

Improving IoT Data Quality in Mobile Crowd Sensing: A Cross Validation Approach

Tie Luo, *Senior Member, IEEE*, Jianwei Huang, *Fellow, IEEE*,
Salil S. Kanhere, *Senior Member, IEEE*, Jie Zhang, *Member, IEEE*, and Sajal K. Das, *Fellow, IEEE*

Abstract—Data quality, or sometimes referred to as data credibility, is a critical issue in mobile crowdsensing (MCS) and more generally Internet of Things (IoT). While candidate solutions such as incentive mechanisms and data mining have been well explored in the literature, the *power of crowds* has been largely overlooked or under-exploited. In this paper, we propose a cross validation approach which seeks a *validating crowd* to ratify the *contributing crowd* in terms of the sensor data contributed by the latter, and uses the validation result to reshape data into a more credible posterior belief of the ground truth. This approach consists of a framework and a mechanism, where the framework outlines a four-step procedure and the mechanism implements it with specific technical components, including a *weighted random oversampling* (WROS) technique and a *privacy-aware trust-oriented probabilistic push* (PATOP²) algorithm. Unlike most prior work, our proposed approach augments rather than redesigning existing MCS systems, and requires minimal effort from the crowd, making it conducive to practical adoption. We evaluate our proposed mechanism using a real-world MCS IoT dataset and demonstrate remarkable (up to %475) improvement of data quality. In particular, it offers a unified solution to reconciling two disparate needs: reinforcing obscure (weakly recognizable) ground truths and discovering hidden (unrecognized) ground truths.

Index Terms—Data quality, Internet of Things, exploration-exploitation tradeoff, chance-constrained programming, Kullback-Leibler divergence, privacy, trust, crowdsourcing.

1 INTRODUCTION

MOBILE crowdsensing (MCS) is a key enabler of the Internet of Things (IoT) by connecting physical objects or “things” to the cyberspace via the medium of “humans-as-sensors”. By leveraging personal sensing devices such as smartphones, wearables, car-borne and soon drone-borne sensors, MCS significantly accelerates the permeation of IoT as compared to the alternative of dedicated sensor deployment by governments and businesses.

However, the issue of *data quality*, or sometimes referred to as *data credibility*, presents a fundamental challenge to MCS and IoT in general. The challenge arises from the fact that the data sources—the *contributing crowd* who own the IoT devices—are barely controllable, unevenly skilled, and hardly accountable. In the literature, a wide variety of candidate solutions have been proposed, taking approaches such as incentive mechanism design [1]–[7], quality and

trust assessment [8]–[12], truth finding [13]–[15], and so on. What is in common is that these approaches all introduce some *exogenous* forces or tools while having overlooked the “power of crowds” per se [16], which could otherwise be exploited to a fuller extent.

In this paper, we propose a *cross validation* (CV) approach to address the data quality issue from a perspective different than prior work. This approach seeks a *validating crowd* to ratify the *contributing crowd* in terms of the sensor data contributed by the latter, and uses the validation result to reshape data into a more credible posterior belief of the ground truth. It comprises a CV framework and a CV mechanism, where the framework outlines a four-step procedure with objectives and requirements, and the mechanism fulfills the framework with specific and concrete technical components. In particular, the mechanism uses a *weighted random oversampling* (WROS) technique to enable truth discovery, and a *privacy-aware trust-oriented probabilistic push* (PATOP²) algorithm that we propose based on the exploration-exploitation principle [17] and stochastic optimization.

One key motivation of our cross validation approach is to leverage the “side information” possessed by people, which includes (diversely) people’s domain knowledge, professional expertise, news learned from their social networks or public media, and so forth. This opens a much broader and powerful channel for acquiring information besides directly sensing the physical phenomenon or targets, thereby offering a more comprehensive perspective for improving IoT data quality.

Our cross validation leads to two key consequences. First, it relaxes the *spatio-temporal constraints* of direct and

- T. Luo is with the Institute for Infocomm Research, A*STAR, Singapore. E-mail: luot@i2r.a-star.edu.sg.
- J. Huang is with the School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, and the Department of Information Engineering, The Chinese University of Hong Kong. E-mail: jwhuang@ie.cuhk.edu.hk.
- S. S. Kanhere is with the School of Computer Science and Engineering, The University of New South Wales, Australia. E-mail: salil.kanhere@unsw.edu.au.
- J. Zhang is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. E-mail: zhangj@ntu.edu.sg.
- S. K. Das is with the Department of Computer Science, Missouri University of Science and Technology, USA. E-mail: sdas@mst.edu.

Manuscript received XXXX...; revised XXXX...; accepted XXXX...
Copyright © IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

physical sensing, which requires IoT devices to be at specific locations within specific time windows and hence is rather restrictive. Second, it relieves the necessary burden of consuming sensing-related resources (esp. energy) which can be substantial, and is less prone to privacy leakage via sensing devices.

This paper makes the following contributions:

- We introduce a cross validation approach which offers a new perspective to improve IoT data quality by exploiting the power of crowds to a fuller extent.
- We present a framework that outlines the general procedure and requirements of performing cross validation, and design a mechanism that substantiates the procedure and fulfills the requirements. In particular, using a WROS technique and a PATOP² algorithm that we propose, the mechanism not only fulfills, with guaranteed success rate, the “hard” constraints imposed by the time-sensitivity of IoT applications, but also satisfies “soft” constraints on the trustworthiness of validation which concerns competency, honesty, and bias.
- Our proposed approach is conducive to practical adoption: (i) unlike most prior work, it does not lead to redesigning existing MCS/IoT systems (which otherwise jeopardizes prior investments), but rather *augments* such systems with a lightweight plug-in; (ii) it requires minimal effort from the validating crowd and zero user intervention when executing the mechanism, and is simple to implement; (iii) it does not assume any distribution of the underlying sensing phenomenon as in Bayesian approaches, nor make any assumption on strict human rationality as in game-theoretical studies; (iv) it is robust to common security threats such as collusion and Sybil attacks.
- Using a real-world IoT dataset, we demonstrate that the proposed cross validation mechanism leads to remarkable (up to 475%) improvement of data quality, which we quantify using both belief contrasts and the *Kullback-Leibler divergence*. In particular, our proposal offers a unified solution to reconciling two disparate needs: reinforcing obscure (weakly recognizable) ground truth, and discovering hidden (unrecognized) ground truths.

2 RELATED WORK

Data quality as a crucial issue in MCS and IoT in general, has attracted a large body of research work that tackles it from different angles.

2.1 Incentive mechanisms

This line of research designs incentive mechanisms in order to influence worker behaviors so that workers will produce high-quality data. Typical incentive mechanisms include auctions [2], [3], lotteries [6], trust and reputation systems [18], bargaining games [19], contracts [20], and market mechanisms [21]. For example, Jin et al. proposed Thanos [2] that incorporates quality of information (QoI) into an incentive mechanism based on reverse combinatorial auctions to achieve near-optimal social welfare. A simple endorsement web (SEW) [18] connects workers into

a socioeconomic network using a trust-based relationship, using both social and economic incentives to encourage high-quality contributions. Theseus [22] is a payment mechanism that improves data quality by counteracting workers’ strategic behavior of reducing sensing effort, so that the aggregated results calculated by truth discovery algorithms are more accurate. Kamar and Horvitz [7] used a consensus prediction rule to induce truthful reporting by comparing each worker’s report against the consensus of the other workers’ reports to calculate the payment for that worker. However, consensus based methods have inherent bias and [7] only applies to single-truth applications. On the other hand, Bayesian truth serum [4], [5] removes the bias by using a scoring method, and can apply to multi-truth applications with subjective answers. However, it requires each worker to explicitly predict the distribution of all the other workers’ reports, which restricts its practicality. For a survey on incentive mechanisms, the reader may refer to [1].

2.2 Quality assessment

Unlike incentives, this line of work takes the contributed data as given, and focuses on evaluating the quality of data or the trustworthiness of workers so as to make informed decisions such as which data or workers to trust. Kantarci et al. [8] assessed the trustworthiness of both workers and their contributed data by combining centralized reputation value with individual vote-based collaborative reputation values. Wu et al. [9] proposed an EndorTrust system that not only assesses but also predicts the trustworthiness of workers without requiring prior contributions from them. This is achieved by using a trust-based worker relationship together with the machine learning technique of collaborative filtering. Huang et al. [11] used the Gompertz function to calculate device reputation scores as a reflection of the trustworthiness of data contributed by that device. Amintoosi and Kanhere [12] proposed a trust framework that uses fuzzy logic to combine two quantities to obtain a final quality assessment of each contribution. One is the quality estimate of the sensor readings contributed by each worker, and the other is the trust score of each worker which is calculated using their social attributes.

2.3 Truth finding

Like quality assessment, this thread of research also takes the indigenous data as given, but it focuses on finding the real truth from the large amount of noisy data, typically using data mining techniques. For example, Wang et al. [13] uses the expectation-maximization algorithm to obtain the maximum likelihood estimate of the probability that a MCS measurement is true, where the measurement must be *binary*. Davami and Sukthankar [14] aimed to predict the true occupancy of parking lots based on crowdsourced data, by combining multiple trust-based data fusion techniques using AdaBoost. Gisdakis et al. [15] proposed a framework called SHIELD to perform outlier detection, which is essentially the opposite of truth finding. It combines Dempster-Shafter theory and data mining to achieve desirable accuracy in the presence of a significant portion of outliers. However, the used complex machine learning model requires a large amount of training data as well as cumbersome private key configuration and operation.

2.4 Our approach

Our proposed approach does not belong to any of the above categories. Instead, on top of the original crowdsensing, it introduces another layer of crowdsourcing which exploits the power of crowds [16] to a fuller extent. This approach does not have to replace or preclude existing solutions, but rather allows them to achieve better result by reshaping the original (possibly obscure or misleading) data into a more trustworthy representation of the reality, before applying existing methods. Meanwhile, it can also work as a standalone solution without relying on exist methods.

Regarding applicability and assumptions, unlike most work such as [7], [13], [14], our approach applies to sensor measurements regardless of whether they are binary or multi-valued, discrete or continuous, and whether there are a single or multiple ground truth(s). Moreover, it does not assume the distribution of underlying sensing phenomena, nor any common prior held by crowdworkers like in Bayesian approaches, nor strict human rationality as in game-theoretical studies (e.g., [4], [5]).

Our approach is also different from peer rating as used by some online Q&A and product review platforms. This will become evident in Section 3 (Step 1).

3 THE CROSS VALIDATION FRAMEWORK

This framework describes a four-step procedure for performing cross validation.

