵ_i^t-DP into $(\epsilon_i^t, \delta_i^t)$-DP for higher utility. Moreover, when K has been determined, it is obvious that a more reliable worker has a smaller sensitivity Δ_i^t and a smaller perturbation due to the smaller σ_i^t. Thus, we can realize the goal of dynamically adding noise on the basis of workers' reliability to reduce the noise magnitude. Note that, although we add less perturbation to more reliable workers at a single perturbation timestamp, we prove that PrivSTD provide the same w-event (ϵ, δ)-DP guarantee for each worker in the whole stream.

4.3 Analysis

4.3.1 Privacy analysis

Theorem 2. *PrivSTD satisfies w-event (ϵ, δ)-DP.*

Proof. We first prove that the reliability-based perturbation mechanism on any worker i satisfies (ϵ^t, δ^t)-DP. We omit i for simplicity in the following proof. As mentioned in the above, $|X_1^t - X_2^t| < \Phi^{-1}(\frac{2-\delta^t}{2})\sqrt{2}\sigma^t$ is with the probability $1 - \delta^t$, and we choose $\Delta^t = \Phi^{-1}(\frac{2-\delta^t}{2})\sqrt{2}\sigma^t$ as the sensitivity of Laplace mechanism, so we have

$$\begin{aligned}
&\Pr\{A(X_1^t) = O\} = \frac{\epsilon^t}{2\Delta^t}\exp\left(-\frac{\epsilon^t|X_1^t - O|}{\Delta^t}\right)\\
&= \frac{\epsilon^t}{2\Delta^t}\exp\left(-\frac{\epsilon^t|X_2^t + X_1^t - X_2^t - O|}{\Delta^t}\right)\\
&\le \exp\left(\frac{\epsilon^t|X_1^t - X_2^t|}{\Delta^t}\right)\frac{\epsilon^t}{2\Delta^t}\exp\left(-\frac{\epsilon^t|X_2^t - O|}{\Delta^t}\right)\\
&\le \exp\left(\frac{\epsilon^t|X_1^t - X_2^t|}{\Delta^t}\right)\Pr\{A(X_2^t) = O\}.
\end{aligned} \tag{16}$$

Since $|X_1^t - X_2^t| < \Delta^t$ is with the probability of $1 - \delta^t$, we have $\exp(\frac{\epsilon^t|X_1^t - X_2^t|}{\Delta^t}) \le e^{\epsilon^t}$ with the probability of $1 - \delta^t$. That means, Eq. (16) satisfies ϵ^t-DP with the probability of $1 - \delta^t$. Therefore, according to Definition 1, the reliability-based perturbation mechanism satisfies (ϵ^t, δ^t)-difference privacy.

As for the budget recycle mechanism, the privacy budget at every timestamp is either $k\epsilon/w$ with $k\delta/w$ ($k \in [1, w]$) or zero. For any perturbed point with an allocated budget of $k\epsilon/w$, it recycles the privacy budget in the previous $k - 1$ timestamps, and cancels the privacy budget in the following $k - 1$ timestamps. Thus, its previous $k - 1$ timestamps and following $k - 1$ timestamps all have zero allocated privacy budget. As a result, the sum of budgets of any w successive timestamps satisfies $0 \le \sum_{k=\tau-w+1}^{\tau} \epsilon^k \le \epsilon$. Based on Theorem 1, the budget recycle mechanism satisfies w-event (ϵ, δ)-differential privacy.

The subsequence truth estimation is executed on the perturbed answers, which would not disclose users' privacy. We treat the truth discovery operations as mapping functions on perturbed answers. According to the post-processing invariant of DP [32], it also satisfies w-event (ϵ, δ)-DP. Thus, PrivSTD satisfies w-event (ϵ, δ)-DP. □

4.3.2 Utility Analysis

Theorem 3. *Assume a streaming answer set $\boldsymbol{X} = \{\boldsymbol{X}^1, \boldsymbol{X}^2, \ldots, \boldsymbol{X}^T\}$, the streaming truth sets calculated by the cloud server with and without PrivSTD over the answer set $\boldsymbol{X}$ are denoted as $\boldsymbol{Z} = \{Z^1, Z^2, \ldots, Z^T\}$ and $\hat{\boldsymbol{Z}} = \{\hat{Z}^1, \hat{Z}^2, \ldots, \hat{Z}^T\}$, respectively. Given the differential privacy parameters ϵ, δ, and sliding window size w, the expectation of the mean absolute error between $\boldsymbol{Z}$ and $\hat{\boldsymbol{Z}}$ satisfies*

$$\mathbb{E}[MAE(Z, \hat{Z})] \le$$
$$\frac{1}{2k-1}\left(\sum_{\alpha=1}^{k} E\left(\frac{\alpha\epsilon}{w}, \frac{\alpha\delta}{w}\right) + (k-1)E\left(\frac{k\epsilon}{w}, \frac{k\delta}{w}\right) + \sum_{b=1}^{k-1} b|\bar{\lambda}|\right).$$

where $E(\frac{\alpha\epsilon}{w}, \frac{\alpha\delta}{w}) \le \frac{\sum_{i'=1}^{M}\sum_{i=1}^{M}\left(((\bar{\sigma}_i)^2 + (\bar{\sigma}_{i'})^2)\sqrt{\frac{2}{\pi}} + \frac{\Phi^{-1}\left(\frac{2-\alpha\delta/w}{2}\right)\sqrt{2}\bar{\sigma}_{i'}}{\alpha\epsilon/w}\right)}{M^2}$

is the expectation function at a single timestamp that perturbed with privacy budget $\frac{\alpha\epsilon}{w}$ and $\frac{\alpha\delta}{w}$. $\bar{\sigma}_i$ and $\bar{\sigma}_i^{'}$ are the average error's variances of worker i and $i^{'}$, respectively. k is the average recycled number of privacy budget for a perturbed point. $|\bar{\lambda}|$ is the average change rate of answers.

Proof. As we analyzed in Section 4.1, workers in each timestamp have three statuses: recycling, perturbing and canceling. Each perturbing point with privacy budget $k\epsilon/w$ and $k\delta/w$ must have $k-1$ recycling points in the front and $k-1$ canceled points in the behind. In order to simplify the problem, we assume all the workers at the same timestamp are in the same status.

Perturbing. We first calculate the expectation of error at a single perturbed timestamp t with privacy budget $k\epsilon/w$ and $k\delta/w$, $k \in [1, w]$.

$$\mathbb{E}[MAE(Z^t, \hat{Z}^t)] = \mathbb{E}\left[\frac{1}{N}\sum_{j=1}^{N} |z_j^t - \hat{z}_j^t|\right] \tag{17}$$

$$= \mathbb{E}\left[\frac{1}{N}\sum_{j=1}^{N} |\frac{\sum_{i=1}^{M} w_i^t x_{i,j}^t}{\sum_{i=1}^{M} w_i^t} - \frac{\sum_{i=1}^{M} \hat{w}_i^t \hat{x}_{i,j}^t}{\sum_{i=1}^{M} \hat{w}_i^t}|\right] \tag{18}$$

$$\leq \mathbb{E}\left[\frac{\sum_{i'=1}^{M}\sum_{i=1}^{M} \hat{w}_{i'}^t w_i^t \left(\frac{1}{N}\sum_{j=1}^{N} |x_{i,j}^t - \hat{x}_{i',j}^t|\right)}{\sum_{i'=1}^{M}\sum_{i=1}^{M} \hat{w}_{i'}^t w_i^t}\right] \tag{19}$$

$$\leq \frac{\sum_{i'=1}^{M}\sum_{i=1}^{M} \left(\frac{1}{N}\sum_{j=1}^{N} \mathbb{E}[|x_{i,j}^t - \hat{x}_{i',j}^t|]\right)}{M^2}. \tag{20}$$

where M is the number of total workers and N is the number of total tasks.

