Maintaining the Balance between Privacy and Data Integrity in Internet of Things

[*]Department of Computer and Information Sciences, Fordham University, New York, 10458 USA
[+]College of Computer Science, and Technology, Huaqiao University, Xiamen, Fujian 361021, China.

ABSTRACT

The recent proliferation of human-carried mobile and smartphone devices has opened up opportunities of using crowd-sensing to collect sensory data in Internet of Things (IoT). As tapping into the sensory data and resources of the smartphones becomes common place, it is necessary to ensure the *privacy of the device user* while maintaining *the accuracy and the integrity* of the data collected. IoT system devices often sacrifice either user privacy or data integrity. It has also become important to limit the *computational cost* and burden on the user devices, as increasingly more services desire to tap into the resource that these devices provide. In this paper we propose a balanced truth discovery (BTD) framework that attempts to meet all three of the aforementioned needs: user privacy, data integrity, and limited computational cost. The BTD framework also reduces user participation in the truth discovery process. The nature of the BTD framework is simple, providing the possibility for easy modification (e.g. cryptography and weight assignment). This reduces computation cost for the user device, but also limits the interactions between the user devices and the server, which is essential to data integrity. BTD framework also takes steps to blur the user device's original sensory data, by processing results in groups called zones. An enhanced method takes privacy preservation a step further, by protecting the user from an untrusted data-collecting party. Analysis of simulations running the BTD framework provides evidence for the preservation of data integrity.

CCS Concepts

C.4 [**Computer-Communication Networks**]: Performance of Systems; H.1.1 [**Systems and Information**]: Value of information; **K.4.1 [Computers and Society]:** Public Policy Issues—Privacy,

Keywords

Internet of Things; Crowd Sensing; Truth Discovery; Privacy; Data Integrity, Smartphones

1. INTRODUCTION

The recent proliferation of human-carried mobile devices (smartphones, smartwatches, smartglasses, etc.) has led to a dramatic increase in the growth of sensory-data resources for the Internet of things (IoT). These are integrated with on-board sensors (e.g., Wi-Fi and Bluetooth radios as sensors, GPS, accelerometer, compass, camera, etc.) that have given rise to crowd sensing in IoT [1], [2], [11]. An important issue with crowd sensing applications of IoT is that the sensory data provided by individual participants are usually not reliable or accurate [3]. An approach is to involve the probability of a user providing accurate data with user weight when aggregating sensory data and generate the aggregated results, which should be close to the information provided by reliable users. The main obstacle is that the user reliability is normally unknown a priori and can be extracted from sensor collected data. To deal with this obstacle, the truth discovery [2-5], [10] is used, i.e., to determine truthful facts from unreliable user information.

The goal of the truth discovery is to infer truthful facts from unreliable sensory data. This has revealed three overwhelming demands in the process in discovering truths: privacy for user, data integrity for data-collection party, and low computational costs as demanded by the natural growth of crowd sensing systems. Existing truth discovery and crowd sensing systems [1-6] have a range of results on the spectrum from private to accurate, as well as a range of computational cost for user devices. These approaches often sacrifice one of the three demands to satisfy another, or simply neglect a demand. A recent approach, privacy-preserving truth discovery (PPTD) [1] focusses on preserving the privacy of a user participating in a crowd sensing system. However, in doing so, the PPTD framework obscures the original sensory data of the user devices and requires a greater sample size before the estimated truth becomes accurate. This is caused specifically when the estimated ground truth is initialized randomly. This is a direct sacrifice of data accuracy for privacy.

Many existing frameworks also place a computational burden on user devices [2], [6]. PPTD, in particular, requires user devices to participate in the weighted calculation and partial decryption of the estimate ground truth. A computation which is by itself small, but occurs in great numbers, can be a burden on the user device. Furthermore, the more user intervention, the greater the chance for loss of data integrity, be it from faulty equipment or an active attack on integrity. The possibility of an active attack is a very real concern. Whether the attacks come from a third party or the device user themselves, it must be addressed to preserve data integrity.