3.1 Step 1: Data presentation and Form of verification

The objective of this step is to determine a proper form for presenting the original sensor data to the validating crowd, and a proper form of verification to be performed by the crowd. The following requirements need to be satisfied:

- Due to the nature of crowdsourcing, both data presentation and verification forms must to be easy to comprehend and handle by the validating crowd.
- The forms should *enable* timely verification due to the *time-sensitivity* of MCS and many other IoT applications, where the value of sensor data decays over time.

It is instrumental to look at a few candidate solutions for a more concrete understanding. One solution is to publish the dataset in the raw (e.g., text or tabulated) form or a summarized (e.g., graphic) version at a public venue such as a website, and request visitors to assess in a certain way (e.g., write a review or vote a poll). This is most common and has been adopted by many review platforms (e.g., Amazon, TripAdvisor, Yelp, Glassdoor) and online Q&A forums (e.g., Stackoverflow, Quora). However, such an opportunistic and ad hoc method is not compatible with the time-sensitivity of MCS/IoT, and its open nature also hinders quality control.

A variation is to present the same form of data to a dedicated group of “elite users” who may be able to provide timely and qualified validation. However, the sheer size of a dataset would still be overwhelming to each validator, letting alone how difficult and costly the recruitment of elite users would be. In addition, this and the previous methods are both prone to a *range-bias* problem: when facing a set of data for evaluation, people tend to favor majority values, or prefer intermediate over boundary values.

Another remedy is to partition the original dataset into smaller subsets for validators to evaluate one subset each, and then aggregate the evaluation results into an overall assessment of the original dataset. For each validator who is given a subset, she may be asked to (i) assign a proper score to each value, (ii) rank all the values, or (iii) pick the “best” value. For this method, first note that option (i) is a generalized (and hence harder) version of (ii) and (iii). Second, aggregating the evaluation results for (ii) or (iii) is in fact the classic *preference aggregation* problem in social choice theory [23]. Unfortunately, although decades of research has achieved promising accomplishments such as *Borda count* and *Condorcet winner*, this problem still remains largely open. For example, finding a Kemeny optimal ranking over m complete ranked lists of n candidates is NP-hard [24], and in our case, it is even harder because we need to aggregate *incomplete* ranked lists (over subsets). Moreover, there is no immediate answer to how to partition a dataset so that the subsets can properly represent the original dataset. Finally, the range-bias issue still exists, albeit milder.

3.2 Step 2: Quest for validation

The objective of this step is to recruit a validating crowd and solicit for their assessment on the sensor data (presented in a form determined by Step 1).

Implementing this step needs to address the following issues tactically:

- How to *perform* timely verification, i.e., quickly recruit a validating crowd and obtain a sufficient number of validation results, to satisfy the time-sensitivity?
- How to ensure good quality of the validation results? The “quality” can have comprehensive semantics as to cover competency, honesty, bias, etc.
- How to handle privacy and security aspects given that interacting with people is susceptible to these concerns?

3.3 Step 3: Consolidation

Given the validation results acquired in Step 2, and the original IoT sensor data, this step is to consolidate these two heterogeneous datasets to obtain a better representation of reality, for example a more credible posterior belief of the ground truth.

This is analogous to the preference aggregation problem discussed in Step 1. But due to the NP-hardness, one needs to devise a feasible solution.

3.4 Step 4: Compensation

Essentially, the proposed cross validation approach over-lays an additional layer of crowdsourcing over the original crowdsensing. Therefore, incentives as a crucial element in crowdsourcing [25] need to be handled, and this last step is meant to close this loop.

However, besides addressing this issue for the validating crowd, note that the original contributing crowd is also affected. This is because the final outcome of MCS as obtained in Step 3 would be different from the original sensor data, which means that we would have a better estimate of the quality of contributed data after cross validation. Therefore, a re-evaluation of the contributing crowd is also necessary.

4 THE CROSS VALIDATION MECHANISM

In this section, we design a cross validation mechanism that implements the framework outlined above, and fulfills the requirements the framework stipulates. An overview of this mechanism is given in Fig. 1.

4.1 Profiling (Step 1-A)

As explained in Section 3.1, a massive crowdsensed dataset would be overwhelming to validators. Therefore, we first create a profile that can concisely represent the original dataset without loss of critical information. Then in Section 4.2, we apply a special sampling technique to this profile to extract values to present to validators.

The said profile, denoted by $\mathcal{F} = (\mathcal{V}, \mathcal{P})$, consists of a set \mathcal{V} of representative values and a probability distribution \mathcal{P} of those values. To create this profile, the IoT cloud (or server) which stores the crowdsensed dataset \mathcal{O} first creates a histogram of \mathcal{O} with an appropriate resolution (bin width) determined by the specific application. For instance, a traffic monitoring application may use a bin width of 3mph while a noise mapping application may find 5dB suitable. Let us index these bins by $i = 1, 2, \dots, n$.

Next, the cloud designates for each bin i a representative value v_i , which can be the mean or median of the bin, or any other quantile when the resolution is sufficiently high. Thus we obtain the representative value set $\mathcal{V} = \{v_i | i = 1, 2, \dots, n\}$.

Finally, the cloud computes a probability measure $p_i = \kappa_i / \sum_{j=1}^n \kappa_j$ for all i , where κ_i is the volume of, or the number of data points in, bin i . Hence we obtain the probability distribution $\mathcal{P} = \{p_i | i = 1, 2, \dots, n\}$.

Sometimes we also refer to \mathcal{P} as the *interim belief* to differentiate from *prior belief* which is a presumed distribution before observing the sensed data. Correspondingly, we refer to the final, consolidated distribution as the *posterior belief*.

4.2 Sampling (Step 1-B)

Given the profile $\mathcal{F} = (\mathcal{V}, \mathcal{P})$, we need to determine how to present it to validators. Based on our deliberation in Section 3, we eventually take up a minimalist design: pick a *single* representative value from \mathcal{V} , show it to a validator and ask her to give a single rating, by choosing one out of a few options such as {"Agree", "Disagree"}. This method requires little effort from a validator and circumvents NP-hardness when consolidating results. It also facilitates quality and time control as will be elaborated in Section 4.4.

This section deals with how to pick representative values from \mathcal{V} , for which we use a *weighted random oversampling* (WROs) technique. This technique samples \mathcal{V} with replacement using a weights vector $S = \{s_i | i = 1, 2, \dots, n\}$ such that each $v_i \in \mathcal{V}$ is sampled with a probability proportional to its weight s_i . The sample size m will be much larger than the population size $n = |\mathcal{V}|$, hence "oversampling".

The reason for using WROs is that it gives an MCS/IoT system flexibility to configure S to meet different needs. For example, we are particularly interested in discovering *hidden* truth or "scavenging outliers". That is, conventional statistical methods generally ignore minority events or classify them as outliers, but this is risky as data is often insufficient

for us to draw such conclusions with confidence. Furthermore, even a large number of observations can sometimes be fallacious, for example due to sensor drift or miscalibration [26], environmental causes (e.g., urban canyon and tunnel shadowing), or large-scale security breach [27]–[29].

Therefore, minority events should not be "conveniently" ignored and they could have contained the ground truth. In this regard, WROs allows us to make a discovery, by assigning higher priority to minority events so as to expose them to more validation opportunities. Specifically, we use two weight configurations as follows.

- *Reverse sampling*: $s_i = d - p_i$ where d is a constant that ensures $s_i \geq 0$.

It is tempting to choose $d = 1$ since it seems to be most natural. However, a closer look reveals that it will blunt the multiplicative difference between s_i 's for small p_i 's. For example, $p_i = 0.2$ is twice of $p_i = 0.1$ but the corresponding $s_i = 0.8$ is close to $s_i = 0.9$ as sampling weights. Hence, the best reverse-weight vector S is one that "mirrors" \mathcal{P} with respect to its "waistline" $(\min \mathcal{P} + \max \mathcal{P})/2$, which translates to $d = \min \mathcal{P} + \max \mathcal{P}$. This configuration is illustrated in Fig. 2.

- *Inverse sampling*: $s_i = 1/p_i$. This results in a greater differentiation between majority and minority events. Events of $p_i = 0$ ($\kappa_i = 0$) are excluded.

In addition, we also include the following two configurations for comparison:

- *Uniform sampling*: $s_i = 1$. Hence all the v_i will be validated equally likely.
- *Proportional sampling*: $s_i = p_i$. Under this setting, more frequently appeared values will be validated more times.

4.3 Quest for Validation: Stochastic Optimization (Step 2-A)

Recall that Step 2 deals with the most critical problem: recruiting a validating crowd and soliciting for assessments (ratings).

Definition 1 (Problem Statement). *The objective is to collect no less than m effective ratings below a shortfall probability θ by a deadline T_0 . Here, m is a number typically much larger than $n = |\mathcal{V}|$, an effective rating is one that is either positive or negative but not neutral, and shortfall means less than m (i.e., not successful).*

On top of these quantitative ("hard") requirements, it is also desirable to have the following qualitative ("soft") properties:

- *Competency*: each effective rating should come from a competent validator, i.e., one who possesses the relevant information or domain knowledge.
- *Honesty*: a validator's rating should truly reflect her opinion.
- *Bias*: while humans are inevitably biased in general, such effect should be curbed as much as possible.

In a word, we aim to only collect *trustworthy* ratings.

To obtain an analytical solution to the above problem (with the hard constraints), suppose we had access to the

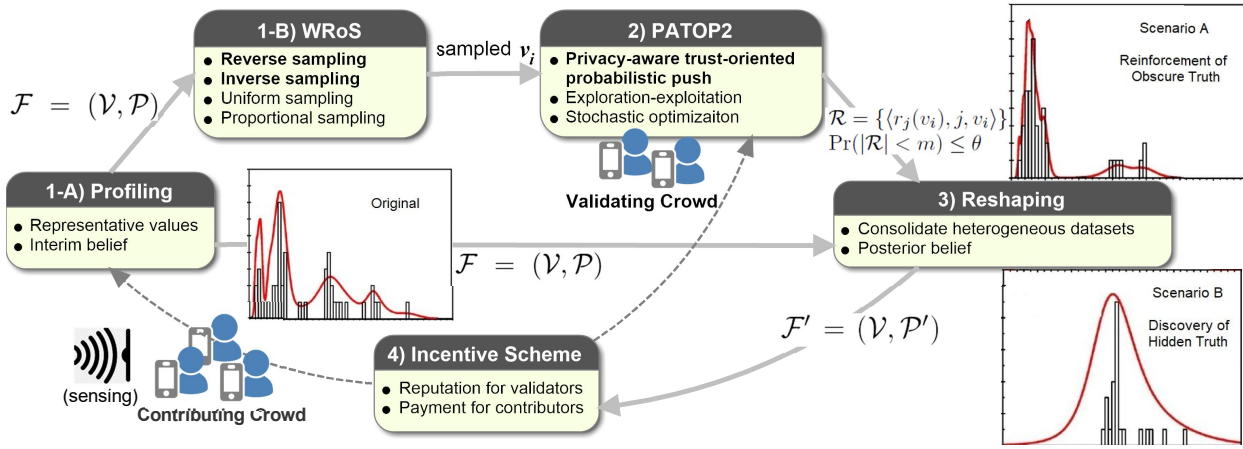


Figure 1: Overview of our proposed cross validation mechanism. The input is the original crowdsensed data which will first be profiled as \mathcal{F} , depicted by the plot “Original”. The output is an improved profile \mathcal{F}' illustrated by the plot “Scenario A” or “Scenario B”: the former reinforces the ground truth when obscure (albeit weakly recognizable) in “Original”, and the latter scavenges the ground truth when unidentifiable in “Original” due to being hidden by noise. Our mechanism works for both scenarios without being told which it is in.

