Robust Truth Discovery Against Data Poisoning in Mobile Crowdsensing

Zonghao Huang
Oklahoma State University
Stillwater, Oklahoma 74078
Email: zonghao.huang@okstate.edu

Miao Pan
University of Houston
Houston, Texas 77204
Email: mpan2@uh.edu

Yanmin Gong
Oklahoma State University
Stillwater, Oklahoma 74078
Email: yanmin.gong@okstate.edu

***Abstract*—Nowadays most mobile devices are equipped with advanced sensors, enabling the measurement of information about surrounding environment or social settings. The ubiquity of mobile devices makes them the perfect platform for massive data collection, which motivates the emergence of mobile crowdsensing paradigm. However, due to the inherent noisy nature of the sensing process and the limited capability of low-cost commodity sensors, crowdsensed information tends to be less reliable compared with sensing results through dedicated sensing hardware, and multiple crowdsensing sources may conflict with each other. Thus, it is important to resolve conflicts in the collected data and discover the underlying truth. Traditional truth discovery approaches usually estimate the reliability of data sources and predict the truth value based on source reliability. However, recent data poisoning attacks greatly degrade the performance of existing truth discovery algorithms, where attackers aim to maximize the utility loss. In this paper, we investigate the data poisoning attacks on truth discovery and propose a robust approach against such attacks through additional source estimation and source filtering before data aggregation. Based on real-world data, we simulate our approach and evaluate its performance under data poisoning attacks, demonstrating the robustness of our approach.**

I. INTRODUCTION

Nowadays, mobile devices have become an indispensable part of our daily life. With their advanced sensing capabilities and ubiquitous presence, they could collect massive information about surrounding environments and social settings [1], [2]. This motivates the emergence of mobile crowdsensing, which employs a large group of mobile users to perform sensing tasks at low cost and extracts the collected information to measure, map, analyze, estimate or infer any processes of common interest [3]. For example, in geo-tagging campaigns, people report locations of conditions they visit which require public attention by their mobile phones with Global Position System sensors. With the provided information, the government could refine the reported geo-tags.

However, multiple data sources usually provide conflicting information about the sensing object due to the inherent noisy nature of the sensing process. The introduced noises and errors could be caused by low calibration of sensors, sensor quality, lack of human attention, and even intended deception. Thus, in order to fully utilize the information in mobile crowdsensing, it is important to recover the truth from the noisy sensory data. A naive way to recover the truth is taking the average of the sensory data or through majority voting. However, in such methods, the credibility of the source is not differentiated and each source contributes equally to the final result. Considering varying credibility across crowd sensors, it is desirable to estimate source reliability and use it as the weight to calculate a weighted sum of sensing results. However, the reliability of the source in mobile crowdsensing is usually unknown a prior. This makes reliability estimation a critical and challenging part for truth discovery. Based on the intuition that the reliability of the source is closely relevant with the accuracy of its sensing results, most truth discovery protocols update the reliability and the truth through an iterative process. There exists tremendous work regarding truth discovery [4]–[12] or reliability-based data aggregation [13]–[16] based on the iterative method.

The openness of the crowdsensing platform makes it an easy target for attackers. An attacker can easily manipulate the sensing results by hiring some malicious workers to submit poisoned data at a low cost, widely known as data poisoning attack. Such data poisoning attack is usually formulated as a bi-level optimization problem, whose objective is to maximize the utility loss [17]–[19]. Although the truth discovery process could provide some level of defense through assigning the malicious workers with low weights to reduce their impacts on the final estimated truth, the attacks could still distort the final result. Thus, in order to guarantee the advantage of crowdsensing system, it is necessary to design additional methods to defend against such data poisoning attacks.

In our paper, we focus on data poisoning attacks on truth discovery and propose a robust approach against such attacks. We consider Conflict Resolution on Heterogeneous data truth discovery algorithm (CRH) [12], and formulate an a bi-level optimization problem for data poisoning attacks on CRH, where malicious attackers collude to maximize the utility loss of the truth discovery. Then we propose our approach to defend against such attacks on truth discovery by designing additional source evaluation and source filtering method. The first step of our approach is to estimate the error bias and variance of each source, which indicate the error level of the workers. Then we remove those workers whose error level is higher than a pre-defined threshold value, and use the remaining data for the truth discovery process. We simulate our approach on real-world data and the simulation result demonstrates the robustness of the proposed approach.