In this paper, we propose a balanced truth discovery framework (BTD) that attempts to satisfy all the three of the demands in IoT. BTD can overcome the limitations with previous frameworks. By combining some of the privacy preserving functions through truth discovery, we propose an idea of zones and blurring data at the individual user level and privacy can be well preserved. We limit the participation of user devices in the computation of the estimated truth results in a reduction in both the cost and burden put onto user devices as well as the damage, caused by an active security attack on data integrity. The BTD framework treats user devices as if they were a database resource of sensory data. It connects to the device, demands a low-cost result, and exits, leaving the device for other services to use (the main idea is further described in Section 2).

The organization/contribution of the paper is as follows. We provide an in-depth discussion of our main idea in Section 2.

Section 3 describes the methods within the framework, including the initialization of the ground truth and zone processing, the extraction method, and the possible modifications as well as suggested approaches to cryptography and reliability weight. Section 4 describes the enhanced method. Finally, we conclude with Section 5 giving the analysis of the simulations running the BTD framework. We provide conclusion in Section 6.

2. MAIN IDEA

The balanced truth discovery (BTD) framework is being proposed regarding the serious limitations with the previous frameworks in terms of *user privacy*, *collected data integrity*, and *low computational costs*, as demanded by the natural growth of crowd sensing in IoT. By limiting the participation of user devices in the computation of the estimated truth, we reduction both the cost and burden placed onto the user devices, as well as the damage caused by an active attack on data integrity. BTD aims to satisfy all the demands of crowd sensing systems. The BTD framework treats user devices as if they were a database resource of sensory data. It connects to the device, demands a low-cost result and gets out, leaving the user device for other services to use.

The process in BTD framework begins by initializing the estimated ground truth with a randomvalue. The randomly-initialized ground truth is sent to *k* user devices; these *k* devices are called a zone. An IoT device receives the value and combines it with its own sensory data. The device then sends its result to the server. The device may encrypt the data using any desired cryptography. This ends the user device's participation in the crowd sensing system. The server then receives the *k* results from *k* devices and finds the average. Using this average, the server then extracts the original randomly-initialized ground truth using an extraction method. The estimated ground truth that the server is left with is an exact match to an average of the original sensory data of user devices.

The method of masking a user device's sensory data with a combined average, summing the results of a zone, then removing the so-called "mask" to be left with a true average will repeat with every zone being processed. The results from each zone is aggregated with the currently estimated ground truth. This method prevents individual data (may be personal data) from being sent to the server and solely representing the currently estimated ground truth. This prevents personal information from being acquired through alteration or eavesdropping.

In the event that the data-collecting party is untrusted, BTD framework provides a method with enhanced privacy. The enhanced method allows user devices to use a random weight when calculating the average of their own sensory data and the estimated ground truth sent by the server. This prevents the server from being able to extract the user device's sensory data with 100% accuracy by using an extraction method. The range of this weight can be provided to the user by the server, to best fit the context of the ground truth, or by the user device for maximum protection. It is suggested that the user device selects its own range based on its evaluation of the data-collection party (i.e. use zero weight variance with trusted sites, +/- *v*% with unknown/untrusted sites).

The BTD framework also leaves the ability to add modifications. Cryptography is left completely up to the user of this framework. The ability to weigh the reliability of user devices is also in the hands of the user of this framework. Some weighting algorithms and methods can be used for our purpose [1], [3], [6], [7].

3. BTD FRAMEWORK

In this section, we describe the BTD framework in details.

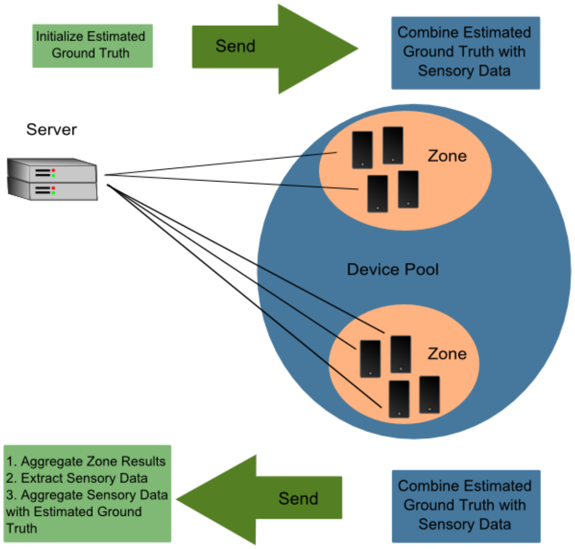
Initialize Estimated Ground Truth
Send
Combine Estimated Ground Truth with Sensory Data
Server
Zone
Device Pool
Zone
1. Aggregate Zone Results
2. Extract Sensory Data
3. Aggregate Sensory Data with Estimated Ground Truth
Send
Combine Estimated Ground Truth with Sensory Data

Figure 1: Balanced truth discovery (BTD) framework.