Eq. (20) has been proved in Lemma 4.4 of [26]. We assume the errors of workers i and i' are $x_{i,j}^t - z_j^t \sim N(0, (\bar{\sigma}_i)^2)$ and $\hat{x}_{i',j}^t - z_j^t \sim N(0, (\bar{\sigma}_{i'})^2)$, where $(\bar{\sigma}_i)^2$ and $(\bar{\sigma}_{i'})^2$ represent the average variances of worker i and i' over all the timestamps, respectively. Then, we have

$$\begin{aligned}
&\frac{1}{N}\sum_{j=1}^{N} \mathbb{E}[|x_{i,j}^t - \hat{x}_{i',j}^t|] = \mathbb{E}[|x_{i,j}^t - \hat{x}_{i',j}^t|] \\
&= \mathbb{E}[|(x_{i,j}^t - z_j^t) - (x_{i',j}^t - z_j^t + Lap(\Delta_{i'}^t/\epsilon_{i'}^t))|] \\
&= \mathbb{E}[|N(0, (\bar{\sigma}_i)^2) - (N(0, (\bar{\sigma}_{i'})^2) + Lap(\Delta_{i'}^t/\epsilon_{i'}^t))|] \\
&\leq ((\bar{\sigma}_i)^2 + (\bar{\sigma}_{i'})^2)\sqrt{\frac{2}{\pi}} + \frac{\Delta_{i'}^t}{\epsilon_{i'}^t}.
\end{aligned} \tag{21}$$

Substitute Eq. (20) with Eqs. (21) and (14), there are

$$\begin{aligned}
&\mathbb{E}[MAE(Z^t, \hat{Z}^t)] \\
&\leq \frac{\sum_{i'=1}^{M}\sum_{i=1}^{M} \left(((\bar{\sigma}_i)^2 + (\bar{\sigma}_{i'})^2)\sqrt{\frac{2}{\pi}} + \frac{\Phi^{-1}(\frac{2-k\delta/w}{2})\sqrt{2}\bar{\sigma}_{i'}}{k\epsilon/w}\right)}{M^2}.
\end{aligned} \tag{22}$$

Obviously, the expectation of error at a single perturbed point is a function of differential privacy parameters, thus we use a function $E(\frac{k\epsilon}{w}, \frac{k\delta}{w})$ to denote $\mathbb{E}[MAE(Z^t, \hat{Z}^t)]$.

Recycling. Since timestamp t is a perturbed point, it must have $k-1$ approximation points in the front. The first approximated timestamp $t-k+1$ cannot have a larger approximation error than $E(\frac{\epsilon}{w}, \frac{\delta}{w})$, otherwise it would not have been approximated. The second approximated timestamp $t-k+2$ must have a smaller approximation error than $E(\frac{2\epsilon}{w}, \frac{2\delta}{w})$. In a similar fashion, the timestamp $t-1$ induces an approximation error smaller than $E(\frac{(k-1)\epsilon}{w}, \frac{(k-1)\delta}{w})$. The total error in the $k-1$ approximation points must be less than $\sum_{\alpha=1}^{k-1} E(\frac{\alpha\epsilon}{w}, \frac{\alpha\delta}{w})$.

Canceling. For the canceled $k-1$ timestamps succeeding timestamp t, we directly use the last perturbed answers to approximate the current answers. In this case, the approximation error is determined by the change rate of answers. We assume that the change rate of task j from worker i is $\lambda_{i,j}$. Thus the expectation approximation error at a single point $t+1$ is

$$\begin{aligned}
\mathbb{E}[MAE(Z^{t+1}, \hat{Z}^t)] &= \mathbb{E}\left[\frac{1}{N}\sum_{j=1}^{N} |z_j^{t+1} - \hat{z}_j^t|\right] \\
&\leq \frac{\sum_{i'=1}^{M}\sum_{i=1}^{M} \left(\frac{1}{N}\sum_{j=1}^{N} \mathbb{E}\left[|x_{i,j}^{t+1} - \hat{x}_{i',j}^t|\right]\right)}{M^2} \\
&= \frac{\sum_{i'=1}^{M}\sum_{i=1}^{M} \left(\frac{1}{N}\sum_{j=1}^{N} \mathbb{E}\left[|x_{i,j}^t + \lambda_{i,j} - \hat{x}_{i',j}^t|\right]\right)}{M^2} \\
&= E\left(\frac{k\epsilon}{w}, \frac{k\delta}{w}\right) + \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} |\lambda_{i,j}|}{MN} \\
&= E\left(\frac{k\epsilon}{w}, \frac{k\delta}{w}\right) + |\bar{\lambda}|,
\end{aligned} \tag{23}$$

where $\bar{\lambda}$ is the average change rate. In a similar fashion, the expectation error of the canceled point $t+2$ is $E(\frac{k\epsilon}{w}, \frac{k\delta}{w}) + 2|\bar{\lambda}|$, and that in point $t+k-1$ is $E(\frac{k\epsilon}{w}, \frac{k\delta}{w}) + (k-1)|\bar{\lambda}|$. The total error in the $k-1$ canceled points are $(k-1)E(\frac{k\epsilon}{w}, \frac{k\delta}{w}) + \sum_{b=1}^{k-1} b|\bar{\lambda}|$.

Combining the three stages, the average error above the $2k-1$ timestamps is at most $\frac{1}{2k-1}(\sum_{\alpha=1}^{k} E(\frac{\alpha\epsilon}{w}, \frac{\alpha\delta}{w}) + (k-1)E(\frac{k\epsilon}{w}, \frac{k\delta}{w}) + \sum_{b=1}^{k-1} b|\bar{\lambda}|)$. Without loss of generality, we assume that k is the average recycled number of privacy budget for a perturbed point, thus the expectation of mean absolute error of streaming truth set satisfies

$$\begin{aligned}
&\mathbb{E}[MAE(\boldsymbol{Z}, \hat{\boldsymbol{Z}})] \\
&\leq \frac{1}{2k-1}\left(\sum_{\alpha=1}^{k} E\left(\frac{\alpha\epsilon}{w}, \frac{\alpha\delta}{w}\right) + (k-1)E\left(\frac{k\epsilon}{w}, \frac{k\delta}{w}\right) + \sum_{b=1}^{k-1} b|\bar{\lambda}|\right).
\end{aligned} \tag{24}$$

From the above theoretical analysis, it can be seen that the error expectation has relationships with the average change rate of answers λ, the privacy parameters ϵ and δ, and the sliding window size w. To verify this, we evaluate these parameters on synthetic datasets in Section 5. □

4.3.3 Computation Cost Analysis

Here, we compare the computation cost of PrivSTD and traditional homomorphic encryption in reference [17] on the worker side, edge server, and cloud server, respectively.

TABLE 2
Computation Cost Analysis

Entities \ Methods	PrivSTD		Homomorphic encryption [17]
	encryption	perturbation	
Worker	αN EXP	N ADD	N EXP
Edge server	$\alpha N M_\pi$ MUL	$O((N+1)M_\pi)$	NM_π MUL
Cloud server	$\vert\pi\vert N$ EXP	$O(MN)$	$\vert\pi\vert N$ EXP and $O(\vert\pi\vert N)$

EXP means one exponentiation operation, MUL denotes a multiplication operation, and ADD means an additive operation. α is the sampling rate of tasks in the secure local truth estimation, M is the number of workers, N is the number of tasks, M_π is the number of workers that interacted with edge server π, and $|\pi|$ is the number of edge servers.