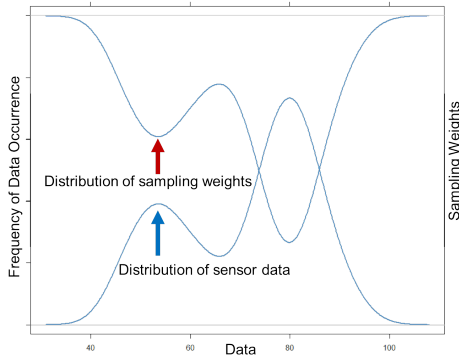


Figure 2: Weighted random oversampling (WRoS) allows for prioritizing over majority and minority events to meet different needs.

conditional probability of obtaining an effective rating from an arbitrary validator who has been recruited. Denote this probability by ξ which we assume to be a random variable rather than a constant in order to capture the heterogeneity among workers. Then, we transform the above problem into one that aims to find the minimum number of workers, y , to be recruited such that the shortfall probability of obtaining m effective ratings is no greater than θ . Formally,

$$\begin{aligned} \min_{0 \leq y \leq |\Psi|} y \\ \text{s.t. } \Pr(\xi y < m) \leq \theta \end{aligned} \quad (1)$$

where Ψ is the set of all the workers available for recruiting (e.g., all the users registered on a crowdsourcing platform such as Amazon Mechanical Turk [30]).

Problem (1) is a stochastic optimization problem because the constraint contains a random variable, ξ . We solve it using chance constrained programming (CCP) [31].

First, we rewrite the constraint of (1) as

$$F_{\xi}\left(\frac{m}{y}\right) \leq \theta \quad (2)$$

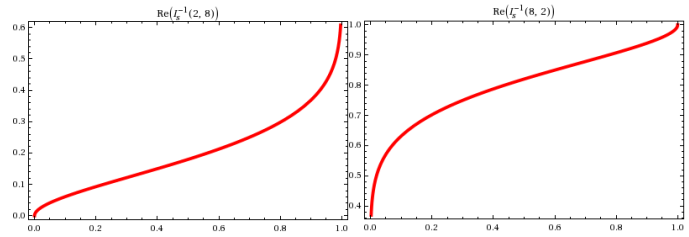


Figure 3: Left: $I_{\theta}^{-1}(\alpha = 2, \beta = 8)$. Right: $I_{\theta}^{-1}(\alpha = 8, \beta = 2)$.

where $F_{\xi}(\cdot)$ is the cumulative distribution function (c.d.f.) of ξ . Next, we introduce the *quantile function* of ξ , which is defined as

$$Q_{\xi}(\theta) = \inf\{x \in \mathbb{R} : F_{\xi}(x) \geq \theta\}. \quad (3)$$

Since $F_{\xi}(\cdot)$ is a monotone increasing function, it follows that $m/y \leq Q_{\xi}(\theta)$, i.e., the solution is given by

$$y^* = \frac{m}{Q_{\xi}(\theta)}. \quad (4)$$

To have an explicit form of (4), consider two common cases. If ξ follows a Beta distribution parameterized by α and β , i.e., $\xi \sim \text{Be}(\alpha, \beta)$, then its c.d.f. is the *regularized incomplete Beta function*, i.e., $F_{\xi}(x) = I_x(\alpha, \beta)$. In this case, the optimal solution to (1) is

$$y_{\text{beta}}^* = \frac{m}{I_{\theta}^{-1}(\alpha, \beta)} \quad (5)$$

where $I_{\theta}^{-1}(\alpha, \beta)$ is the inverse of the regularized incomplete Beta function and can be computed by tools such as Matlab using the `betaincinv` function, or Mathematica using the `InverseBetaRegularized` function. For example, Fig. 3 plots $I_{\theta}^{-1}(\alpha, \beta)$ versus θ (X axis) for $(\alpha = 2, \beta = 8)$ and $(\alpha = 8, \beta = 2)$, respectively.

If ξ follows a Gaussian distribution as $\xi \sim \mathcal{N}(\bar{\xi}, \sigma^2)$ where $\bar{\xi} \in (0, 1)$, then since $(\xi - \bar{\xi})/\sigma \sim \mathcal{N}(0, 1)$, a similar derivation as from (2) to (4) yields $(\frac{m}{y} - \bar{\xi})/\sigma \leq \Phi^{-1}(\theta)$, or equivalently $y \geq m/(\sigma\Phi^{-1}(\theta) + \bar{\xi})$. Here, $\Phi^{-1}(\cdot)$ is the *probit*

function which is the quantile function for standard normal distribution. Hence, the optimal solution to problem (1) is given by

$$y_{gauss}^* = \frac{m}{\sigma\Phi^{-1}(\theta) + \xi}. \quad (6)$$

The probit function $\Phi^{-1}(\theta)$ can be computed using Z-table [32]. For example, $\Phi^{-1}(0.05) = -1.65$, $\Phi^{-1}(0.1) = -1.28$.¹

While having an analytical solution is desirable, the assumption of having precise knowledge of the distribution of ξ (i.e., type and associated parameters α, β or ξ, σ) limits practicality. Therefore, in the next section, we provide a more practical solution to the problem in Definition 1. In addition, it also satisfies the three soft constraints.

4.4 Quest for Validation: Heuristic Solution (Step 2-B)

This heuristic takes an *exploration-exploitation* approach² to predict and also leverage the conditional probability ξ . During the exploration phase, it “probes” a crowd and uses regression analysis to predict ξ by learning from the interaction with the probed crowd. During the exploitation phase, it launches another, more targeted, round of interaction with crowd based on the predicted ξ and other exploration results. Both of the interaction processes employ a “push” model (as opposed to the “pull” model used by most websites), which proactively approaches a tactically selected group of workers to seek their validation (i.e., ratings). The entire procedure is formulated as a *privacy-aware trust-oriented probabilistic push* (PATOP²) algorithm (see Algorithm 1 for pseudo-code), and is elaborated below.

4.4.1 Exploration

Crowd behaviors are highly dynamic and uncertain when it comes to reacting to unsolicited requests. One may dismiss (decline) a request or may fail to notice it, and if she does respond, the response may be delayed arbitrarily and may not be an effective rating. Furthermore, we need to collect at least m effective ratings by a certain deadline T_0 , without abusing the crowd by simply bombarding the entire or an arbitrarily large crowd with the requests.

To overcome this challenge, we use an exploration phase to learn the crowd behaviors online, in order to reduce the uncertainty. Unlike most exploratory online algorithms, where an initial set of data has to be sacrificed to establish a reference for comparison and cannot be utilized, our exploration process is fully efficient in the sense that no data collected from it will be discarded.

We designate the period $[0, t^*]$ as the exploration phase and $[t^*, T_0]$ the exploitation phase. At time $t = 0$, we select m workers and send each of them a rating task. How the m workers are selected and what a rating task looks like will be described in Section 4.4.3. For now, let us focus on the regression-based prediction.

The response dynamics of the m workers under exploration can be characterized by two non-decreasing functions (with unknown forms) depicted in Fig. 4. During the exploration phase, we construct a regression dataset \mathcal{D} by

1. σ is sufficient small so that $\sigma\Phi^{-1}(\theta) + \xi > 0$.

2. While it may sound resemblant to reinforcement learning and particularly multi-armed bandits (MAB), we will explain in Section 4.4.4 that the MAB model does not fit our problem.

Algorithm 1: Privacy-aware trust-oriented probabilistic push (PATOP²)

Input: All crowdworkers \mathcal{U} , contributors \mathcal{C} , profile $\mathcal{F} = (\mathcal{V}, \mathcal{P})$, target m , deadline T_0

Output: Effective ratings $\mathcal{R} = \{\langle r_j(v_i), j, v_i \rangle | r_j(v_i) \neq 0, j \in \mathcal{U}, v_i \in \mathcal{V}\}$

// Initialization:

- 1 $t \leftarrow 0, \Psi \leftarrow \mathcal{U} \setminus \mathcal{C}, \mathcal{R} \leftarrow \emptyset, \mathcal{D} \leftarrow \emptyset$
- // Exploration:
- 2 Select a set \mathcal{M}_1 of m workers from Ψ using Eq. (10)
- 3 **for** each $j \in \mathcal{M}_1$ **do**
- 4 Sample one $v_i \in \mathcal{V}$ using a predetermined WRoS method (Section 4.2)
- 5 Wrap v_i in a rating task and push it to worker j to seek rating $r_j(v_i)$
- 6 **while** $t \leq T_0/2$ **do**
- 7 // collect effective ratings:
- 8 $\mathcal{R} \leftarrow \mathcal{R} \cup \{\langle r_j(v_i), j, v_i \rangle | r_j(v_i) \neq 0\}$
- 9 // construct regression dataset:
- 10 **if** $t \bmod \tau = 0$ **then**
- 11 $\mathcal{D} \leftarrow \mathcal{D} \cup \{t, |\mathcal{R}|\}$
- 12 $t++$
- 13 $m_Y(T_0/2) \leftarrow |\mathcal{R}|$ // no. of effective ratings at $t = T_0/2$
- 14 Predict $m_Y(t = T_0)$ to be $\hat{m}_Y(t = T_0)$ using function $\hat{m}_Y(t)$, which is the estimate of the target function $m_Y(t)$ and is obtained via regression over \mathcal{D}
- // Exploitation:
- 15 $\Psi \leftarrow \Psi \setminus \mathcal{M}_1$
- 16 Compute $m_{exploit}$ using Eq. (9)
- 17 Select a set \mathcal{M}_2 of $m_{exploit}$ workers from Ψ using (10)
- 18 **for** each $j \in \mathcal{M}_2$ **do**
- 19 the same as Lines 4–5
- 20 **while** $t \leq T_0$ **do**
- 21 the same as Line 7
- 22 $t++$
- 23 **return** \mathcal{R}

uniformly picking k samples over $[0, t^*]$, as $\mathcal{D} = \{\langle t_i := i \cdot t^*/k, m_Y(t_i) \rangle | i = 1, 2, \dots, k\}$, where $m_Y(t_i)$ is the number of workers who have responded with an effective rating by time t_i . We can then recover a function $\hat{m}_Y(t)$ via nonlinear regression over \mathcal{D} , which approximates the target function $m_Y(t)$, and thus predict (extrapolate) the target function value at the deadline to be $\hat{m}_Y(T_0)$.³

4.4.2 Exploitation

The exploration phase tells us two things. First, in expectation, we will be in short of $m - \hat{m}_Y(T_0)$ effective ratings by the deadline T_0 . Second, the “conversion rate” from approached workers to effective ratings at the end of a time window $[0, t]$ is

$$\hat{\xi}(t) := \hat{m}_Y(t)/m,$$

3. This can be done using, for example, the SciPy function `curve_fit()` or the Matlab function `interp1()`.

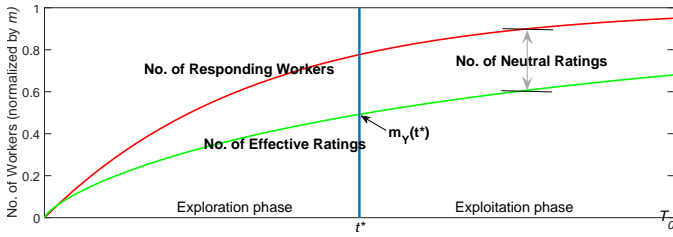


Figure 4: Modeling crowd dynamics for workers under *exploration*.

which is actually an estimate of the conditional probability ξ as a function of elapsed time t .