In summary, our contributions in this paper are:

1) We consider an optimal data poisoning attack strategy in truth discovery system, where malicious workers aim to maximize the utility loss of truth discovery to render the estimated truth useless. We formulate such data poisoning attack strategy as a bi-level optimization problem.
2) We propose a robust truth discovery algorithm, which integrates source evaluation and source filtering process into the CRH method. The source evaluation estimates the error bias and variance of the sources, and the source filtering process uses the estimated bias and variance as the criteria to remove unreliable sources.
3) We conduct extensive experiments on real-world data. Our simulation results show that our approach could provide accurate and reliable results in the presence of data poisoning attacks.

In the remaining parts of our paper, we firstly describe the problem setting and formulate the truth discovery problem as an optimization problem in Section II . In Section III, we consider an optimal data poisoning attacks model. Then we propose our robust approach against such attacks in Section IV. In Section V, we conduct an experiment based on real-world data, and evaluate our simulation. In Section VI and Section VII, we describe some related work and conclude our paper.

II. Problem Statement

A. Problem Setting

In this paper, we consider a general crowdsensing framework with two parties: mobile users and the server. Mobile users work as workers to provide sensory data on the objects within the object set $\mathcal{N}$. We use v_i^k to denote the sensory data on the i^{th} object from the k^{th} worker. Among the mobile users, we assume that there are normal workers and malicious workers. The malicious workers could manipulate their sensory data in order to achieve their attack goal, which is usually known as data poisoning attack. In Section III, we will describe such attack with respect to the attack goal, attackers' knowledge, the attackers' capability and their attack strategy. Here we use $\mathcal{K}$ and $\mathcal{M}$ to denote the normal worker set and the malicious worker set respectively. The server collects all the data $\{v_i^k\}_{k\in\mathcal{K}\cup\mathcal{M},i\in\mathcal{N}}$ from workers and aims to estimate the ground truth of the objects and the qualities of the workers. We use v_i^*, $\hat{v}_i^*$, and w_k to denote the ground truth of the i^{th} object, the estimated truth of the i^{th} object, and the assigned weight to the k^{th} worker respectively.

Throughout this paper, we assume that the normal workers sense data independently and we only focus on the setting where the sensory data is continuous.

B. Truth Discovery

After receiving all the sensory data $\{v_i^k\}_{k\in\mathcal{K}\cup\mathcal{M},i\in\mathcal{N}}$, the server estimate the truth $\{\hat{v}_i^*\}_{i\in\mathcal{N}}$ among the conflicting information. Following [12], [20]–[22], here we model such truth discovery problem as an optimization problem described as:

$$\underset{\{w_k\},\{\hat{v}_i^*\}}{\operatorname{argmin}} \quad \sum_{k\in\mathcal{K}\cup\mathcal{M}} w_k \sum_{i\in\mathcal{N}} d(v_i^k,\hat{v}_i^*), \tag{1a}$$

$$\text{s.t.} \quad \delta(\{w_k\}_{k\in\mathcal{K}\cup\mathcal{M}}) = 1, \tag{1b}$$

where $d(v_i^k,\hat{v}_i^*)$ refers to the Euclidean distance between the estimated truth $\hat{v}_i^*$ and the observation v_i^k: $d(v_i^k,\hat{v}_i^*) = (v_i^k - \hat{v}_i^*)^2$, and $\delta(\{w_k\}_{k\in\mathcal{K}\cup\mathcal{M}})$ is the regularization function reflecting the distribution of $\{w_k\}_{k\in\mathcal{K}\cup\mathcal{M}}$. Here, we follow the widely adopted truth discovery method CRH proposed in [12], where the regularization function is defined by:

$$\delta(\{w_k\}_{k\in\mathcal{K}\cup\mathcal{M}}) = \sum_{k\in\mathcal{K}\cup\mathcal{M}} \exp(-w_k). \tag{2}$$

In order to solve the optimization problem, the iterative method is commonly used, where the estimated truth and assigned weights are updated alternatively. More details on the CRH truth discovery algorithm are shown in Algorithm 1.

Algorithm 1 CRH Truth Discovery Algorithm

Input: $\{v_i^k\}_{k\in\mathcal{K}\cup\mathcal{M},i\in\mathcal{N}}$
  $w_k \leftarrow 1$;
  repeat
    for $i \in \mathcal{N}$ **do**
      Compute $\hat{v}_i^* = \frac{\sum_{k\in\mathcal{K}\cup\mathcal{M}} v_i^k w_k}{\sum_{k\in\mathcal{K}\cup\mathcal{M}} w_k}$;
    end for
    for $k \in \mathcal{K}\cup\mathcal{M}$ **do**
      Compute $w_k = -\log \frac{\sum_{i\in\mathcal{N}} (v_i^k - \hat{v}_i^*)^2}{\sum_{k\in\mathcal{K}\cup\mathcal{M}}\sum_{i\in\mathcal{N}} (v_i^k - \hat{v}_i^*)^2}$;
    end for
  until results converge
Output: $\{\hat{v}_i^*\}_{i\in\mathcal{N}}$ and $\{w_k\}_{k\in\mathcal{K}\cup\mathcal{M}}$

III. Data Poisoning Attacks

In this section, we introduce the attack model including the attacker's goal, adversarial knowledge, adversarial capability, and the data poisoning attack strategy.