PrivSTD first samples a fraction of tasks from each worker to securely estimate the local truths, based on which an elaborate DP algorithm is further proposed to perturb all the answers, while the traditional homomorphic encryption securely estimates the global truths on the whole answers without perturbation. Summary of the comparison results are presented in Table 2. We use EXP to denote the one exponentiation operation, MUL to denote one multiplication operation and ADD to denote an additive operation. We assume there are $|\pi|$ edge servers and one cloud server. α is the sampling rate of tasks in the secure local truth estimation, N is the number of tasks, M is the number of workers, and M_π is the number of worker that interacted with edge server π.

On the worker side, PrivSTD requires each worker i to encrypt the answers of sampled tasks in the secure local truth estimation, which needs αN EXP. Note that, we ignore the number of iterations, because it is usually much smaller than the number of tasks. Then each worker needs to perturb the answers in the perturbed global truth estimation, of which the cost of each worker is N ADD. While for the traditional homomorphic encryption, it only encrypts all the answers without perturbation, thus it costs N EXP.

On the edge server, PrivSTD first needs to aggregate the ciphertexts from interacted workers for each task in the secure local truth estimation, hence the cost is $N\bar{M}$ MUL, where $\bar{M}_\pi$ is the average number of workers per task at the edge server π, i.e., $\bar{M}_\pi = \alpha M_\pi$. Then in the perturbed global truth estimation, it should calculate the perturbed local truths and allocate privacy budgets, of which the running time are linear with the number of answers, i.e., $O(NM_\pi)$, and $O(M_\pi)$, respectively. While for the traditional homomorphic encryption, the computation cost of aggregating the ciphertexts is NM_π MUL.

On the cloud server, both PrivSTD and traditional homomorphic encryption decrypt the ciphertexts from edge server in the secure local truth estimation, which cost $|\pi|N$ EXP. In the perturbed global truth estimation, PrivSTD implements the truth discovery algorithm on all perturbed answers to estimate the global truths, of which the time complexity is $O(MN)$, while the traditional homomorphic encryption estimates global truths on the local truths, thus its time complexity is $O(|\pi|N)$.

From the comparison results in Table 2, we can see that PrivSTD reduces the computation cost on the worker side from N EXP to αN EXP through sampling. We also show their scalability on the worker side in Fig. 2. The result shows that PrivSTD is much more scalable than traditional homomorphic encryption.

5 Performance Evaluation

In this section, we evaluate the performance of PrivSTD on synthetic datasets under different parameter settings and demonstrate the superiority of PrivSTD by comparing it with two state-of-the-art approaches on three real-world datasets.

5.1 Datasets and Experiment Setup

Synthetic Datasets. The number of total tasks is set as 100. We first generate the ground truth of task j by following a sine function $z_j^t = 10\sin(\omega t) + \phi_j$, where ϕ_j is the initial phase of task j that sampled from a uniform distribution $U(0,5)$, ω is the smooth factor that controls the smooth level of truth, and t is the timestamp that changes from 1 to 100. Note that, each ω corresponds to a streaming dataset. We simulate 100 workers with different reliabilities $(\sigma_i^t)^2$ for every timestamp, which is randomly sampled from 1 and 3. As mentioned early, the error of worker i follows $N(0,(\sigma_i^t)^2)$. Thus, we can generate the answers for every worker at every timestamp by adding Gaussian noise $N(0,(\sigma_i^t)^2)$ into the corresponding ground truth.

Real-World Datasets. We use two real-world datasets,[4] including flight and weather, for performance evaluation, which are described as follows.

- Flight. We extract the departure information for 1,200 flights from 38 sources every day in December 2011. All the departure information is transformed into minutes (i.e., 7:00 is transformed into 420), so the domain of answers is 1,440.
- Weather. The weather data is collected in 30 major USA cities from 18 websites every 45 minutes on a day in March 2010. We extract the temperature information as answers, of which the maximum value is 74, so the domain is 74.

Experiment Setup. We adopt the CRH [30] algorithm to iteratively calculate the truths and workers' weights as Eqs. (1) and (2). We follow the previous works [4], [37] to use the Mean Absolution Error (MAE), Root of Mean Squared Error (RMSE) and Mean Relative Error (MRE) as the utility metrics. It is worthy to note that the errors are the distances between calculated perturbed truths and calculated unperturbed truths, which are defined as follows.

$$MAE(\mathbf{Z}, \hat{\mathbf{Z}}) = \frac{1}{T \times N} \sum_{t=1}^{T} \sum_{j=1}^{N} |\hat{z}_j^t - z_j^t|, \tag{25}$$

[4] http://lunadong.com/fusionDataSets.htm

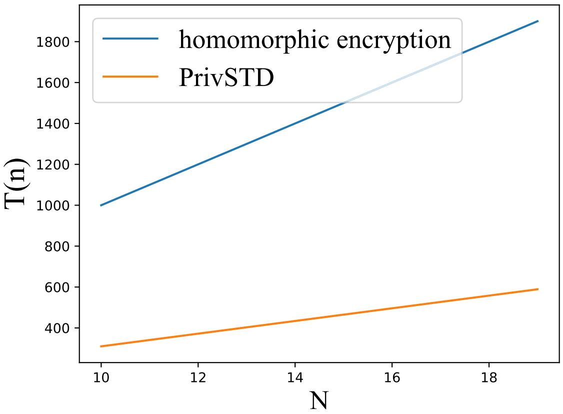


Fig. 2. Comparison of computation cost on the worker side. The sampling rate α of PrivSTD is set as 0.3.

$$RMSE(\mathbf{Z}, \hat{\mathbf{Z}}) = \sqrt{\frac{1}{T \times N} \sum_{t=1}^{T} \sum_{j=1}^{N} (\hat{z}_j^t - z_j^t)^2}, \tag{26}$$

$$MRE(\mathbf{Z}, \hat{\mathbf{Z}}) = \frac{1}{T \times N} \sum_{t=1}^{T} \sum_{j=1}^{N} \frac{|\hat{z}_j^t - z_j^t|}{\max(z_j^t, \gamma)}, \tag{27}$$

where $\mathbf{Z}$ and $\hat{\mathbf{Z}}$ are the estimated truth sets without and with perturbation, respectively. γ is a bound to mitigate the effect of excessively small value. T is the total number of timestamps. We execute all experiments 100 times and take the average results.

Mechanisms for Comparison. In order to verify the performance of PrivSTD, we compare it with two benchmarks [34] and three state-of-the-art approaches. The two benchmarks are traditional privacy-preserving truth discovery methods that do not consider the reliabilities of workers. The first is *Uniform*, which evenly allocates the privacy budget ϵ to every timestamp, and injects to answers the same Laplace noise scaled by $\lambda = \Delta w/\epsilon$, where Δ is the domain of answer and w is the size of sliding window. The second is *Sample*, which samples a portion of timestamps to inject noise. Specifically, we set the sample rate as $1/w$, which means that we only sample one timestamp in a sliding window to inject noise. Without loss of generality, we perturb the answers at the tth timestamp Laplace noise scaled by $\lambda = \Delta/\epsilon$ if $t \bmod w = 1$.

The three state-of-the-arts are the truth discovery with matrix factorization based on DP [24] (we call it TD-MF for simplification), Paris-TD [27], and BA [34], Since TD-MF and Paris-TD are designed for a one-time dataset, we perform them at every timestamp and allocate to them the same privacy budget as *Uniform* to satisfy w-event different privacy. The BA mechanism is modified to fit our streaming data truth discovery scenario, which has the same budge recycle mechanism with PrivSTD. But it uses the traditional Laplace mechanism to perturb answers because it does not consider workers' reliability. Note that, the comparison among *Uniform* and *Sample* is to evaluate the performance of adaptive privacy budget allocation. The comparison between PrivSTD and BA is to show the effectiveness of the reliability-based perturbation mechanism, and the comparison of PrivSTD to TD-MF and Paris-TD is to exhibit the performance of our privacy-preserving truth discovery framework.