We illustrate BTD framework in Figure 1, which is comprised of the following functions:

1. SERVER: Randomly initialize estimated ground truth
2. SERVER: Send estimated ground truth to *k* devices (a zone)
3. DEVICE(S): Using equation (1), calculate the average of the estimated ground truth and device's sensory data
4. DEVICE(S): Send result to server
5. SERVER: Calculate the average of all *k* results from a zone
6. SERVER: Extract sensory data by using equation (2) to remove the data sent in function 2.
7. SERVER:
    a. If this is first zone, set the estimated ground truth equal to the result of function 6.
    b. If this is not first zone, aggregate the result of function 6 with the current estimated ground truth using equation (3).

3.1 Initialization and Zone Processing

In this subsection we provide a detailed explanation of functions 1-5. The BTD framework begins by initializing the estimated ground truth to a randomized value [1]. The initialization happens within the server by the data-collection party. The randomized value will have little to no effect on the estimated ground truth when using the BTD framework. However, it is still suggested to use a random value that makes contextual sense, especially when using the enhanced method described in Section 4. This initialized value is used so that no individual sensory data is sent to the server. Algorithms 1 and 2 are about the zone processing and user calculate in a certain zone, respectably.

The randomized ground truth is sent to *k* user devices, which the server wishes to utilize. These *k* devices represent a zone and will be processed together, so no individual sensory data is handled by the server. The number *k* represents the size of the zone. The value of *k* can be chosen by the data-collection party. If the aforementioned party utilizes only a small number of devices, it is suggested to use a small value for *k*. A small value for *k* ensures that the estimated ground truth is updated frequently. The value of *k* may also be determined within context. The way *k* devices are chosen for a particular zone is entirely up to the data-collection party to decide. It may make contextual sense to divide a crowd into n zones of size *k* based on location, device specifications or simply as they become available to the server. The degree of anonymity required varies from context to context.

Each of the *k* user devices calculate the summation of their own sensory and the estimated ground truth sent by the server. The summation is calculated as if there are only two parts, their own weighed at 50% and the servers, also weighed at 50%. The formula appears below:

$$x^k = sensoryData * 0.5 + x * 0.5 \qquad (1)$$

Each of the *k* user devices sends the data back to the server once their individual calculations are complete. The server aggregates the results. The BTD framework simulated for analysis purposes simply finds the mean of the results without weighted reliability.

The utilization of zone processing obscures data at an individual level, but the bigger picture (the one of *k* devices) remains crystal clear. As an example, the location and change of location of a group of people in New York City can be monitored. If someone attempts to learn the individual location and habits of a person, the data becomes indecipherable within the group. However, useful information can found that can answer the following questions: what sites do people go in New York City visit in the evening versus during the day, which boroughs have the most active nightlife, and what is the demographic or origin of people who visit this particular area at this particular time?

Algorithm 1: Zone Processing

```
        Input: k user devices: device[k]
        Output: Estimated ground truth: x

1 Randomly initialize the ground truth for each object;
2 repeat
3           for each user device k do
4           Send ground truth to user device;
5           Aggregate device results;
6     end
7           if this is first zone do
8           Extract randomized value;
9           x = remainder;
10    else do
11          Extract value;
12          Aggregate remainder and x;
13  until all zones have been processed;
14  return x;
```

Algorithm 2: User Calculation

```
        Input: Estimated ground truth: x
        Output: Result

1           Result = aggregate sensory data and x    (e.g., Eqn. (1));
2  return Result;
```