Attackers' goal. The attacker aims to maximize the error of the truth discovery result in order to render the estimated truth useless, which is usually called *availability attack*. Specifically, the attackers' goal is to maximize the distance between the estimated truth from truth discovery algorithm before and after the attacks. We use $\{\tilde{v}_i^*\}$ to denote the estimated truth without malicious workers involved. We could formulate the attackers' goal as a maximization problem:

$$\max_{\{v_i^j\}_{j\in\mathcal{M}}} \quad \sum_{i\in\mathcal{N}} \left(\hat{v}_i^* - \tilde{v}_i^*\right)^2.$$

Adversarial knowledge. We assume that the attackers could have access to all the observations $\{v_i^k\}_{i\in\mathcal{N},k\in\mathcal{K}\cup\mathcal{M}}$ from the normal workers. The attackers could obtain this information by eavesdropping the communication between the normal workers and the server. Besides, the attackers have complete knowledge about the truth discovery algorithm including how the server updates the weight and estimates the truth values.

Adversarial capability. The capability of the attackers is limited by the malicious worker ratio, which could be defined by $\rho = |\mathcal{M}|/|\mathcal{M} \cup \mathcal{K}|$. Usually, the malicious worker ratio is small because in reality the attackers could only hire a small fraction of malicious workers. According to the previous works [17], the ratio is no higher than 20%.

Data poisoning attack strategy. According to the attackers' goal and their knowledge, we could formulate the data poisoning attacks as a bi-level optimization problem [23]:

$$\max_{\{v_i^j\}_{j\in\mathcal{M}}} \quad \sum_{i\in\mathcal{N}} \left(\hat{v}_i^* - \tilde{v}_i^*\right)^2, \tag{3a}$$

$$\text{s.t.} \quad v_i^j \in [\min_k\{v_i^k\}, \max_k\{v_i^k\}], \quad j \in \mathcal{M}, \tag{3b}$$

$$\{\hat{v}_i^*\} = \operatorname{argmin} \sum_{k\in\mathcal{K}\cup\mathcal{M}} w_k \sum_{i\in\mathcal{N}} d(v_i^k, \hat{v}_i^*),$$

$$\text{s.t.} \quad \sum_{k\in\mathcal{K}\cup\mathcal{M}} \exp(-w_k) = 1, \tag{3c}$$

$$\{\tilde{v}_i^*\} = \operatorname{argmin} \sum_{k\in\mathcal{K}} w_k \sum_{i\in\mathcal{N}} d(v_i^k, \tilde{v}_i^*),$$

$$\text{s.t.} \quad \sum_{k\in\mathcal{K}} \exp(-w_k) = 1. \tag{3d}$$

Constraints (3b) ensure that the poisoned data is located within the normal range to avoid detection. Note that in [18], [19], the authors use a similar bi-level optimization problem to formulate data poisoning attack strategy on truth discovery but they focus on categorical data, which is different from ours. To solve the bi-level optimization problem, we use the gradient-based method to search for the optimal solution, where the gradient of the objective function with respect to v_i^j is defined by:

$$\nabla_{v_i^j} f = 2 \cdot (\hat{v}_i^* - \tilde{v}_i^*) \cdot \frac{w_j}{\sum_{k\in\mathcal{K}\cup\mathcal{M}} w_k}. \tag{4}$$

The details of the data poisoning attack algorithm are given in Algorithm 2.

IV. Robust Truth Discovery against Data Poisoning Attacks

In Algorithm 2, there are two steps in each iteration: truth discovery to update the estimated truth and the assigned weight, and the poisoned data update. In the poisoned data update, the sign of the gradient is determined by $\hat{v}_i^* - \tilde{v}_i^*$. If the $\hat{v}_i^*$ is larger than $\tilde{v}_i^*$, the update will increase the value of poisoned data, and such value increase will have a positive feedback on the truth discovery process and make the estimated truth $\hat{v}_i^*$ larger. Therefore, as the result of the data poisoning attacks, the poisoned data $\{v_i^j\}_{i\in\mathcal{N},j\in\mathcal{M}}$ from malicious workers deviate from the truth and reach the boundary ($\min_k\{v_i^k\}$ or $\max_k\{v_i^k\}$) in order to maximize the utility loss. For each observation v_i^k, we could divide it into two components including ground truth and error:

$$v_i^k = v_i^* + e_i^k. \tag{5}$$

It means that as the result of data poisoning attacks, the poisoned data would have large error part e_i^k.