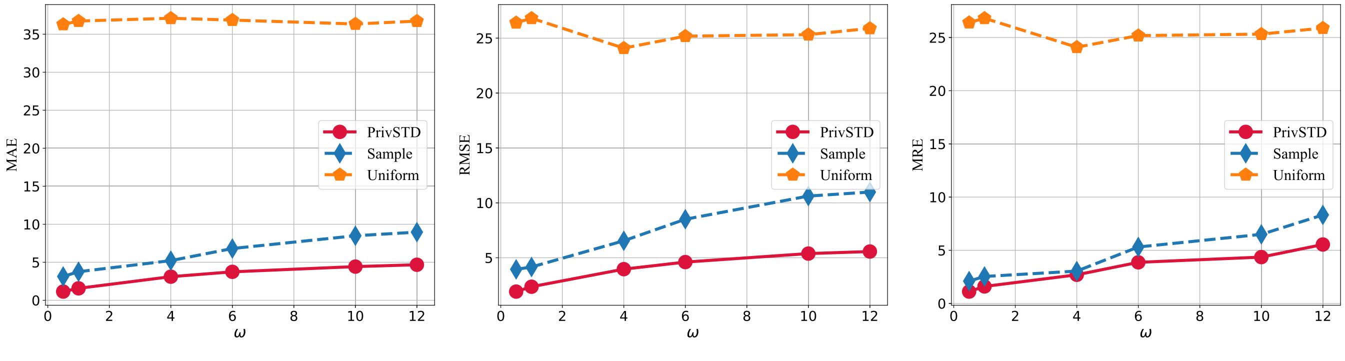
5.2 Experiments on Synthetic Dataset

In this subsection, we evaluate the effects of parameters, including the smooth level of truth, privacy budget, the number of the edge servers and the sliding window size, on the performance of PrivSTD, and the utility-privacy tradeoff on the synthetic datasets.

Effect of Smooth Level. PrivSTD adaptively perturbs answers according to the evolving pattern of truths in adjacent timestamps, which is influenced by the smooth level of truth. In the synthetic stream dataset, we control the smooth level by parameter ω. The smaller the ω is, the smoother the truths are. We set ϵ=1, δ=0.02, the sampling rate of tasks during local truth estimation $\alpha = 0.3$, sliding window size $w = 10$ and the number of edge servers as 5. The MAEs of truth and answer under various ω are shown in Fig. 3.

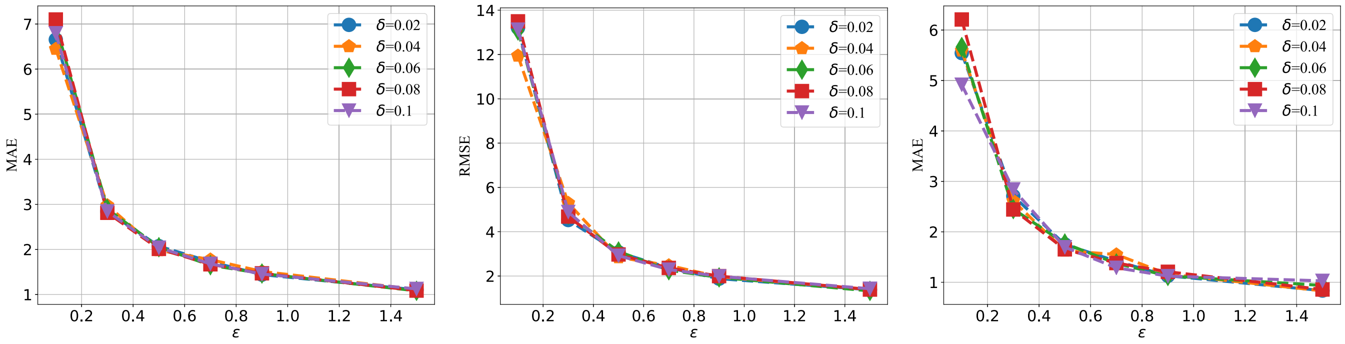
As shown in Fig. 3, we can see that PrivSTD always achieves the smallest MAE, RMSE and MRE compared to the two baseline approaches, and the *Uniform* has the worst performance. This is because *Uniform* adds the maximum perturbation to each worker at each timestamp. We also observe that *Uniform* is robust to the change of smooth factor ω, while PrivSTD and *Sample* slightly grow as ω increases. The reason is that *Uniform* only has perturbation errors, which have no relationship with ω, while PrivSTD and *Sample* have approximation errors at the non-perturbed points, which will increase if the truths' evolution becomes steep.

Effect of Privacy Parameters. We then verify the effects of privacy parameters (ϵ, δ) of PrivSTD. We set $\omega = 1$, the sampling rate of tasks during local truth estimation $\alpha = 0.3$, the



(a) MAE (b) RMSE (c) MRE

Fig. 3. Effect of the smooth factor ω on the synthetic dataset.



(a) MAE (b) RMSE (c) MRE

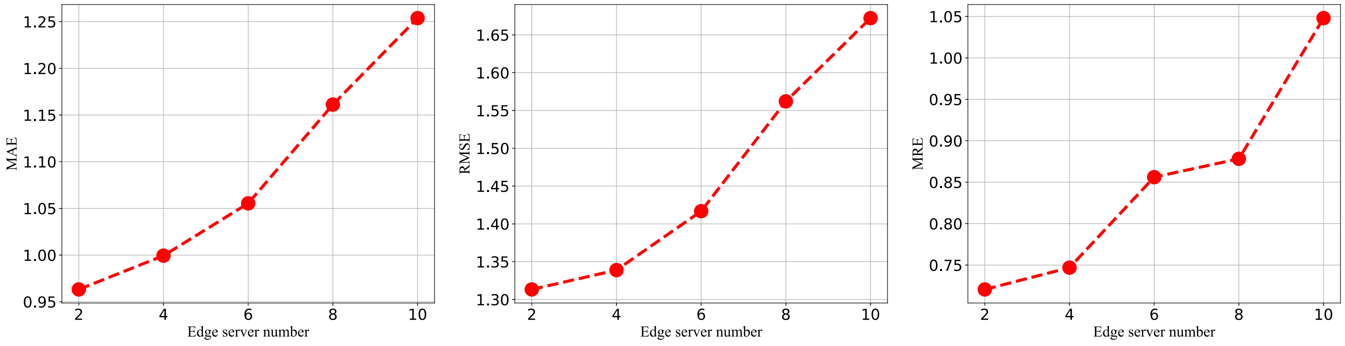
Fig. 4. Effect of privacy parameters (ϵ, δ) on synthetic dataset.

sliding window size $w = 10$ and the number of edge server as 5, the results under various ϵ and δ are shown in Fig. 4, where δ varies from 0.02 to 0.1 with a step of 0.02.

As shown in Fig. 4, we can see that the error of PrivSTD decreases with the increment of ϵ, as less perturbation is added for a larger value of ϵ. Another important observation is that there MAE, RMSE and MRE are almost the same under different δ. This is because that δ has little effect on perturbation comparing with the worker's reliability according to Eq. (14). As a result, we can choose a small δ to get a higher privacy protection level.