Thus for the exploitation phase which starts at t^* , we can determine the expected size of the crowd to approach as

$$\bar{m}_2 = \frac{m - \hat{m}_Y(T_0)}{\hat{\xi}(T_0 - t^*)}. \quad (7)$$

To cater for the randomness of $\xi(\cdot)$ with respect to the shortfall probability θ , we use the CCP method introduced in Section 4.3 to determine the actual size of crowd to approach, which we denote by m_2 , as follows. Assuming that the prediction error is Gaussian as is most common, we can directly apply (6) where y_{gauss}^* corresponds to m_2 , and on the right hand side of (6), we substitute m by \bar{m}_2 , $\bar{\xi}$ by $\hat{\xi}(\cdot)$, and

$$\sigma \leftarrow \sqrt{\chi^2/(k-1)}$$

which is the *corrected* sample standard deviation [33] where

$$\chi^2 = \sum_{i=1}^k |m_Y(t_i) - \hat{m}_Y(t_i)|^2, \quad t_i \in \mathcal{D} \quad (8)$$

is the sum of squared errors of regression. Thus putting all together, we have

$$m_2 = \frac{m(m - \hat{m}_Y(T_0))}{\hat{m}_Y(T_0 - t^*) \left(\sqrt{\frac{\chi^2}{k-1}} \Phi^{-1}(\theta) + \hat{m}_Y(T_0 - t^*) \right)} \quad (9)$$

Choice of t^* : t^* is the delimiter of the exploration phase and the exploitation phase. It affects the accuracy of m_2 as given by (9) as follows:

- In general, the larger t^* is, the more accurate $\hat{m}_Y(T_0)$ is. This is because $\hat{m}_Y(T_0)$ is an extrapolation of data collected over $[0, t^*]$ where $t^* < T_0$.
- A larger t^* , however, does not improve the accuracy of $\hat{m}_Y(T_0 - t^*)$ when $t^* \geq T_0/2$. This is because $\hat{m}_Y(T_0 - t^*)$ can be *measured* (rather than predicted) during the exploration phase when $t^* \geq T_0/2$.

Moreover, a larger t^* will lead to a shorter exploitation phase, which means that more responses (ratings) are more likely to arrive *after* deadline T_0 and hence be wasted.

Based on the above three considerations, we choose $t^* = T_0/2$ which strikes a reasonable tradeoff. It also allows us to use in (9) the measured $m_Y(T_0/2)$ rather than a predicted $\hat{m}_Y(T_0/2)$ via $\hat{m}_Y(t)$, which (the latter) is more prone to inaccuracy.

4.4.3 Validator Recruitment

Now we explain how we select the m workers in the exploration phase, whom we collectively denote by \mathcal{M}_1 , and the $m_{exploit}$ workers in the exploitation phase, denoted by \mathcal{M}_2 . This worker selection process is also called validator recruitment.

To recruit a set of validators \mathcal{M} from a pool of available workers Ψ , we assign each worker $j \in \Psi$ a weight

$$q_j(t) = \frac{1 - e^{-\lambda_j(t-t_j^-)}}{1 + e^{-wR_j}}, \quad \forall j \in \Psi. \quad (10)$$

With this assignment, we perform a *weighted sampling without replacement* over Ψ to obtain $|\mathcal{M}|$ validators, and push to each $j \in \mathcal{M}$ a rating task at time t . In the above,

$$\begin{cases} \Psi = \mathcal{U} \setminus \mathcal{C}, & \mathcal{M} = \mathcal{M}_1, & \text{if } t = 0, \\ \Psi = \mathcal{U} \setminus \mathcal{C} \setminus \mathcal{M}_1, & \mathcal{M} = \mathcal{M}_2, & \text{if } t = \frac{T_0}{2}. \end{cases} \quad (11)$$

where \mathcal{U} is the entire population of all the workers, and \mathcal{C} is the contributors of the original crowdsensed data.

Eq. (10) is the product of logistic function $1/(1 + e^{-wR_j})$ and $1 - e^{-\lambda_j(t-t_j^-)}$, which represent a trust component and a privacy component, respectively. Let us explain below.

Trust oriented: Every worker $j \in \mathcal{U}$ is associated with a reputation score $R_j \in \mathbb{R}$, which characterizes how reliable j as a validator is, based on the credibility (accuracy) of her past ratings. The logistic function (where $w > 0$ is a constant) makes it such that more reputable workers will have higher chance to receive rating tasks, in order to collect higher quality of ratings overall.

A rating task (Fig. 5) consists of a single representative value $v_i \in \mathcal{V}$ sampled using WRoS, a task description, and a list of rating options such as {"Agree", "Unsure", "Disagree"}.

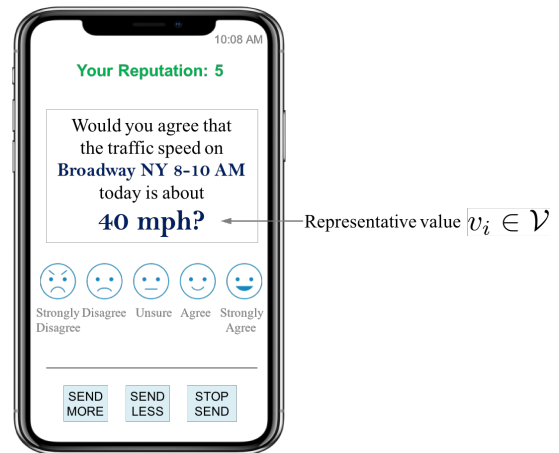


Figure 5: A rating task illustrated on a user interface.

Now, let us recall the three soft properties about trustworthiness: competency, honesty, and bias. We approach competency and honesty using R_j as part of our incentive scheme (Section 4.6): R_j only increases if j 's rating is consistent with the belief adjustment (toward the real truth), which requires the validator to be competent at this particular rating task *and* rate honestly; otherwise, R_j would decrease, constituting a *penalty*. The reputation R_j is initialized as 0 for new workers, and can go both positive and negative.

If a validator is not competent at a rating task but she is honest, she can choose the neutral rating to avoid being penalized. This is why our rating task should always keep a neutral rating option no matter how many (e.g., 3 or 5) options will be offered.

On the aspect of human bias, we incorporate two counter-measures. First, we exclude \mathcal{C} from \mathcal{U} . This eliminates contributors' biases toward their own respective contributions. Second, we ensure that no validator can provide more than one rating (to minimize the effect from any validator who does have bias), by sampling *without replacement* over Ψ to obtain \mathcal{M} and pushing each $j \in \mathcal{M}$ a single rating task.

Privacy aware: We have employed a proactive push model in order to suit the time-sensitivity of MCS/IoT and to have better quality control (as we can select validators). But on the other hand, a push model can be potentially *privacy-intrusive* if the push frequency is too high or not properly aligned with validators' personal preferences. We address this issue using two elasticity elements, one global and one individual.

The global elasticity element is the exponent $t - t_j^-$ in (10), where t_j^- is the last time when j received a rating task, or when she was enrolled as a worker if never received a rating task before. Hence, those who just received rating tasks will be much less likely to be pushed again; for those who did not, $g_j(t)$ does increase but the marginal increase is diminishing. Hence the overall effect is that the pushes to any one worker is naturally *spaced out* on the timeline.

The individual element is realized by a personal preference indicator λ_j . It is initialized as a constant (e.g., 1) and then updated as $\lambda_j \leftarrow \min(\lambda_j + \delta, \lambda_{max})$, $\lambda_j \leftarrow \max(\lambda_j - \delta, 0)$, and $\lambda_j \leftarrow 0$, respectively, when the validator j (optionally) chooses "Send me more", "Send me less", and "Stop sending" (see Fig. 5). Here, δ is the step size (e.g., 0.2), λ_{max} is a cap (e.g., 2) which prevents malicious users from abusing λ_j to offset their low reputation R_j .⁴

4.4.4 Comparison with MAB

Our exploration-exploitation approach may be reminiscent of the multi-armed bandit (MAB) problem [34]. However, there are key differences that set our problem apart from the MAB model, making its existing solutions not applicable.

In an MAB setting, there are multiple arms each associated with a random reward following an unknown and different distribution. An agent pulls an arm each time to receive a reward, and aims to maximize the total reward (or minimize the regret as compared to the optimal reward). Thus, the agent faces an exploration-exploitation dilemma: whether to explore (try) more arms or each arm more times in order to find the best arms, or to exploit (concentrate on) the seemingly most rewarding arms so far.

In an attempt to frame our problem under MAB, it seems plausible to model each worker or each group of workers as an arm. However, an arm like this does not have *repeatability*, and hence leaves no opportunity for exploitation after being

4. One could use a more sophisticated method such as a gradient-descent-like algorithm (which is still empirical) to adjust λ_j . However, we keep it simple for practicality and also because obtaining the "optimal" value or updating-model (if ever exists) of user preference is not critical to our problem.

explored. In addition, exploration on this type of arm does not reveal the outcome until the deadline, which also leaves no room for exploitation. Therefore, existing solutions do not apply and we must devise our own, as provided above.

4.5 Consolidation: Reshaping (Step 3)

Thus far, we have obtained a profile $\mathcal{F} = (\mathcal{V}, \mathcal{P})$ of the original MCS/IoT data, and a collection of effective ratings $\mathcal{R} = \{ \langle r_j(v_i), j, v_i \rangle \}$. The next step is to consolidate these two heterogeneous datasets into a (better) posterior belief of the ground truth.