The general idea of our approach is to remove the data from those workers with large error before the data aggregation. In our approach, the server firstly estimates the bias b_k and the variance σ_k^2 of data error from each worker, where

$$b_k = \frac{1}{|\mathcal{N}|}\sum_{i\in\mathcal{N}} e_i^k \quad \text{and} \quad \sigma_k^2 = \frac{1}{|\mathcal{N}|}\sum_{i\in\mathcal{N}} \left(e_i^k - b_k\right)^2. \tag{6}$$

Then the server removes those workers with large error level defined by $b_k^2 + \sigma_k^2$, which measures the expected euclidean distance between the sensory data and the ground truth, and finally uses the remaining data for truth discovery.

Algorithm 2 Data Poisoning Attack Algorithm

Input: $\{v_i^k\}_{i\in\mathcal{N},k\in\mathcal{K}}$
  Initialize $\{v_i^j\}_{i\in\mathcal{N},j\in\mathcal{M}}$;
  $w_k \leftarrow 1$;
  Compute $\{\tilde{v}_i^*\}$ by Algorithm 1 with input $\{v_i^k\}_{i\in\mathcal{N},k\in\mathcal{K}}$;
  repeat
    repeat
      for $i \in \mathcal{N}$ **do**
        Compute $\hat{v}_i^* = \frac{\sum_{k\in\mathcal{K}\cup\mathcal{M}} v_i^k w_k}{\sum_{k\in\mathcal{K}\cup\mathcal{M}} w_k}$;
      end for
      for $k \in \mathcal{K}\cup\mathcal{M}$ **do**
        Compute $w_k = -\log \frac{\sum_{i\in\mathcal{N}} (v_i^k - \hat{v}_i^*)^2}{\sum_{k\in\mathcal{K}\cup\mathcal{M}} \sum_{i\in\mathcal{N}} (v_i^k - \hat{v}_i^*)^2}$;
      end for
    until results converge
    for $j \in \mathcal{M}$ **do**
      for $i \in \mathcal{N}$ **do**
        Compute $v_i^j \leftarrow v_i^j + 2 \cdot (\hat{v}_i^* - \tilde{v}_i^*) \cdot \frac{w_j}{\sum_{k\in\mathcal{K}\cup\mathcal{M}} w_k}$;
      end for
    end for
  until results converge
Output: $\{v_i^j\}_{i\in\mathcal{N},j\in\mathcal{M}}$

A. Source Evaluation

The server firstly estimates the bias $\{b_k\}_{k\in\mathcal{K}\cup\mathcal{M}}$ based on the collected data $\{v_i^k\}_{i\in\mathcal{N},k\in\mathcal{K}\cup\mathcal{M}}$. Although we could not obtain the error e_i^k due to the unknown ground truth, we could obtain the difference $\gamma(k,j)$ between any two biases by:

$$\gamma(k,j) = b_k - b_j = \frac{1}{|\mathcal{N}|}\sum_{i\in\mathcal{N}} (v_i^k - v_i^j). \tag{7}$$

In order to estimate the bias $\{b_k\}_{k\in\mathcal{K}\cup\mathcal{M}}$, we formulate a loss minimization problem:

$$\min_{\{b_k\}_{k\in\mathcal{K}\cup\mathcal{M}}} \quad \sum_{k\in\mathcal{K}\cup\mathcal{M}} \sum_{j\in\mathcal{K}\cup\mathcal{M}} \Big(b_k - b_j - \gamma(k,j)\Big)^2, \tag{8a}$$

$$\text{s.t.} \quad \sum_{k\in\mathcal{K}\cup\mathcal{M}} b_k = 0, \tag{8b}$$

where the objective refers to the loss measuring the distance between the estimated parameters $\{b_k\}_{k\in\mathcal{K}\cup\mathcal{M}}$ and the observation $\{\gamma(k,j)\}_{k\in\mathcal{K}\cup\mathcal{M},j\in\mathcal{K}\cup\mathcal{M}}$.