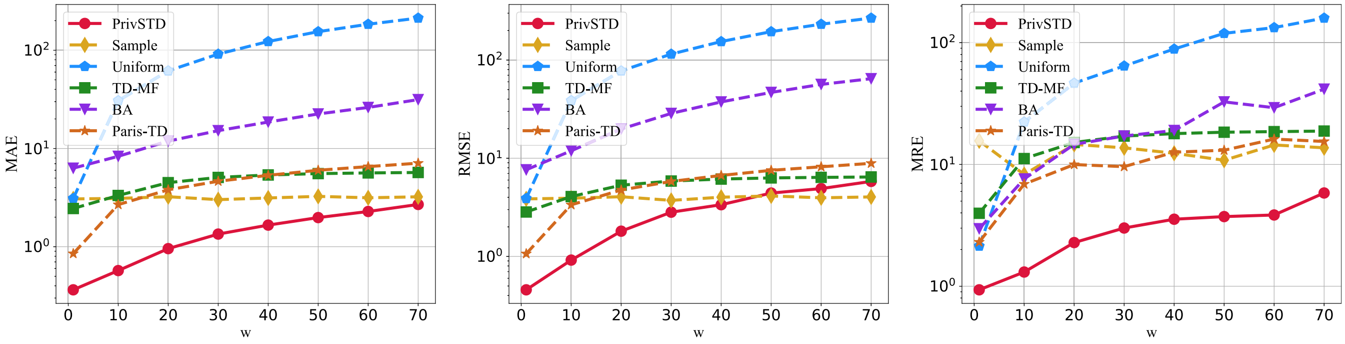
Effect of Edge Server Number. To evaluate the effect of edge server number on the performance of PrivSTD, we measure the errors under the different number of edge servers with $\epsilon = 1, \delta = 0.02, \omega = 1$, the sampling rate of tasks during local truth estimation $\alpha = 0.3$, and $w = 5$. From Fig. 5, it can be seen that the three metrics increase with the increasing number of edge servers. This is because when the number of edge servers increases, the received number of workers for every edge server decreases. As a result, the estimated local truths would have larger bias due to the reduced sampled tasks. Thus, the global truths have greater errors. If the worker number is large enough, the local truths will be unbiased estimations of global truths.

Effect of Sliding Window Size. Fig. 6 shows the effect of sliding window size w on the performance of PrivSTD. The experiments are performed under $\epsilon = 1, \delta = 0.02, \omega = 0.1$, the sampling rate of tasks during local truth estimation $\alpha = 0.3$, and the number of edge servers as 5. From this figure we can observe that (1) PrivSTD always has smaller error than



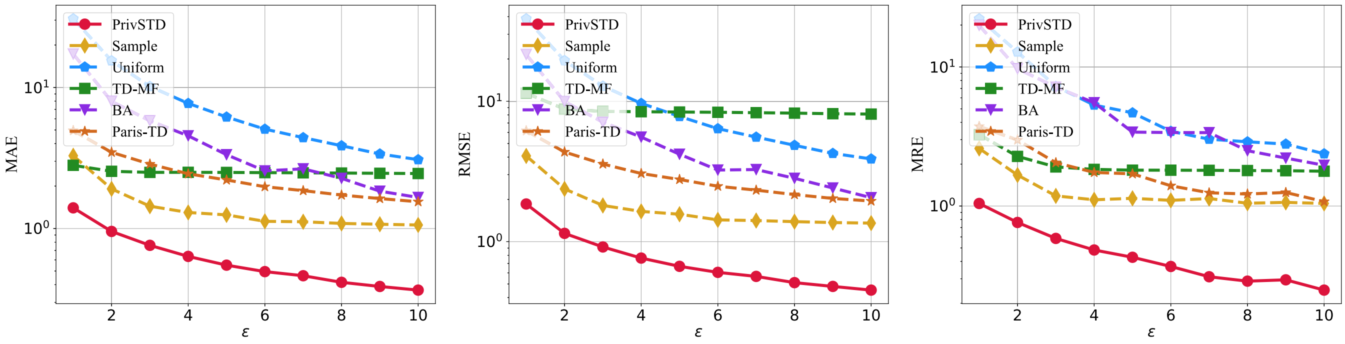
(a) MAE (b) RMSE (c) MRE

Fig. 5. Effect of edge server number on the synthetic dataset.



(a) MAE (b) RMSE (c) MRE

Fig. 6. Effect of sliding window size w on the synthetic dataset.

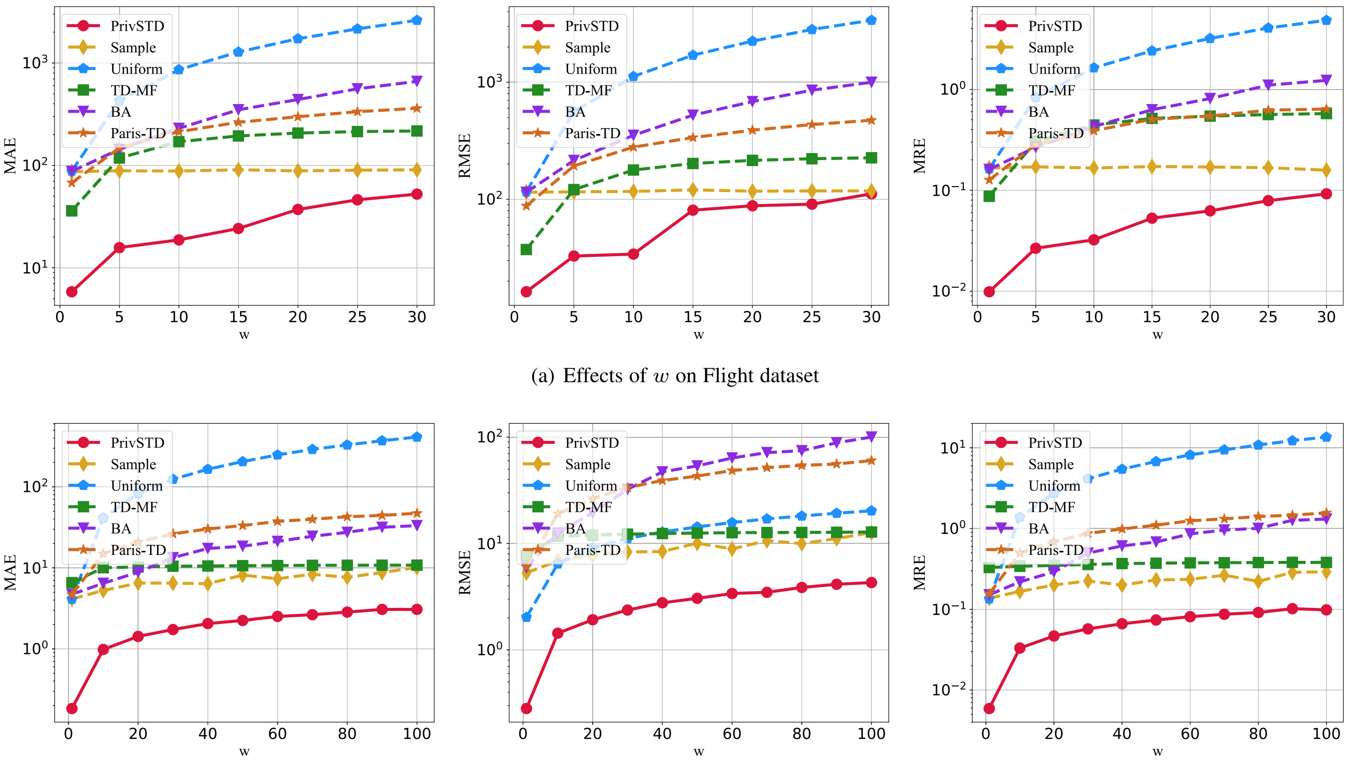


(a) MAE (b) RMSE (c) MRE

Fig. 7. Effect of privacy budget ϵ on the synthetic dataset.