3.2 Extraction Method

In this section we discuss functions 6-7 of the BTD framework. Randomly initializing the ground truth (within a range based on context), presents a data accuracy issue if not properly addressed. The BTD framework addresses the issue using a method which extracts the estimated ground truth sent to *k* users within a zone from their aggregated sensory data. As a reminder, the user calculation aggregates the user device's sensory data with the estimated ground truth provided by the server as two equal parts. Therefore, the aggregated sensory data of the *k* user devices in a zone is one half user device sensory data and one half estimated ground truth. The following equation extracts the estimated ground truth and leaves the server with the aggregated sensory data:

$$s^z = x^z * 0.5 + [x^z - 2(x - x^z)] * 0.5 \qquad (2)$$

Where s^z represents the aggregated sensory data of zone *z*, x^z represents the aggregated results supplied by user devices (i.e. s^z pre-extraction), and *x* is the currently estimated ground truth. This extraction method is not only used to extract the randomized value that the estimated ground truth is initialized to, but also in conjunction with the following equation to update the ground truth:

$$x = s^z * k/c + x * (c-k)/c \qquad (3)$$

where *k* is the number of user devices in a zone and *c* is the number of user devices that have participated in the crowd sensing system including the *k* user devices of the zone currently being processed. The extraction method, used in conjunction with the randomly initialized value and the idea of cloaking the sensory data of a user device, allows for the preservation of privacy without a sacrifice in data integrity. The results of this method are theoretically exact copies of the aggregation of the sensory data of all *c* devices.

3.3 Possible Modifications for Security

We describe the security features of BTD framework in this subsection, including cryptography.

3.3.1 *Cryptography*

The BTD framework does not implement a specific cryptography method. However, it is still suggested to use a cryptosystem with the BTD framework to correctly preserve privacy of all parties involved. A potential threat exists if a third party intercepts both the data sent from the server to the user device and the data sent from the user device to the server. If the attacker who intercepted the data understood the nature of the BTD framework, they could use the extraction method to obtain the individuals sensory data (the enhanced method proposed in Section 4 will protect against this). It is also important to note that the estimated ground truth for which the data-collection party is aggregating data, may not be information the party wishes to disclose to a third-party

In order to meet the demand for low-cost computation on the user device side, it is suggested that low-cost cryptosystem be used. Examples of possible cryptosystems include: AES, DES, RSA, Blowfish, etc. The cryptosystem used in PPTD [1] is complex and requires user device participation at a great degree; cryptosystems like these should be avoided if possible.

3.3.2 *Weighted Reliability*

The BTD framework includes a truth discovery method to weigh the reliability of user devices and to determine the truth with the devices, i.e., to check whether the data is altered or not. Each device has an equal say (decision) in the aggregation of the ground truth. Weighted reliability theoretically protects the integrity of the data. Conventional weighting methods can be found [1], [3], [7], 8]. For example, [1] offers a weighted reliability measurement method by which each device weight information needs to be sent to the user device, whereas BTD framework requires such a method be calculated and used purely on the server side. In BTD framework, we compute sensor status value to check whether or not personal is altered at the transmission. The basic idea is that a device's status value can be given a high value if the device transmitted data is close to estimated ground truths. Typically, the sensor status values are computed as follows:

$$S_k = \log(\frac{\sum_{k'=1}^{K}\sum_{m=1}^{M} d(x_m^{k'}, x_m^*)}{\sum_{m=1}^{M} d(x_m^k, x_m^*)}) \qquad (2)$$

d(.) is the distance function which measures the difference between sensors observation values $x_m^{k'}$ and the estimated ground truths x_m^*[4]. *d*(.) relies on particular sensing application scenarios. The proposed framework is intended to deal with crowed sensing applications, where the sensory data is continuous, we adopt the following normalized squared distance function:

$$d(x_m^k, x_m^*) = \frac{(x_m^k - x_m^*)^2}{std_m} \qquad (3)$$

These equations (2-3), when used together, base the weight of a user device on the standard deviation and the normalized squared distance function. It is also important to note that the reliability of a device may differ from context to context [5]. For best possible data integrity, it is suggested that weighted reliability be calculated by the server for each user device for each context (i.e. if the server is finding two or more ground truths using different sensors).