Next, we estimate the variance $\{\sigma_k^2\}_{k\in\mathcal{K}\cup\mathcal{M}}$ based on the estimated bias. Similarly, since we do not know the ground truth, we could not obtain the variance of error directly, but we could obtain the sum of any two variances as follows:

$$\begin{aligned}\beta(k,j) &= \sigma_k^2+\sigma_j^2\\ &= \frac{1}{|\mathcal{N}|}\sum_{i\in\mathcal{N}}\left(e_i^k-b_k\right)^2+\frac{1}{|\mathcal{N}|}\sum_{i\in\mathcal{N}}\left(e_i^j-b_j\right)^2\\ &= \frac{1}{|\mathcal{N}|}\sum_{i\in\mathcal{N}}\left(e_i^k-b_k-e_i^j+b_j\right)^2-2\cdot cov(e_i^k,e_i^j)\\ &= \frac{1}{|\mathcal{N}|}\sum_{i\in\mathcal{N}}\left(v_i^k-v_i^j\right)^2-2\cdot cov(e_i^k,e_i^j).\end{aligned} \tag{9}$$

Assume that normal workers sense data independently, we have $cov(e_i^k,e_i^j)\approx 0$. Thus, we have:

$$\beta(k,j)=\sigma_k^2+\sigma_j^2=\frac{1}{|\mathcal{N}|}\sum_{i\in\mathcal{N}}\left(v_i^k-v_i^j\right)^2. \tag{10}$$

In order to estimate the variance of error, we also formulate a loss minimization problem:

$$\min_{\{\sigma_k^2\}_{k\in\mathcal{K}\cup\mathcal{M}}} \sum_{k\in\mathcal{K}\cup\mathcal{M}}\sum_{j\in\mathcal{K}\cup\mathcal{M}}\left(\sigma_k^2+\sigma_j^2-\beta(k,j)\right)^2, \tag{11a}$$

$$\text{s.t.}\quad |\mathcal{K}\cup\mathcal{M}|\sum_{k\in\mathcal{K}\cup\mathcal{M}}\sigma_k^2=\sum_{k\in\mathcal{K}\cup\mathcal{M}}\sum_{j\in\mathcal{K}\cup\mathcal{M}}\beta(k,j). \tag{11b}$$

We could solve the optimization problems (8) and (11) by using Lagrange multiplier method, and then get the estimated error bias $\{b_k\}_{k\in\mathcal{K}\cup\mathcal{M}}$ and variance $\{\sigma_k^2\}_{k\in\mathcal{K}\cup\mathcal{M}}$.

B. Threshold-Based Source Filtering

After the source evaluation, the server could estimate the error bias and variance of each worker. Intuitively, we remove those data sources with high error level in order to avoid the impacts of the poisoned data. We could set up a threshold T, and remove those workers whose error level $b_k^2+\sigma_k^2$ is higher than the threshold T. We will discuss the choice of threshold value in our simulation. More details on the threshold-based source filtering are given in Algorithm 3.

Algorithm 3 Source Filtering Algorithm

Input: $\{b_k\}_{k\in\mathcal{K}\cup\mathcal{M}}$, $\{\sigma_k^2\}_{k\in\mathcal{K}\cup\mathcal{M}}$, T, and $\mathcal{K}\cup\mathcal{M}$
  Initialize $\mathcal{P}=\emptyset$;
  for $k\in\mathcal{K}\cup\mathcal{M}$ **do**
    if $b_k^2+\sigma_k^2<T$ **then**
      $\mathcal{P}\leftarrow\mathcal{P}\cup\{k\}$;
    end if
  end for
Output: selected worker set: $\mathcal{P}$

C. The Final Algorithm

In this section, we give an overview of our approach. The inputs of our approach are all sensory data from both normal workers and malicious workers. We firstly evaluate the workers by estimating the error bias and variance (by solving problems (8) and (11)). Then we use the source filtering mechanism (Algorithm 3) to remove the malicious sources and get the selected source set $\mathcal{P}$. Then the truth discovery algorithm works on the selected data to get the estimated truth. More details on our approach is shown in Algorithm 4.

Algorithm 4 Robust Truth Discovery Algorithm

Input: $\{v_i^k\}_{i\in\mathcal{N},k\in\mathcal{K}\cup\mathcal{M}}$
  Compute $\{b_k\}_{k\in\mathcal{K}\cup\mathcal{M}}$ and $\{\sigma_k^2\}_{k\in\mathcal{K}\cup\mathcal{M}}$;
  $\mathcal{P}\leftarrow$ Algorithm 3;
  Initialize w_k;
  repeat
    Compute
    for $i\in\mathcal{N}$ **do**
      Compute $\hat{v}_i^*=\frac{\sum_{k\in\mathcal{P}}v_i^k w_k}{\sum_{k\in\mathcal{P}}w_k}$;
    end for
    for $k\in\mathcal{P}$ **do**
      Compute $w_k=-\log\frac{\sum_{i\in\mathcal{N}}(v_i^k-\hat{v}_i^*)^2}{\sum_{k\in\mathcal{P}}\sum_{i\in\mathcal{N}}(v_i^k-\hat{v}_i^*)^2}$;
    end for
  until results converge
Output: $\{\hat{v}_i^*\}_{i\in\mathcal{N}}$

V. Experimental Evaluation

In this section, we simulate our approach (Algorithm 4) and present the simulation results to show the robustness and the efficiency of our work.