TD-MF, Paris-TD, and BA even when w is very small (e.g., $w = 1$, equivalent to one-time dataset), which demonstrates the effectiveness of the budget recycle mechanism and the reliability-based perturbation mechanism. (2) The errors of all the mechanisms except *Sample* increase with the increment of sliding window size w. This is because when total privacy privacy is fixed, the larger the w, the smaller the privacy budget allocated to each event due to ϵ/w, and thus more noise would be introduced into data, resulting in the worse utility. It indicates that the privacy level of DP for streaming data is determined by both sliding window size w and total privacy budget ϵ. The choice of parameters w and ϵ needs to be balanced according to the trade-off between utility and privacy. (3) When sliding window size w is sufficiently large, PrivSTD performs worse than *Sample*. This is because the perturbation error of *Sample* is independent of sliding window size w, while that of PrivSTD will increase with w due to the decreasing privacy budget ϵ/w. Although the budget recycle mechanism and the reliability-based perturbation mechanism of PrivSTD can reduce the errors to some extent, they cannot cancel out the excessive errors caused by the initial large random noise $Lap(\frac{\Delta}{\epsilon/w})$. Nevertheless, it should be noted that, a large sliding window for w-DP is a very extreme situation, since it would bring excessive noise and thus invalidate data utility. In general cases, PrivSTD can achieve better utility than *Sample*.

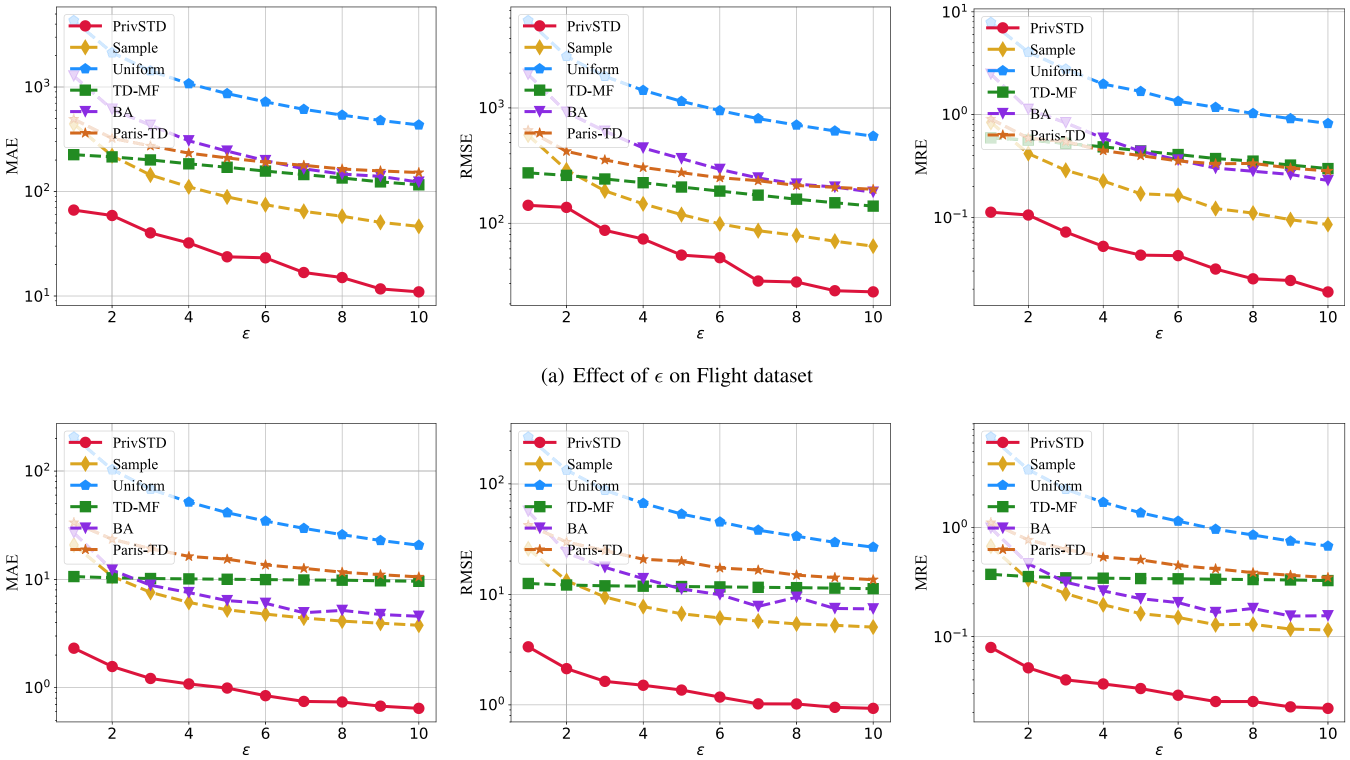
Effect of Privacy Budget. We evaluate the trade-off between privacy and utility on synthetic dataset as shown in Fig. 7, where $\delta = 0.01$, $\omega = 1$, $w = 5$, the sampling rate of tasks during local truth estimation $\alpha = 0.3$, and privacy budget ϵ varies from 1 to 10. From this figure, we can observe that (1) a larger privacy budget incurs lower MAE, RMSE and MRE, which correspond to higher utility but lower privacy protection level. This observation is in agreement with the pattern of the utility-privacy trade-off.



(a) Effects of w on Flight dataset

(b) Effects of w on Weather dataset

Fig. 8. Effect of sliding window size w on real-world datasets.



(a) Effect of ϵ on Flight dataset

(b) Effect of ϵ on Weather dataset

Fig. 9. Effect of privacy budget ϵ on real-world datasets.

(2) TD-MF is robust to large noise, as it adds perturbation into the loss function of matrix factorization instead of the original answers. (3) PrivSTD always has the best performance compare with other mechanisms.

5.3 Experiments on Real-World Datasets

In the subsection, we evaluate the performances of PrivSTD, two baseline methods and two state-of-the-art approaches over two real-world datasets under different privacy budgets and sliding window sizes.

Effect of Sliding Window Size. We vary the sliding window size and measure the MAE, RMSE and MRE of truth under the conditions that $\epsilon = 5$, $\delta = 0.02$, the sampling rate of tasks during local truth estimation $\alpha = 0.3$, and the number of edge servers is 3. The results are shown in Fig. 8. From the results of the two datasets, we have the following observations. First, although the two datasets are different, the changing trends of the errors of all methods on these datasets are almost the same and such trend is also similar to that on the synthetic dataset. Second, *Uniform* still has the largest MAE, since it ignores both the different reliability of individuals and the correlations between truths over time, thus it adds the maximum noise to each worker at each timestamp. Third, *Sampling*, TD-MF, Paris-TD and BA have better performance than *Uniform* because they either sample appropriate perturbed timestamps to reduce overall noise or decrease the effect of noise on answers. Last and most importantly, PrivSTD always performers better than other methods in terms of the accuracy of truth discovery.

Effect of Privacy Budget. Fig. 9 shows the utility-privacy trade-off when privacy budget ϵ varies from 1 to 10 under the conditions that sliding window size $w = 10$, $\delta = 0.02$, the sampling rate of tasks during local truth estimation $\alpha = 0.3$, and the number of edge servers is 3. This figure shows the same pattern as Fig. 7, which demonstrates that PrivSTD is effective in protecting the privacy of streaming truth discovery.