To do so, we assign each rating option a score of $-L, -L + 1, \dots, -1, 0, 1, \dots, L - 1, L$ corresponding to its position in the list of the $2L + 1$ rating options, where 0 corresponds to the neutral rating. Then for each v_i , we separately aggregate positive scores and negative scores in terms of their absolute value normalized by L , as

$$\begin{aligned} g_i &= \frac{1}{L} \sum_j r_j(v_i) \mathbb{1}_{r_j(v_i) > 0}, \\ b_i &= -\frac{1}{L} \sum_j r_j(v_i) \mathbb{1}_{r_j(v_i) < 0}. \end{aligned} \quad (12)$$

Here we slightly abuse notation by using $r_j(v_i)$ to denote both a rating score and a rating option (e.g., "Agree").

Recall from Section 4.1 that $p_i = \kappa_i / \sum_{j=1}^n \kappa_j$ is the interim belief of how likely v_i is the ground truth (κ_i is the bin volume of i). It can be interpreted as κ_i out of $\sum_{j=1}^n \kappa_j$ contributors have "voted" for v_i to be the ground truth. Similarly, we can interpret (12) as, during cross validation, another g_i out of $g_i + b_i$ validators voted for v_i . Thus, the interim belief p_i can be reshaped to

$$\tilde{p}_i = \frac{\kappa_i + \eta \times g_i}{\sum_{j=1}^n \kappa_j + \eta \times (g_i + b_i)}, \quad \forall v_i \in \mathcal{V}$$

which aggregates the two groups of votes. Here, an additional factor η is introduced to allow for weighing a (full-score) rating against a direct data contribution. For example, one can set $\eta = 0.5$ if sensors are generally reliable, and $\eta = 1$ otherwise.

However, the above \tilde{p}_i is dominated by the larger of the contributing crowd and the validating crowd if they are very different in size. Therefore, we need another factor for *balancing* purposes, which leads to

$$\begin{aligned} \hat{p}_i &= \frac{\kappa_i + \eta \times \frac{g_i}{|\mathcal{R}|} \sum_{j=1}^n \kappa_j}{\sum_{j=1}^n \kappa_j + \eta \times \frac{g_i + b_i}{|\mathcal{R}|} \sum_{j=1}^n \kappa_j} \\ &= \frac{p_i + \eta \times \frac{g_i}{|\mathcal{R}|}}{1 + \eta \times \frac{g_i + b_i}{|\mathcal{R}|}}, \quad \forall v_i \in \mathcal{V}. \end{aligned} \quad (13)$$

The final posterior belief, p'_i , is then calculated by normalization:

$$p'_i = \hat{p}_i / \sum_{i=1}^n \hat{p}_i. \quad (14)$$

Thus we have obtained the reshaped profile $\mathcal{F}' = (\mathcal{V}, \mathcal{P}')$, where $\mathcal{P}' = \{p'_i | i = 1, 2, \dots, n\}$.

Robustness control: A subtle issue to address is the *imprecision of human perception*. That is, unlike sensor readings which are precise (if the sensors are reliable), human ratings

are largely based on their estimation which is generally imprecise. As a result, values near ground truth v^* are likely to receive similar positive ratings as v^* , which will create “humps”—blunt peaks that make ground truths less distinguishable—in a profile.

To be robust to human imprecision, we add a rectifying procedure before applying Eq. (13). First, construct a vector $\vec{\gamma} = (\gamma_i := \frac{g_i}{g_i + b_i} | i = 1, 2, \dots, n)$ and detect “humps” in $\vec{\gamma}$ by looking for sequences of *prominent local maxima*.⁵ Second, for each hump represented by a sequence $(\gamma_i | i = i_l, \dots, i_r)$, designate its gravity center as $i_c = \arg \max_{i \in \{i_l, \dots, i_r\}} \gamma_i$ (breaking tie using the mean). Third, for each hump, update g_i where $i = i_l, \dots, i_r$ to

$$g'_i = \begin{cases} g_i a_i^l, & \text{where } a_i^l = \frac{i_c - i}{(i_c - i_l + 1)^2}, & \text{if } i \in [i_l, i_c) \\ g_i a_i^r, & \text{where } a_i^r = \frac{i - i_c}{(i_r - i_c + 1)^2}, & \text{if } i \in (i_c, i_r] \\ g_i + \sum_{j=i_l}^{i_c-1} g_j \left(1 - a_j^l\right) + \\ \quad \sum_{j=i_c+1}^{i_r} g_j \left(1 - a_j^r\right), & \text{if } i = i_c. \end{cases} \quad (15)$$

The ratios a_i^l and a_i^r serve the purpose of shifting a major portion of each g_i to the “gravity mass” g_{i_c} , where the portion size is larger if i is closer to i_c (because the votes for such a v_i are more likely due to its closeness to v_{i_c}). On the other hand, b_i is kept unchanged because a negative vote means that the validator disagrees with this particular v_i and does not indicate what other value she would agree with. Hence eventually, we substitute g_i with g'_i when applying (13).

4.6 Compensation: Incentive Scheme (Step 4)

As pointed out by the framework in Step 4, we need to both compensate the validating crowd and re-evaluate the compensation for the contributing crowd. Below, we provide such an incentive scheme to close the loop.

Validating crowd: Given the reshaped profile $\mathcal{F}' = (\mathcal{V}, \mathcal{P}')$, we update the reputation R_j of each validator j who gave an effective rating $r_j(v_i) \neq 0$ on v_i , as

$$R'_j = R_j + \begin{cases} \frac{p'_i - p_i}{1 - p_i} \times \frac{r_j(v_i)}{L}, & \text{if } p'_i > p_i \\ \frac{p_i - p'_i}{p_i} \times \frac{r_j(v_i)}{L}, & \text{if } p'_i < p_i. \end{cases} \quad (16)$$

The gist of (16) is twofold. First, whether a validator j will gain or lose reputation is determined by whether her rating r_j is *consistent* with the belief adjustment $p'_i - p_i$, which can be positive or negative. Second, the amplitude of reputation gain or loss is determined by (i) the normalized belief adjustment (against p_i or $1 - p_i$), which measures the *impact* of cross validation, and (ii) her normalized rating score r_j/L , which measures how much her rating has contributed to the above impact.⁶

We remark that reputation has been widely adopted in practice as an incentive in the form of “digital currency”. On top of that, it can also be assigned monetary value such as vouchers or coupons, or other tangible benefits such as

entitling users to privileged services or the access to more profitable sensing tasks.

Contributing crowd: Denote by $\pi_c(u_c, \mathbf{u}_{-c})$ the payment to a contributor $c \in \mathcal{C}$ as stipulated by the original incentive scheme (without cross validation),⁷ where u_c is the quality of c ’s contribution, and \mathbf{u}_{-c} are the qualities of all the other contributors’ contributions. Suppose the data point contributed by c is represented by v_i (i.e., her contribution falls in the i -th bin in our profiling step). Then after cross validation, her payment π_c is revised to

$$\pi'_c = \pi_c \left(u_c \frac{p'_i(c)}{p_i(c)}, \mathbf{u}'_{-c} \right) \quad (17)$$

$$\text{where } \mathbf{u}'_{-c} = \left\{ u_{\tilde{c}} \frac{p'_i(\tilde{c})}{p_i(\tilde{c})} \mid \tilde{c} \in \mathcal{C} \setminus \{c\} \right\},$$

$p_i(c)$ and $p'_i(c)$ are just p_i and p'_i (Eq. (14)) with associated contributor explicitly indicated, and $p_{\tilde{i}}(\tilde{c})$ and $p'_i(\tilde{c})$ are defined similarly, in which \tilde{c} is the contributor of $v_{\tilde{i}}$. Hence, (17) means that the original incentive scheme π is treated as a black box (which gains us maximal generality) while only its input u_c is substituted by $u_c \frac{p'_i(c)}{p_i(c)}$ for all $c \in \mathcal{C}$. The rationale is that, since $p_i(c)$ and $p'_i(c)$ are the likelihoods of v_i being the ground truth before and after cross validation, respectively, $\frac{p'_i(c)}{p_i(c)}$ re-scales u_c according to c ’s validated (and presumably more accurate) quality of contribution.

Note that the revised payment π'_c (17) does not guarantee the same *total* payment. Hence if there is a fixed *budget constraint* to satisfy, one can simply normalize π'_c (17) to

$$\pi''_c = \frac{\pi'_c}{\sum_{\tilde{c} \in \mathcal{C}} \pi'_{\tilde{c}}} \sum_{\tilde{c} \in \mathcal{C}} \pi'_{\tilde{c}}, \quad \forall c \in \mathcal{C}. \quad (18)$$

5 PERFORMANCE EVALUATION

We evaluate our proposed cross validation mechanism using a real dataset from a transportation MCS application called *Mobile Century* [40] built by UC Berkeley. To date, this dataset remains one of the most comprehensive public GPS datasets for traffic monitoring research [41].

5.1 Dataset

The Mobile Century application used cellphone-borne GPS sensors to measure vehicular speeds on the California I-880 highway. It accumulated 8 hours of GPS trajectory data on a 10-mile stretch of I-880, and the dataset is accessible at [42].

Specifically, we use the virtual trip line (VTL) data which consists of 44,374 north bound (NB) speed records and 43,403 south bound (SB) speed records. Each such record contains a VTL ID, the timestamp of the GPS reading, the coordinate of the GPS sensor, and the vehicle speed (mph) when crossing the VTL.

7. There is a rich literature on incentive mechanism design for MCS, for example [2], [7], [35], [36]. For a comprehensive survey, see [1], [37]–[39].

5. This can be done using, for example, the SciPy function `find_peaks()` or the Matlab function `findpeaks()`.

6. This does not imply that a higher rating is always advantageous, because it simultaneously bears the risk of losing more reputation if it is opposite to the belief adjustment. Therefore, one should always rate in accord with her confidence level.

5.2 Simulation Setup

Putting our experiment into perspective, one can imagine that there is a grand pool of workers \mathcal{U} registered on Amazon mTurk, among which a set \mathcal{C} has participated in Mobile Century to contribute their GPS data to the above NB and SB datasets. Now, we aim to collect $m = 2,000$ effective ratings from \mathcal{U} below a shortfall probability $\theta = 0.1$ within deadline $T_0 = 1$ hour, as per our problem statement given in Definition 1.