A. Simulation Setup

In our experiments, we use MATLAB to simulate our approach based on real-world data.

1) Dataset: Our simulation is based on real-world data. The real-world data used for our simulation is weather data [24]. The weather dataset describes the weather information in 30 major USA cities, thus we have 30 sensing objects ($|\mathcal{N}|=30$). Each data entry includes temperature, humidity, weather condition. The weather data is collected from 16 weather websites, thus there are 16 data sources. We assume these 16 data sources are normal workers ($|\mathcal{K}|=16$). In our simulation, we only focus on the temperature data.

2) Malicious Worker Simulation: In order to test the robustness and accuracy of our approach in the presence of attackers, we need to simulate some malicious workers in addition to the 16 normal workers. These malicious workers would follow the optimal data poisoning strategy (Algorithm 2) to update their poisoned data. In this simulation, we need to control the malicious worker ratio $\rho=|\mathcal{M}|/|\mathcal{M}\cup\mathcal{K}|$. As we discuss before, the ratio ρ should be less than 20%, thus we set $|\mathcal{M}|=\{0,1,2,3,4\}$ in our simulation.

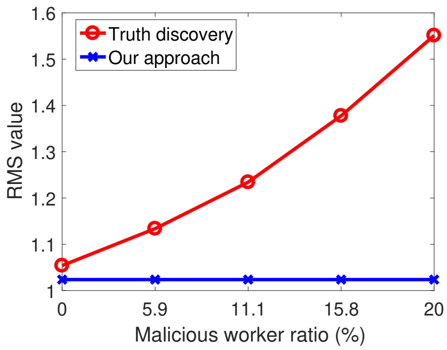

Fig. 1: Comparison between traditional truth discovery and our approach under data poisoning attack.

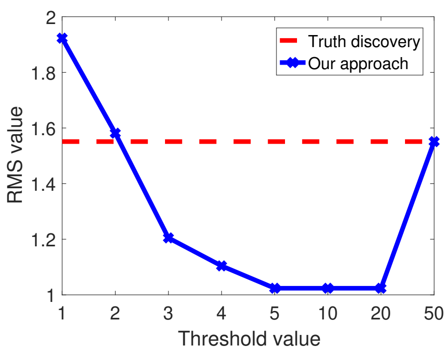

Fig. 2: The affect of various threshold values on our approach.

3) Performance Metrics: In the simulation, the performance metric is root mean square (RMS) value:

$$\text{RMS} = \sqrt{\frac{1}{|\mathcal{N}|}\sum_{i\in\mathcal{N}}(\hat{v}_i^* - v_i^*)^2} \tag{12}$$

which measures the distance between the estimated truth and the ground truth. The larger the RMS value is, the utility of the algorithm is smaller.

4) Baseline Algorithms: The baseline algorithm is truth discovery algorithm (Algorithm 1 with input $\{v_i^k\}_{i\in\mathcal{N},k\in\mathcal{K}\cup\mathcal{M}}$). We compare our approach with the baseline algorithm under the data poisoning setting (Algorithm 2). By comparing the baseline algorithm, we could prove the robustness and accuracy of our approach.

B. Simulation Results

In this part, we evaluate the robustness of our approach by comparing the baseline algorithm in terms of accuracy (RMS value).

1) Malicious Worker Ratio: We firstly evaluate the robustness of our approach under the settings with different malicious worker ratios. Since the ratio is usually less than 20% due to attackers' limited capability, we change the number of malicious workers from 0 to 4 in our simulation. In this simulation, we fix the threshold value T to be 5. The evaluation results are plotted in Figure 1. Figure 1 shows that with more malicious workers involved, the accuracy of the truth discovery algorithm will decrease. Furthermore, our approach could defend against the data poisoning attacks and the accuracy of our approach is even better than that one without attack (malicious worker ratio = 0%). Even though there is no attacker, there are also some normal workers with low quality. Our approach could remove those workers, which would decrease the utility of truth discovery algorithm. Thus, our approach is robust against both attacks and the impact from the data of low quality.

2) Threshold Value: Here we evaluate the robustness of our approach when we change the threshold value T used in our approach. In this simulation, we fix the malicious worker ratio to be 20%. The threshold value is important since it decides which data should be removed. As shown in Figure 2, when the threshold value is small, the utility of our approach is low (even lower than the utility after attack). The small threshold value enables our algorithm to remove both malicious workers and some workers of good quality, thus the algorithm has poor performance. When we choose the threshold value within $[3, 20]$, our approach has better utility than the traditional method, demonstrating our robustness against attacks. When we set the threshold to be 50, our approach has the same result as the truth discovery under data poisoning attacks since with high threshold value, our approach loses the capability to remove those malicious workers.