6 Conclusion

In this paper, we have proposed an edge computing based privacy-preserving streaming data truth discovery mechanism, called PrivSTD, which considers both the evolving pattern of truths and the different reliabilities of workers, to achieve accurate truth discovery for streaming data while protecting the privacy of workers in crowdsourcing system. We introduced multiple edge servers to help workers securely estimate local truths and workers' reliabilities, based on which the budget recycle mechanism and reliability-based perturbation mechanism are proposed. The budget recycle mechanism enables PrivSTD to reduce the probability of adding perturbation into streaming answers on the basis of truth evolution, and the reliability-based perturbation mechanism further reduces the perturbation magnitude according to workers' reliabilities. Theoretical analysis and extensive experimental results on both synthetic and real-world datasets demonstrated that PrivSTD achieves better performance than the state-of-the-art approaches while still satisfies rigorous w-event (ϵ, δ)-DP.

Limitations and Future Works. PrivSTD is quite useful in preserving the privacy of streaming truth discovery, however, it still has some limitations.

First, we follow the prior works to assume that each source has the same degree of reliability on his/her different observations. Actually, workers may have different expertise on different domains. We plan to extend our work to integrate with the domain-ware truth discovery [38], [39]. In this case, we need to consider finer-grained privacy protection and perturbation.

Second, in mobile crowdsourcing, when the amount of data is very large, edge computing (EC) is an effective technology to aggregate and process partial data before they reach the cloud server. Besides, artificial intelligence (AI) techniques with edge computing have been considered as powerful tools for processing big data. We are considering adopting AI techniques for EC-based crowdsourcing system to provide additional functions, such as extracting the behaviors of physical/networking resources and users in different times and scenarios, dynamically monitoring and adjusting the configuration of network resources [40], [41].

Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant 2019YFA0706403, National Natural Science Foundation of China under Grants 62072472, 61702562, U19A2067, U20A20182 and 61872274, Natural Science Foundation of Hunan Province, China under Grant 2020JJ2050, 111 Project under Grant No. B18059, the Young Elite Scientists Sponsorship Program by CAST under Grant 2018QNRC001 and the Young Talents Plan of Hunan Province of China under Grant 2019RS2001.

References

[1] R. Buettner, “A systematic literature review of crowdsourcing research from a human resource management perspective,” in *Proc. 48th Hawaii Int. Conf. Syst. Sci.*, 2015, pp. 4609–4618.

[2] J. Prpić, A. Taeihagh, and J. Melton, “The fundamentals of policy crowdsourcing,” *Policy Internet*, vol. 7, no. 3, pp. 340–361, 2015.

[3] C. Miao *et al.*, “Privacy-preserving truth discovery in crowd sensing systems,” *ACM Trans. Sensor Netw.*, vol. 15, no. 1, 2019, Art. no. 9.

[4] Y. Li *et al.*, “On the discovery of evolving truth,” in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2015, pp. 675–684.

[5] Y. Li *et al.*, “A survey on truth discovery,” *ACM SIGKDD Explorations Newslett.*, vol. 17, no. 2, pp. 1–16, 2016.

[6] R. W. Ouyang, L. M. Kaplan, A. Toniolo, M. Srivastava, and T. J. Norman, “Parallel and streaming truth discovery in large-scale quantitative crowdsourcing,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 10, pp. 2984–2997, Oct. 2016.

[7] B. Zhao and J. Han, “A probabilistic model for estimating real-valued truth from conflicting sources,” in *Proc. 10th Int. Workshop Quality Databases*, 2012, pp. 1–7.

[8] B. Zhao, B. I. Rubinstein, J. Gemmell, and J. Han, “A Bayesian approach to discovering truth from conflicting sources for data integration,” *Proc. VLDB Endowment*, vol. 5, no. 6, pp. 550–561, 2012.

[9] J. Pasternack and D. Roth, “Knowing what to believe (when you already know something),” in *Proc. 23rd Int. Conf. Comput. Linguistics*, 2010, pp. 877–885.

[10] Z. Wang, X. Pang, J. Hu, W. Liu, Q. Wang, Y. Li, and H. Chen, “When mobile crowdsensing meets privacy,” *IEEE Commun. Mag.*, vol. 57, no. 9, pp. 72–78, Sep. 2019.

[11] Z. Wang *et al.*, “Personalized privacy-preserving task allocation for mobile crowdsensing,” *IEEE Trans. Mobile Comput.*, vol. 18, no. 6, pp. 1330–1341, Jun. 2019.

[12] H. Wu, L. Wang, and G. Xue, “Privacy-aware task allocation and data aggregation in fog-assisted spatial crowdsourcing,” *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 1, pp. 589–602, First Quarter 2020.

[13] H. Wu, L. Wang, G. Xue, J. Tang, and D. Yang, “Enabling data trustworthiness and user privacy in mobile crowdsensing,” *ACM Trans. Netw.*, vol. 27, no. 6, pp. 2294–2307, 2019.

[14] C. Miao *et al.*, “Cloud-enabled privacy-preserving truth discovery in crowd sensing systems,” in *Proc. 13th ACM Conf. Embedded Netw. Sensor Syst.*, 2015, pp. 183–196.

[15] Y. Zheng, H. Duan, X. Yuan, and C. Wang, “Privacy-aware and efficient mobile crowdsensing with truth discovery,” *IEEE Trans. Dependable Secure Comput.*, vol. 17. no. 1, pp. 121–133, Jan./Feb. 2020.

[16] C. Zhang, L. Zhu, C. Xu, K. Sharif, and X. Liu, “PPTDS: A privacy-preserving truth discovery scheme in crowd sensing systems,” *Inf. Sci.*, vol. 484, pp. 183–196, 2019.

[17] C. Zhang, L. Zhu, C. Xu, X. Liu, and K. Sharif, “Reliable and privacy-preserving truth discovery for mobile crowdsensing systems,” *IEEE Trans. Dependable Secure Comput.*, early access, May 28, 2019, doi: 10.1109/TDSC.2019.2919517.

[18] C. Zhang, L. Zhu, C. Xu, K. Sharif, X. Du, and M. Guizani, “LPTD: Achieving lightweight and privacy-preserving truth discovery in CIoT,” *Future Gener. Comput. Syst.*, vol. 90, pp. 175–184, 2019.

[19] X. Tang, C. Wang, X. Yuan, and Q. Wang, “Non-interactive privacy-preserving truth discovery in crowd sensing applications,” in *Proc. IEEE Conf. Comput. Commun.*, 2018, pp. 1988–1996.

[20] Y. Zheng, H. Duan, and C. Wang, “Learning the truth privately and confidently: Encrypted confidence-aware truth discovery in mobile crowdsensing,” *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 10, pp. 2475–2489, Oct. 2018.

[21] T. Wang, N. Li, and S. Jha, “Locally differentially private heavy hitter identification,” *IEEE Trans. Dependable Secure Comput.*, early access, Jul. 9, 2019, doi: 10.1109/TDSC.2019.2927695.

[22] M. U. Hassan, M. H. Rehmani, and J. Chen, “Differential privacy techniques for cyber physical systems: A survey,” *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 746–789, First Quarter 2020.

[23] H. Jiang, J. Pei, D. Yu, J. Yu, B. Gong, and X. Cheng, “Differential privacy and its applications in social network analysis: A survey,” 2020, *arXiv: 2010.02973*.

[24] H. Sun, B. Dong, H. W. Wang, T. Yu, and Z. Qin, “Truth inference on sparse crowdsourcing data with local differential privacy,” in *Proc. IEEE Int. Conf. Big Data*, 2018, pp. 488–497.

[25] Y. Li *et al.*, “An efficient two-layer mechanism for privacy-preserving truth discovery,” in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 1705–1714.

[26] Y. Li *et al.*, “Towards differentially private truth discovery for crowd sensing systems,” 2018, *arXiv: 1810.04760*.

[27] P. Sun, Z. Wang, L. Wu, Y. Feng, and Z. Wang, “Towards personalized privacy-preserving incentive for truth discovery in mobile crowdsensing systems,” *IEEE Trans. Mobile Comput.*, early access, Jun. 19, 2020, doi: 10.1109/TMC.2020.3003673.