We use the following user model to simulate worker behaviors. A worker reacts to a validation request with a delay that follows the exponential distribution with a 10-minute mean. Whenever a worker j reacts, she dismisses (declines) the request with probability $1 - a_j$ and responds with probability a_j , where $a_j \sim \text{Be}(2, 10)$ and hence the mean is 0.2. To respond (by giving a rating), she compares the value v_i contained in the task (e.g., 40mph as in Fig. 5) with her estimated or believed truth ν_j , and rates “Agree” (+1) if $|v_i - \nu_j| < 0.2\nu_j$ and “Disagree” (-1) otherwise ($L = 3$). Here, $\nu_j \sim \mathcal{N}(\nu^*, (0.15\nu^*)^2)$ where ν^* is the ground truth, which means that %95 of the estimates ν_j are within $\pm 30\%$ of the ground truth ν^* (negative ν_j will be regenerated). Workers who give such -1/+1 ratings only constitute 80% of all the workers who respond; the other 20% give the neutral rating (“Unsure”) because they either do not have a clear estimate ν_j or are simply not sure of what to rate.

In the consolidation or reshaping step (Section 4.5), $\eta = 0.75$ (cf. (13)).

5.3 Result of Profiling

We first profile the NB and SB datasets by following the procedure described in Section 4.1. We set the number of bins to 40 for a sufficiently fine-grained resolution (bin width is 2.175mph for NB and 2.025mph for SB traffic). The resulting profiles are presented in Fig. 6, which shows that the NB traffic has some ambiguity while the SB traffic is rather clear. Thus, we will focus on the NB dataset henceforth. Furthermore, for a more meaningful evaluation, we further obscure the data slightly by pruning the highest bin (at about 65mph) down to the average height of its two adjacent bins. Fig. 7 shows the final profile, where extreme values (above 80mph) are cleaned. This profile will go through the rest of the procedure of our cross validation.

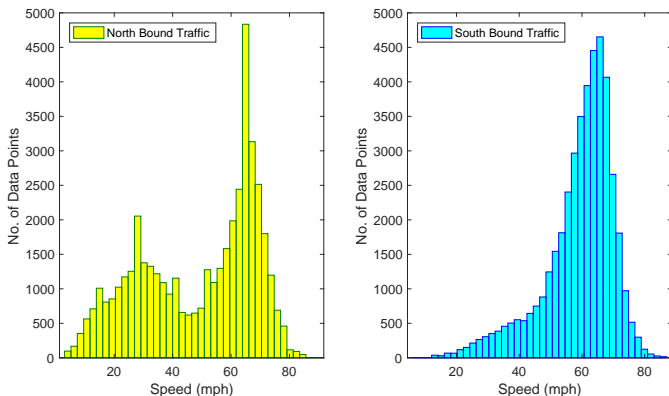


Figure 6: Profiling the original Mobile Century traffic data.

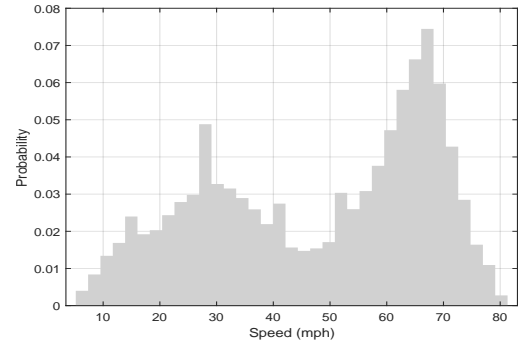


Figure 7: North bound traffic profile after pruning and cleaning.

5.4 Main Result

Apart from visual comparison, we also use the *Kullback-Leibler divergence* to characterize the change of belief (from interim to posterior) due to cross validation. The KL divergence measures the difference between two probability distributions, and in fact is the only measure of such difference that satisfies a set of desired canonical properties [43]. It is defined as

$$D_{KL}(\mathcal{P}'||\mathcal{P}) = - \sum_{i=1}^n p'_i \log \frac{p_i}{p'_i}. \quad (19)$$

where we adopt the same notation as of our case, so \mathcal{P} is the interim belief (based on original crowdsourced data) and \mathcal{P}' is the posterior belief (after cross validation). A larger value of D_{KL} indicates a larger information gain (hence a bigger change of belief).

5.4.1 Scenario A - Reinforcing Obscure Truth

We consider two typical scenarios. In Scenario A, the ground truth is obscure despite being somewhat recognizable. This corresponds to Fig. 7 where, even though 67mph may indeed be the ground truth, we would not be confident enough to draw that conclusion because its surrounding neighbors have similar probabilities too, and 28mph seems to be a promising truth as well.

After we carry out cross validation, the result is presented in Fig. 8. We see that the originally obscure truth is evidently reinforced: the interim belief about 67mph is increased from 0.0744 to the posterior belief of up to 0.1575 under different sampling methods, tantamount to a substantial increase of up to 111.7%, as tabulated in Table 1. Meanwhile, the other competitor, 28mph, becomes less salient, which further corroborates the prominence of 67mph as the ground truth.

Table 1. Improvement of Belief in Ground Truth: Scenario A

Interim belief	0.0744			
WROS method	Uniform	Proportional	Reverse	Inverse
Posterior belief	0.1175	0.1575	0.1155	0.09875
Enhancement	57.9%	111.7%	55.2%	32.7%

Among the four WROS methods, Proportional performs the best. This is because the interim belief about the truth 67mph is (indistinctly) the highest, so proportional sampling will allocate most validation opportunities to that value, which in turn receives most positive ratings to boost

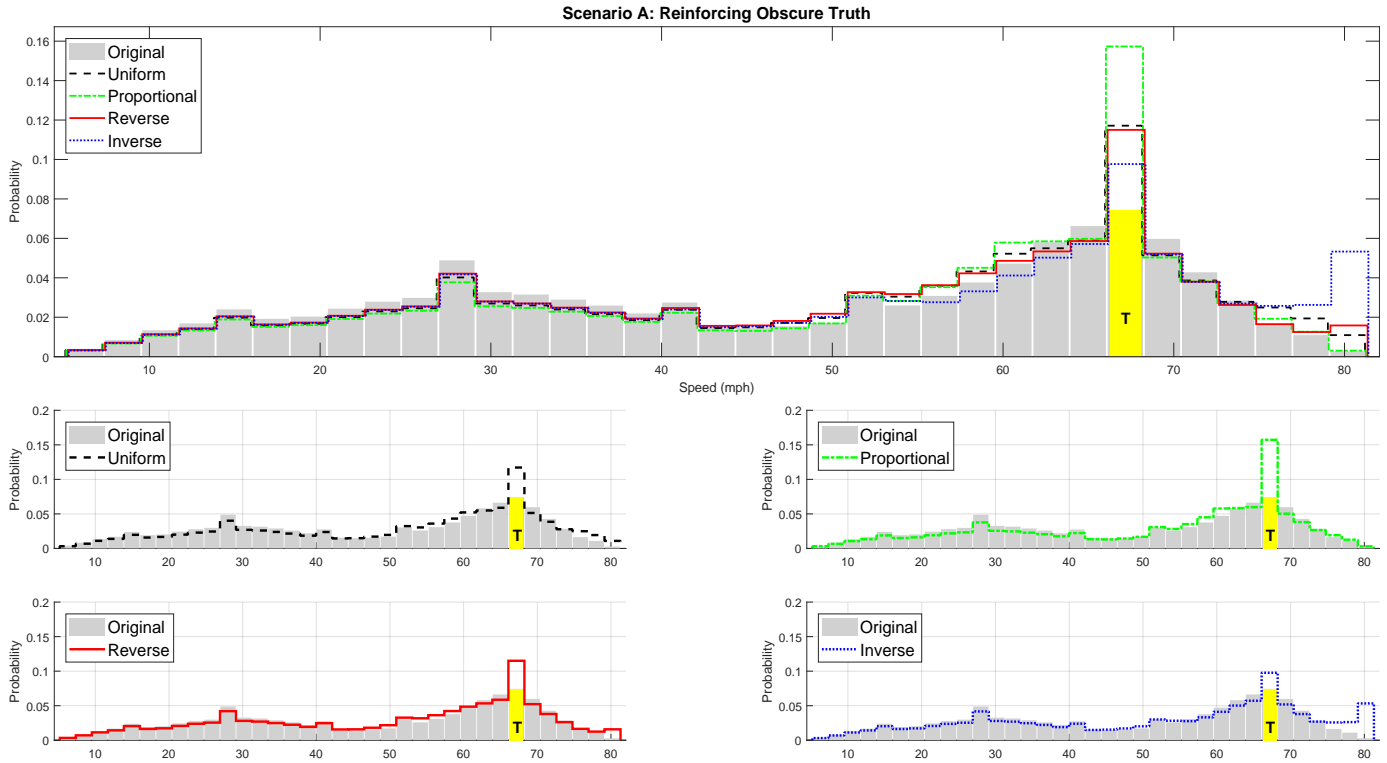


Figure 8: Reinforcement of obscure truth (Scenario A). The top figure gives the overall comparison, and the four sub-figures provide individual comparisons for better clarity. The yellow bar with letter “T” indicates the ground truth.

its prominence. By the same reasoning, *Inverse* allocates the least validation opportunities to the truth and thus receives the least boost. Moreover, we notice that *Inverse* creates a “heavy tail” near the right end. This is because those low-probability values were allocated many validation chances and, as they are also near the truth value, they received a good number of positive ratings too.

5.4.2 Scenario B - Discovering Hidden Truth

In Scenario B, the ground truth is buried among much noise and thereby become unidentifiable under conventional statistical methods. This corresponds to Fig. 7 when the ground truth is, for example, 45mph. In practice, such a scenario could be caused by low-quality or faulty sensors, unskilled or malicious contributors, sensor drift or miscalibration [26], environmental causes, security breach [27]–[29], etc.

Cross validation has the capability of discovering such hidden truth, as demonstrated by the results shown in Fig. 9. The interim belief about the hidden truth 45mph is boosted significantly from 0.016 to the posterior belief of 0.06–0.092, which is equivalent to a remarkable increase of 275–475% as shown in Table 2. Meanwhile, the two originally ostensible truth candidates (due to their prominence), 28mph and 67mph, are also mitigated to becoming even lower than the probability of 45mph (except for proportional sampling).

Table 2. Improvement of Belief in Ground Truth: Scenario B

Interim belief	0.016			
WRoS method	Uniform	Proportional	Reverse	Inverse
Posterior belief	0.08	0.06	0.092	0.0815
Enhancement	400%	275%	475%	409%

Among the four methods, *Reverse* performs the best. This is because it allocated more validation opportunities to the hidden truth than the ostensible truths (28 and 67mph), which enabled the ramp-up that “unearthed” the buried truth. Similarly, this explains why *Proportional* has the lowest improvement among the four methods. On the other hand, it is not intuitive why *Inverse* does not top all the methods, since it can be considered an “exaggerated” version of *Reverse*. The reason is that it wasted a lot of validation opportunities on very low-probability values (such as those near 4mph and 80mph), thereby leaving relatively less opportunities for the real hidden truth.