VI. RELATED WORK

There have been tremendous research efforts on truth discovery. Some work considers a semi-supervised method by utilizing the available labeled truth to guide source reliability estimation and truth inference [25]–[27]. However, the ground truth is usually difficult to obtain in practice and thus we have to use unsupervised method to estimate the truth. In unsupervised settings, the truth discovery problem is modeled as an optimization problem [12], [20]–[22]. They solve the optimization problem by an iterative method [4], [5], [7], [8], where truth estimation and weight estimation are conducted iteratively until the results converge. There are also some work considering a probabilistic graphical model [9], [10], [28]. In [4], the authors propose a TruthFinder algorithm which uses Bayesian analysis to estimate the truth and the reliability,. In AccuSim [5], [6], the authors assume the source dependency and use an implication function to capture the similarity between the observations to help truth estimation. CRH proposed in [12] considers different types of data and adopts different types of distance functions to capture the characteristics. Reputation-based data aggregation has also been investigated to discover the truth value. Kerchove and Dooren propose an iterative filtering algorithm in reputation system to infer the reputation of sources [13]. Based on the iterative filtering method, there are a number of published studies trying different discriminant functions [15], [16].

It is commonly believed that truth discovery algorithms or reliability based data aggregation methods are robust to the impact from unreliable data or attacks. However, some attack scenarios could still greatly disturb the truth value.

In [14], a collusion attack has been proposed to disturb the truth discovery result, and an new initialization method is used to defend such collusion attacks. In [18], an optimal data poisoning attacks strategy in truth discovery is proposed to maximize the attack utility and disguise the malicious workers as normal workers, but they only focus on the setting where the data is categorical. In our work, we investigate such optimal data poisoning attacks in continuous data setting and propose our robust approach by estimating the error level of each worker and removing those workers with high error level before the data aggregation. Besides, there are also some works focusing on data poisoning attacks or the countermeasures in crowdsensing application [29]–[31] and machine learning [17], [32]–[34].

VII. Conclusion

In this paper, we have investigated the data poisoning attacks on truth discovery process and proposed a robust approach to such attacks. We have analyzed the impacts of data poisoning attacks and demonstrated that existing truth discovery processes could suffer from high utility losses from such attacks. To mitigate the impacts and improve the utility of the truth discovery results, we have designed a robust truth discovery approach that filters out malicious sources before fusing the sensing results. Extensive experiments have been conducted on real-world data to demonstrate the accuracy and robustness of our approach.

References

[1] Y. Liu, X. Liu, S. Gao, L. Gong, C. Kang, Y. Zhi, G. Chi, and L. Shi, “Social sensing: A new approach to understanding our socioeconomic environments,” *Annals of the Association of American Geographers*, vol. 105, no. 3, pp. 512–530, 2015.

[2] C. C. Aggarwal and T. Abdelzaher, “Social sensing,” in *Managing and mining sensor data*. Springer, 2013, pp. 237–297.

[3] R. K. Ganti, F. Ye, and H. Lei, “Mobile crowdsensing: current state and future challenges,” *IEEE Communications Magazine*, vol. 49, no. 11, pp. 32–39, 2011.

[4] X. Yin, J. Han, and S. Y. Philip, “Truth discovery with multiple conflicting information providers on the web,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 6, pp. 796–808, 2008.

[5] X. L. Dong, L. Berti-Equille, and D. Srivastava, “Integrating conflicting data: the role of source dependence,” *Proceedings of the VLDB Endowment*, vol. 2, no. 1, pp. 550–561, 2009.

[6] X. Li, X. L. Dong, K. Lyons, W. Meng, and D. Srivastava, “Truth finding on the deep web: Is the problem solved?” in *Proceedings of the VLDB Endowment*, vol. 6, no. 2. VLDB Endowment, 2012, pp. 97–108.

[7] A. Galland, S. Abiteboul, A. Marian, and P. Senellart, “Corroborating information from disagreeing views,” in *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 2010, pp. 131–140.

[8] J. Pasternack and D. Roth, “Knowing what to believe (when you already know something),” in *Proceedings of the 23rd International Conference on Computational Linguistics*. Association for Computational Linguistics, 2010, pp. 877–885.

[9] B. Zhao and J. Han, “A probabilistic model for estimating real-valued truth from conflicting sources,” *Proc. of QDB*, 2012.

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

[11] D. Wang, L. Kaplan, H. Le, and T. Abdelzaher, “On truth discovery in social sensing: A maximum likelihood estimation approach,” in *Information Processing in Sensor Networks (IPSN), 2012 ACM/IEEE 11th International Conference on*. IEEE, 2012, pp. 233–244.