[28] G. Xu *et al.*, “Catch you if you deceive me: Verifiable and privacy-aware truth discovery in crowdsensing systems,” in *Proc. 15th ACM Asia Conf. Comput. Commun. Secur.*, 2020, pp. 178–192.

[29] A. Vadavalli and R. Subhashini, “An improved differential privacy-preserving truth discovery approach in healthcare,” in *Proc. IEEE 10th Annu. Inf. Technol. Electron. Mobile Commun. Conf.*, 2019, pp. 1031–1037.

[30] Q. Li, Y. Li, J. Gao, B. Zhao, W. Fan, and J. Han, “Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation,” in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2014, pp. 1187–1198.

[31] Q. Li *et al.*, “A confidence-aware approach for truth discovery on long-tail data,” *Proc. VLDB Endowment*, vol. 8, no. 4, pp. 425–436, 2014.

[32] C. Dwork *et al.*, “The algorithmic foundations of differential privacy,” *Found. Trends Theor. Comput. Sci.*, vol. 9, no. 3/4, pp. 211–407, 2014.

[33] Q. Wang, Y. Zhang, X. Lu, Z. Wang, Z. Qin, and K. Ren, “Real-time and spatio-temporal crowd-sourced social network data publishing with differential privacy,” *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 4, pp. 591–606, Jul./Aug. 2018.

[34] G. Kellaris, S. Papadopoulos, X. Xiao, and D. Papadias, “Differentially private event sequences over infinite streams,” *Proc. VLDB Endowment*, vol. 7, no. 12, pp. 1155–1166, 2014.

[35] Z. Wang *et al.*, “Privacy-preserving crowd-sourced statistical data publishing with an untrusted server,” *IEEE Trans. Mobile Comput.*, vol. 18, no. 6, pp. 1356–1367, Jun. 2019.

[36] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in *Proc. Int. Conf. Theory Appl. Cryptogr. Techn.*, 1999, pp. 223–238.

[37] Y. Zheng, G. Li, Y. Li, C. Shan, and R. Cheng, “Truth inference in crowdsourcing: Is the problem solved?” *Proc. VLDB Endowment*, vol. 10, no. 5, pp. 541–552, 2017.

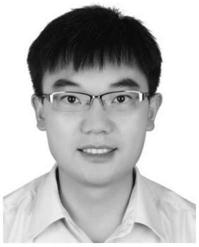
[38] F. Ma *et al.*, "FaitCrowd: Fine grained truth discovery for crowdsourced data aggregation," in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2015, pp. 745–754.

[39] X. Lin and L. Chen, "Domain-aware multi-truth discovery from conflicting sources," *Proc. VLDB Endowment*, vol. 11, no. 5, pp. 635–647, 2018.

[40] Y. Zhao *et al.*, "Privacy-preserving blockchain-based federated learning for IoT devices," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1817–1829, Aug. 2020.

[41] Q.-V. Pham *et al.*, "A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and state-of-the-art," *IEEE Access*, vol. 8, pp. 116974–117017, Jun. 2020.

Dan Wang (Student Member, IEEE) received the BE degree in electronic engineering and the MSc degree in control science and engineering from China Agricultural University, China, in 2015 and 2017, respectively. She is currently working toward the PhD degree at the School of Computer Science and Technology, Central South University, China. Her research interest focuses on privacy protection for IoT data.



Ju Ren (Member, IEEE) received the BSc, MSc, and PhD degrees all in computer science, from Central South University, China, in 2009, 2012, and 2016, respectively. During 2013-2015, he was a visiting PhD student at the Department of Electrical and Computer Engineering, University of Waterloo, Canada. Currently, he is a professor with the School of Computer Science and Engineering, Central South University, China. His research interests include Internet-of-Things, network computing and edge computing. He currently serves/has served as an associate editor for the *IEEE Transactions on Vehicular Technology* and the *Peer-to-Peer Networking and Applications*, a guest editor for the *IEEE Wireless Communications*, *IEEE Transactions on Industrial Informatics* and *IEEE Network*, and a TPC member of many international conferences including IEEE INFOCOM'21/20/19/18, etc. He also served as the general co-chair for IEEE BigDataSE'20, the TPC co-chair for IEEE BigDataSE'19, a poster co-chair for IEEE MASS'18, a track co-chair for IEEE/CIC ICCC'19, I-SPAN'18 and IEEE VTC'17 Fall, and an active reviewer for more than 20 international journals. He received many best paper awards from IEEE flagship conferences, including IEEE ICC'19 and IEEE HPCC'19, etc., and the IEEE TCSC Early Career Researcher Award (2019).

Zhibo Wang (Senior Member, IEEE) received the BE degree in automation from Zhejiang University, China, in 2007, and the PhD degree in electrical engineering and computer science from the University of Tennessee, Knoxville, Tennessee, in 2014. He is currently a professor with the Institute of Cyber Science and Technology, College of Computer Science and Technology, Zhejiang University, China. His currently research interests include AI security, Internet of Things, network security and privacy protection. He is a member of ACM.

Xiaoyi Pang received the BE degree in information security from Wuhan University, China, in 2018. She is currently working toward the master's degree at the School of Cyber Science and Engineering, Wuhan University, China. Her research interest focuses on privacy protection in mobile crowdsensing system.

Yaoxue Zhang (Senior Member, IEEE) received the BSc degree from the Northwest Institute of Telecommunication Engineering, China, in 1982, and the PhD degree in computer networking from Tohoku University, Japan, in 1989. Currently, he is a professor with the Department of Computer Science and Technology, Tsinghua University, China and also a professor with the School of Computer Science and Engineering, Central South University, China. His research interests include computer networking, operating systems, ubiquitous/pervasive computing, transparent computing, and big data. He has published more than 200 technical papers in international journals and conferences, as well as nine monographs and text-books. Currently, he is serving as the editor-in-chief of the *Chinese Journal of Electronics*. He is a fellow of the Chinese Academy of Engineering.

Xuemin (Sherman) Shen (Fellow, IEEE) received the BSc degree from Dalian Maritime University, China, in 1982, and the MSc and PhD degrees from Rutgers University, Newark, New Jersey, in 1987 and 1990, respectively, all in electrical engineering. He is currently a University professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. He was the associate chair for graduate studies from 2004 to 2008. His research focuses on resource management in interconnected wireless/wired networks, wireless network security, social networks, smart grid, and vehicular ad hoc and sensor networks. He was a recipient of the Distinguished Performance Award from the Faculty of Engineering, University of Waterloo, Canada, in 2002 and 2007; the Premiers Research Excellence Award from the Province of Ontario, in 2003; the Outstanding Performance Award from the University of Waterloo, Canada, in 2004, 2007, 2010, and 2014; and the Excellent Graduate Supervision Award, in 2006. He served as the Technical Program Committee chair/co-chair for IEEE Globecom'16, ACM MobiHoc'15, IEEE INFOCOM'14, IEEE VTC-Fall'10, the Symposia chair for IEEE ICC'10, the tutorial chair for IEEE VTC-Spring'11 and IEEE ICC'08, and the Technical Program Committee chair for IEEE GLOBECOM'07. He also serves/has served as the editor-in-chief for the *Peer-to-Peer Networking and Application*, the *IEEE Internet of Things Journal*, *IET Communications*, and *IEEE Network*. He is a registered professional engineer of Ontario, Canada, an Engineering Institute of Canada fellow, a Canadian Academy of Engineering fellow, a Royal Society of Canada fellow, a Chinese Academy of Engineering Foreign fellow and a distinguished lecturer of the IEEE Vehicular Technology and Communications Societies.