5.4.3 Choice of WRoS method

In practice, the challenge is that we do not have prior knowledge of what scenario we are facing when choosing the “best” sampling method. A trial-and-error approach (trying each method and picking the best) is not viable because each trial inevitably entails a large-scale outreach to crowd, which violates our objective of minimizing it. Therefore, we need to make the best choice in advance.

Our recommendation is *Reverse*, based on the trade-off as follows. First, it has the most superior discovering capability as demonstrated in Scenario B. Second, its reinforcement effect as demonstrated in Scenario A is good enough, which we quantify below:

- The relative strength of the obscure truth (67mph) against the most salient competitor (28mph) after reinforcement is $0.1155/0.0425 = 2.72$, which means that the true signal is nearly triple the second strongest signal, making it well distinguishable from noise. In

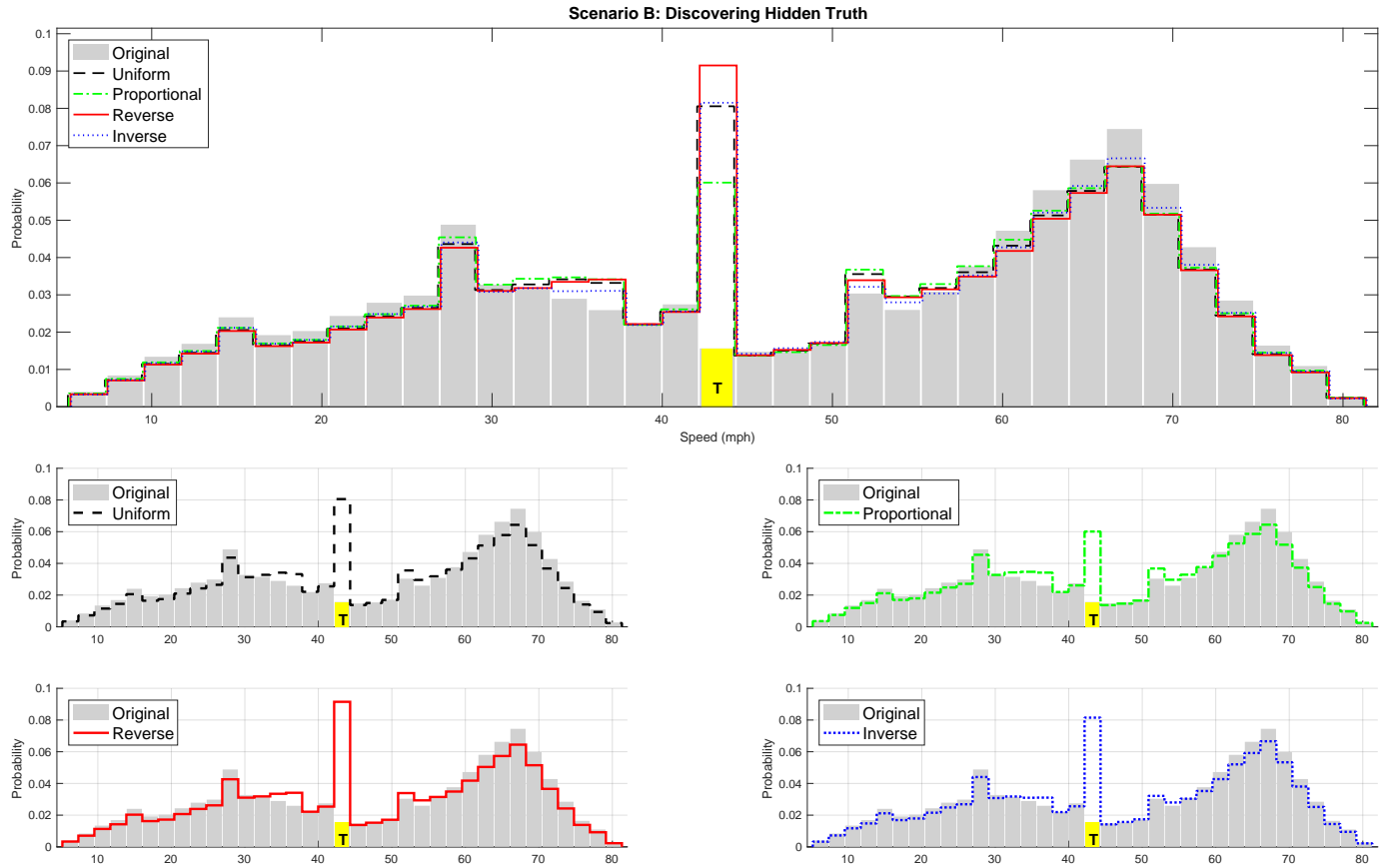


Figure 9: Discovery of hidden truth (Scenario B). The top figure gives the overall comparison, and the four sub-figures provide individual comparisons for better clarity. The yellow bar with letter “T” indicates the ground truth.

comparison, the relative strength as in the original dataset is $0.0744/0.0488 = 1.52$ only.

- The KL divergence, which measures the information gain, is higher for *Reverse* (3.22×10^{-2}) than for *Uniform* (3.14×10^{-2}), as tabulated in Table 3. Note that the KL value for *Inverse* (12.87×10^{-2}) is an outlier, because it is due to the heavy tail explained in Section 5.4.1. Moreover, the KL divergence for Scenario B is also provided in Table 3 for completeness, which corroborates the superiority of *Reverse*.

Table 3. Kullback-Leibler Divergence

WRoS method	Uniform	Proportional	Reverse	Inverse
$D_{KL}(\mathbf{A}) \times 100$	3.14	4.91	3.22	12.87
$D_{KL}(\mathbf{B}) \times 100$	7.50	4.35	9.45	7.37

6 DISCUSSION

6.1 Multiple Truths

Our cross validation approach is agnostic to the number of truths. While we demonstrate its performance with a single truth for clarity, it should have been evident that it applies to multi-truth applications as well. This is because we do not make any single-truth assumptions like in maximum likelihood estimation (MLE) and many other truth-finding studies in the literature.

On another note, the proposed approach also applies to both continuous and discrete types of data, which are unified by the profiling step (Section 4.1).

6.2 Resistance to Security Attacks

Due to the close interaction with people, a cross validation approach as such may be subject to the following security attacks. However, our mechanism is robust to them.

- *Collusion attack*: User rating systems commonly face this security threat where individual raters collude with product providers (in our case data contributors) to give unfair (usually higher) ratings; or in another case, a group of raters collaborate to give adverse or favorable ratings to a specific (set of) product(s). However, our proactive and probabilistic push combined with the randomness of WRoS, ensures that no one knows for sure who will be selected as raters and which product (data v_i) will be pushed to which rater. This makes it practically not feasible for the above collusion to succeed, whether individual or group based.
- *Sybil attack*: This refers to the case where a user creates or controls multiple accounts to gain unwarranted benefits, such as increasing the chance of being selected as a validator. However, our reputation-based recruitment grants Sybil accounts little chance, and even if one such account happens to be selected, it will be made worse off under our trust-oriented design if it provides biased or dishonest validation (cf. Section 4.4.3). Moreover, the

stochasticity of our push and sampling method makes it improbable for a Sybil account to validate its intended targets (e.g., friend or foe's contributions).

In any case, one cannot rate her own contributions because one of our anti-bias measures excludes contributors from the candidate pool of validators. This in effect disincentivizes most security attacks in the first place.

7 CONCLUSION

In essence, the cross validation approach proposed in this paper overlays another layer of crowdsourcing (for meta-data) on top of the original crowdsensing (for raw data). This offers a new perspective to tackle the long-standing data quality challenge for MCS-based IoT applications. By leveraging the diverse side information people possess, it alleviates the strict spatio-temporal constraints and the resource-consuming burden imposed by direct and physical sensing.

The approach is embodied by a cross validation mechanism, which hinges on a number of key components such as oversampling with WRoS, stochastic information solicitation using PATOP², and vote-based reshaping. It satisfies the hard constraints due to the time-sensitivity of IoT applications, as well as the soft constraints on the trustworthiness of validation. Not built on premise of Bayesian or game-theoretical assumptions, it is conducive to practical adoption, by virtue of augmenting rather than redesigning existing MCS systems, minimal user effort requirement, as well as resistance to common security attacks.

Performance evaluation based on a real IoT dataset has demonstrated that cross validation provides a unified solution to two disparate scenarios: reinforcement of obscure truth and discovery of hidden truth. In particular, hidden truth commonly remains unidentified under conventional statistical and data mining methods. Quantitative measurements via posterior belief enhancement and KL divergence indicate remarkable improvement in data quality as well.

ACKNOWLEDGMENTS

This work is supported by the General Research Fund CUHK 14219016 from Hong Kong UGC, the Presidential Fund from the Chinese University of Hong Kong, Shenzhen, and the National Science Foundation (NSF) grants CNS-1818942, CCF-1725755, CNS-1545050, and CCF-1533918.