[12] 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 *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*. ACM, 2014, pp. 1187–1198.

[13] C. De Kerchove and P. Van Dooren, “Iterative filtering in reputation systems,” *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 4, pp. 1812–1834, 2010.

[14] M. Rezvani, A. Ignjatovic, E. Bertino, and S. Jha, “Secure data aggregation technique for wireless sensor networks in the presence of collusion attacks,” *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 1, pp. 98–110, 2015.

[15] Y.-B. Zhou, T. Lei, and T. Zhou, “A robust ranking algorithm to spamming,” *EPL (Europhysics Letters)*, vol. 94, no. 4, p. 48002, 2011.

[16] P. Laureti, L. Moret, Y.-C. Zhang, and Y.-K. Yu, “Information filtering via iterative refinement,” *EPL (Europhysics Letters)*, vol. 75, no. 6, p. 1006, 2006.

[17] M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li, “Manipulating machine learning: Poisoning attacks and countermeasures for regression learning,” *arXiv preprint arXiv:1804.00308*, 2018.

[18] C. Miao, Q. Li, H. Xiao, W. Jiang, M. Huai, and L. Su, “Towards data poisoning attacks in crowd sensing systems,” in *Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 2018, pp. 111–120.

[19] C. Miao, Q. Li, L. Su, M. Huai, W. Jiang, and J. Gao, “Attack under disguise: An intelligent data poisoning attack mechanism in crowdsourcing,” in *Proceedings of the 2018 World Wide Web Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2018, pp. 13–22.

[20] B. I. Aydin, Y. S. Yilmaz, Y. Li, Q. Li, J. Gao, and M. Demirbas, “Crowdsourcing for multiple-choice question answering.” in *AAAI*, 2014, pp. 2946–2953.

[21] Y. Li, Q. Li, J. Gao, L. Su, B. Zhao, W. Fan, and J. Han, “On the discovery of evolving truth,” in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 675–684.

[22] Q. Li, Y. Li, J. Gao, L. Su, B. Zhao, M. Demirbas, W. Fan, and J. Han, “A confidence-aware approach for truth discovery on long-tail data,” *Proceedings of the VLDB Endowment*, vol. 8, no. 4, pp. 425–436, 2014.

[23] J. F. Bard, *Practical bilevel optimization: algorithms and applications*. Springer Science & Business Media, 2013, vol. 30.

[24] X. L. Dong, L. Berti-Equille, Y. Hu, and D. Srivastava, “Global detection of complex copying relationships between sources,” *Proceedings of the VLDB Endowment*, vol. 3, no. 1-2, pp. 1358–1369, 2010.

[25] X. L. Dong, B. Saha, and D. Srivastava, “Less is more: Selecting sources wisely for integration,” in *Proceedings of the VLDB Endowment*, vol. 6, no. 2. VLDB Endowment, 2012, pp. 37–48.

[26] X. Liu, X. L. Dong, B. C. Ooi, and D. Srivastava, “Online data fusion,” *Proceedings of the VLDB Endowment*, vol. 4, no. 11, pp. 932–943, 2011.

[27] X. Yin and W. Tan, “Semi-supervised truth discovery,” in *Proceedings of the 20th international conference on World wide web*. ACM, 2011, pp. 217–226.

[28] J. Pasternack and D. Roth, “Latent credibility analysis,” in *Proceedings of the 22nd international conference on World Wide Web*. ACM, 2013, pp. 1009–1020.

[29] Y. Hu and R. Zhang, “Secure crowdsourced radio environment map construction,” in *2017 IEEE 25th International Conference on Network Protocols (ICNP)*. IEEE, 2017, pp. 1–10.

[30] S.-H. Chang and Z.-R. Chen, “Protecting mobile crowd sensing against sybil attacks using cloud based trust management system,” *Mobile Information Systems*, vol. 2016, 2016.

[31] S. Jagabathula, L. Subramanian, and A. Venkataraman, “Identifying unreliable and adversarial workers in crowdsourced labeling tasks,” *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 3233–3299, 2017.

[32] H. Xiao, B. Biggio, G. Brown, G. Fumera, C. Eckert, and F. Roli, “Is feature selection secure against training data poisoning?” in *International Conference on Machine Learning*, 2015, pp. 1689–1698.

[33] S. Alfeld, X. Zhu, and P. Barford, “Data poisoning attacks against autoregressive models.” in *AAAI*, 2016, pp. 1452–1458.

[34] B. Li, Y. Wang, A. Singh, and Y. Vorobeychik, “Data poisoning attacks on factorization-based collaborative filtering,” in *Advances in neural information processing systems*, 2016, pp. 1885–1893.