REFERENCES

- [1] T. Luo, S. S. Kanhere, J. Huang, S. K. Das, and F. Wu, "Sustainable incentives for mobile crowdsensing: Auctions, lotteries, and trust and reputation systems," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 68–74, March 2017.
- [2] H. Jin, L. Su, D. Chen, H. Guo, K. Nahrstedt, and J. Xu, "Thanos: Incentive mechanism with quality awareness for mobile crowd sensing," *IEEE Transactions on Mobile Computing*, August 2018.
- [3] T. Luo, H.-P. Tan, and L. Xia, "Profit-maximizing incentive for participatory sensing," in *IEEE INFOCOM*, 2014, pp. 127–135.
- [4] G. Radanovic and B. Faltings, "A robust Bayesian truth serum for non-binary signals," in *AAAI*, 2013, pp. 833–839.
- [5] D. Prelec, "A Bayesian truth serum for subjective data," *Science*, vol. 306, no. 5695, pp. 462–466, October 2004.
- [6] T. Luo, S. S. Kanhere, H.-P. Tan, F. Wu, and H. Wu, "Crowdsourcing with Tullock contests: A new perspective," in *IEEE INFOCOM*, 2015, pp. 2515–2523.
- [7] E. Kamar and E. Horvitz, "Incentives and truthful reporting in consensus-centric crowdsourcing," MSR-TR-2012-16, Microsoft Research, Tech. Rep., 2012.
- [8] B. Kantarci, P. M. Glasser, and L. Foschini, "Crowdsensing with social network-aided collaborative trust scores," in *IEEE Globecom*, 2015.
- [9] C. Wu, T. Luo, F. Wu, and G. Chen, "EndorTrust: An endorsement-based reputation system for trustworthy and heterogeneous crowdsourcing," in *IEEE Globecom*, 2015.
- [10] L. Zeynalvand, J. Zhang, T. Luo, and S. Chen, "MASA: Multi-agent subjectivity alignment for trustworthy Internet of things," in *21st International Conference on Information Fusion (FUSION)*, 2018, pp. 2017–24.
- [11] K. L. Huang, S. S. Kanhere, and W. Hu, "Are you contributing trustworthy data?: the case for a reputation system in participatory sensing," in *ACM MSWiM*, 2010, pp. 14–22.
- [12] H. Amintoosi and S. S. Kanhere, "A trust framework for social participatory sensing systems," in *Mobiquitous*, 2012, pp. 237–249.
- [13] D. Wang, L. Kaplan, H. Le, and T. Abdelzaher, "On truth discovery in social sensing: A maximum likelihood estimation approach," in *ACM/IEEE IPSN*, 2012, pp. 233–244.
- [14] E. Davami and G. Sukthankar, "Improving the performance of mobile phone crowdsourcing applications," in *AAMAS*, 2015, pp. 145–153.
- [15] S. Gisdakis, T. Giannetsos, and P. Papadimitratos, "SHIELD: A data verification framework for participatory sensing systems," in *ACM Conference on Security & Privacy in Wireless and Mobile Networks (WiSec)*, 2015.
- [16] P. Michelucci and J. L. Dickinson, "The power of crowds," *Science*, vol. 351, no. 6268, pp. 32–33, 2015.
- [17] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge: MIT Press, 1998.
- [18] T. Luo, S. S. Kanhere, and H.-P. Tan, "SEW-ing a simple endorsement web to incentivize trustworthy participatory sensing," in *IEEE SECON*, 2014, pp. 636–644.
- [19] Y. Zhan, Y. Xia, Y. Liu, F. Li, and Y. Wang, "Incentive-aware time-sensitive data collection in mobile opportunistic crowdsensing," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 9, pp. 7849–7861, Sept 2017.
- [20] L. Duan, T. Kubo, K. Sugiyama, J. Huang, T. Hasegawa, and J. Walrand, "Motivating smartphone collaboration in data acquisition and distributed computing," *IEEE Transactions on Mobile Computing*, vol. 13, no. 10, pp. 2320–2333, Oct 2014.
- [21] C. Tham and T. Luo, "Quality of contributed service and market equilibrium for participatory sensing," *IEEE Transactions on Mobile Computing*, vol. 14, no. 4, pp. 829–842, April 2015.
- [22] H. Jin, L. Su, and K. Nahrstedt, "Theseus: Incentivizing truth discovery in mobile crowd sensing systems," in *ACM MobiHoc*, 2017.
- [23] C. List, "Social choice theory," in *The Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, Winter 2013.
- [24] J. Bartholdi, C. A. Tovey, and M. A. Trick, "Voting schemes for which it can be difficult to tell who won the election," *Social Choice and welfare*, vol. 6, no. 2, pp. 157–165, 1989.
- [25] T. Luo, S. K. Das, H. P. Tan, and L. Xia, "Incentive mechanism design for crowdsourcing: An all-pay auction approach," *ACM Trans. on Intelligent Systems and Technology*, vol. 7, no. 3, pp. 35:1–26, 2016.
- [26] M. Holmberg and T. Artursson, "Drift compensation, standards, and calibration methods," *Handbook of Machine Olfaction: Electronic Nose Technology*, pp. 325–346, 2002.
- [27] L. Kugler, "Why GPS spoofing is a threat to companies, countries," *Communications of the ACM*, vol. 60, no. 9, pp. 18–19, Sep 2017.
- [28] M. Harris, "Researcher hacks self-driving car sensors," <https://spectrum.ieee.org/cars-that-think/transportation/self-driving/researcher-hacks-selfdriving-car-sensors>, Sep 2015.
- [29] N. C. Moore, "Scientists can hack sensors in cars and phones with a \$5 speaker," <https://www.futurity.org/sensors-accelerometer-hacking-sound-1379602-2/>, Mar 2017.
- [30] Amazon Mechanical Turk, <http://www.mturk.com>.
- [31] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on stochastic programming: Modeling and theory*. SIAM and Mathematical Programming Society, 2009.
- [32] TIBCO Statistica, "Compare distribution tables," <http://www.statsoft.com/Textbook/Distribution-Tables>.

- [33] "Bessel's correction," https://en.wikipedia.org/wiki/Bessel's_correction.
- [34] V. Kuleshov and P. Doina, "Algorithms for the multi-armed bandit problem," *J. of Machine Learning Research*, vol. 1, pp. 1–48, 2000.
- [35] T. Luo, S. S. Kanhere, S. K. Das, and H.-P. Tan, "Incentive mechanism design for heterogeneous crowdsourcing using all-pay contests," *IEEE Transactions on Mobile Computing*, vol. 15, no. 9, pp. 2234–2246, Sept 2016.
- [36] —, "Optimal prizes for all-pay contests in heterogeneous crowdsourcing," in *IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*, 2014, pp. 136–144.
- [37] X. Zhang, Z. Yang, W. Sun, Y. Liu, S. Tang, K. Xing, and X. Mao, "Incentives for mobile crowd sensing: A survey," *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 54–67, 2016.
- [38] F. Restuccia, S. K. Das, and J. Payton, "Incentive mechanisms for participatory sensing: Survey and research challenges," *ACM Transactions on Sensor Networks (TOSN)*, vol. 12, no. 2, pp. 13:1–13:40, May 2016.
- [39] L. G. Jaimes, I. J. Vergara-Laurens, and A. Raj, "A survey of incentive techniques for mobile crowd sensing," *IEEE Internet of Things Journal*, vol. 2, no. 5, pp. 370–380, Oct 2015.
- [40] J.-C. Herrera, D. Work, J. Ban, R. Herring, Q. Jacobson, and A. Bayen, "Evaluation of traffic data obtained via GPS-enabled mobile phones: the Mobile Century experiment," *Transportation Research Part C*, vol. 18, no. 3, pp. 568–583, 2010.
- [41] D. Work, <https://my.vanderbilt.edu/danwork/open-data-2/>.
- [42] <http://traffic.berkeley.edu/project/downloads/mobilecenturydata>.
- [43] A. Hobson, *Concepts in Statistical Mechanics*. New York: Gordon and Breach, 1971.



Tie Luo is a Program Lead and Research Scientist with the Institute for Infocomm Research (I2R), A*STAR, Singapore. He received the PhD degree in electrical and computer engineering from the National University of Singapore. His research interests are Internet of Things analytics with machine learning, trust management for IoT, and security and privacy. He was a Best Paper Award nominee of IEEE INFOCOM 2015, Best Paper Award recipient of ICTC 2012, and Best Student Paper Award recipient of AAIM 2018.

He has delivered a technical tutorial at IEEE ICC 2016. His research on mobile crowdsourcing was featured in the IEEE Spectrum magazine. He is a Senior Member of the IEEE.



Jianwei Huang is a Presidential Chair Professor and the Associate Dean of the School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen. He received the Ph.D. degree from Northwestern University in 2005, and worked as a Postdoc Research Associate at Princeton University during 2005–2007. He has been a Distinguished Lecturer of IEEE Communications Society and a Clarivate Analytics Highly Cited Researcher in Computer Science. He is the co-author of 9 Best Paper Awards,

including IEEE Marconi Prize Paper Award in Wireless Communications in 2011. He received the CUHK Young Researcher Award in 2014 and IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2009. He has served as an Associate Editor of IEEE Transactions on Mobile Computing, IEEE/ACM Transactions on Networking, IEEE Transactions on Network Science and Engineering, IEEE Transactions on Wireless Communications, IEEE Journal on Selected Areas in Communications - Cognitive Radio Series, and IEEE Transactions on Cognitive Communications and Networking. He has served as the Chair of IEEE ComSoc Cognitive Network Technical Committee and Multimedia Communications Technical Committee. He is a Fellow of the IEEE.



Salil Kanhere received his M.S. and Ph.D. degrees both in Electrical Engineering from Drexel University, Philadelphia, USA. He is currently a Professor in the School of Computer Science and Engineering at UNSW Sydney, Australia. He is also a conjoint researcher at CSIRO Data61 and serves on the advisory board for two technology start-ups. His research interests include Internet of Things, pervasive computing, blockchain, crowdsourcing, data analytics, privacy and security. He has published over 200 peer-reviewed articles and delivered over 30 tutorials and keynote talks on these topics. His research has been featured on ABC News Australia, Forbes, IEEE Spectrum, Wired, ZDNET, Computer World, Medium, MIT Technology Review and other media outlets. He regularly serves on the organizing committee of a number of IEEE and ACM conferences, and is on the Editorial Board of Elsevier's Pervasive and Mobile Computing and Computer Communications. He is a recipient of the Alexander von Humboldt Research Fellowship, and is an ACM Distinguished Speaker. He is a Senior Member of both the IEEE and the ACM.



Jie Zhang is an Associate Professor of the School of Computer Science and Engineering, Nanyang Technological University, Singapore. He is also an Associate of the Singapore Institute of Manufacturing Technology (SIMTech), A*STAR. He obtained Ph.D. in Cheriton School of Computer Science from University of Waterloo, Canada, in 2009. During PhD study, he held the prestigious NSERC Alexander Graham Bell Canada Graduate Scholarship rewarded for top PhD students across Canada. He was also the

recipient of the Alumni Gold Medal at the 2009 Convocation Ceremony. The Gold Medal is awarded once a year to honour the top PhD graduate from the University of Waterloo. His papers have been published by top journals and conferences and won several best paper awards. Jie Zhang is also active in serving research communities.



Sajal K. Das is a Professor of Computer Science and the Daniel St. Clair Endowed Chair at the Missouri University of Science and Technology, where he was the Chair of Computer Science Department during 2013–2017. His research interests include cyber-physical security and trustworthiness, wireless sensor networks, mobile and pervasive computing, crowdsensing, cyber-physical systems and IoTs, smart environments (e.g., smart city, smart grid and smart health care), cloud computing, biological and social networks, and applied graph theory and game theory. He has published extensively in these areas with over 700 research articles in high quality journals and refereed conference proceedings. He holds 5 US patents and coauthored 4 books, the most recent of which is Principles of Cyber-Physical Systems: An Interdisciplinary Approach (Cambridge University Press, 2018). His h-index is 83 with more than 28,000 citations according to Google Scholar. He is a recipient of 10 Best Paper Awards at prestigious conferences like ACM MobiCom and IEEE PerCom, and numerous awards for teaching, mentoring and research including the IEEE Computer Society's Technical Achievement Award for pioneering contributions to sensor networks and mobile computing, and University of Missouri System President's Award for Sustained Career Excellence. He serves as the founding Editor-in-Chief of Elsevier's Pervasive and Mobile Computing Journal, and as Associate Editor of several journals including the IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Mobile Computing, and ACM Transactions on Sensor Networks. He is a Fellow of the IEEE.