4. ENHANCED METHOD

When handling the privacy of device users, it is important to address all possible threats to privacy. This includes the possibility of untrustworthy data-collection parties. If a server claims to be running the BTD framework, but is actually phishing for personal data, the individual sensory data of a user device may be obtained by using the extraction method; this is assuming the server fails to aggregate the data among *k* user devices in a zone. To correctly preserve privacy, we must protect privacy against threats from all parties not just outside threats and eavesdropping.

The enhanced method addresses this issue by blurring the individual sensory data once more [2]. The method allows user devices to use a randomized weight (within a range in context) when combining its own sensory data to the estimated ground truth provided by the server. This prevents the server from extracting the exact individual sensory data. The extraction method is only exact when the result being sent from the user device to the server is 50% device sensory data and 50% estimated ground truth. This does have a negative effect on data integrity. However, as data is processed within a zone of *k* user devices all with their own small variance on weight the aggregated result is very accurate. The data integrity become imperfect, but within certain contexts the effect is minuscule. The enhance method is given in Algorithm 3.

When using the enhanced method, there are multiple ways to assign a proper weight variance. The variance on weight may be supplied by the server, in order to satisfy the requirements of the context, or may be held within the device itself. It is suggested to take a combined approach. Let the server define the context, but if the server declares a variance that is too small, risking the privacy of the device user, the user device can use a value declared within its own range of safe variance options.

5. PERFORMANCE EVALUATION

In this section, we evaluate the proposed BTD framework for maintaining balance between privacy and data integrity framework.

Algorithm 3: Enhanced Method (User calculation)

Input: Estimated ground truth: *x*, Variance:
Output: Result

```
if v is not a sufficient value do
        Replace v with desirable value;
end
weight = random between 50-v and 50+5;
Result = (sensory data * weight/100) + (x * 100-
        weight/100)
return Result;
```

5.1 Simulation Settings

We have implemented BTD framework in an extension of the OMNeT++ (Network Simulation Framework) that supports dual-radios for each node [8]. Although sensor Bluetooth is not yet supported by OMNeT, we have emulated Bluetooth by using the Zigbee IEEE 802.15 WPAN protocol, with the communication range adjusted appropriately to 10m. Each of the simulations has run for 500 virtual minutes and has been repeated 12 times. Moreover, since our focus is on the mobile devices and their sensors (i.e., mobile crowd), each of the simulated sensor nodes has been programmed to move randomly 4m/s every minute. The target environment has been set to be 50m *x* 50m.

Experiment results on real-world data traces collected from sensing systems [1]. For the purpose of experimentation and testing, the simulation environment running the BTD framework was created using C++. The simulation makes use of three classes: 1) the simulation class; 2) server; and 3) device. The server class defines a server object, which has the necessary methods to mimic the behavior of a server running BTD as described throughout this paper. The device class, in a similar manner, mimics the behavior expected of a user device. The simulation class provides the server with the necessary devices to calculate a ground truth. The simulation program, as a whole, is equipped to run a crowd sensing simulation using BTD framework both with and without the enhanced method. We provide an example set of simulations, with simulation set: device pool: 500 devices, zone Size: Variable, Zone Count: 500 devices / Zone Size, Variance: +/- 0.5%. The purpose of Simulation 2 is to calculate the effect that zone size has on data accuracy when using the enhanced method. Zone sizes 2-50 are used in the simulation and the device pool is a fixed size.

As shown in Figure 2, the results of the simulation demonstrate that an increase in zone size causes an increase in difference (decrease in accuracy) while maintaining a fixed device count. The magnitude of the effects is small (around +/- 0.01% accuracy with

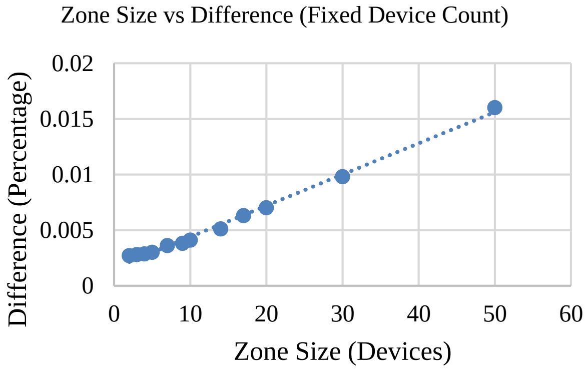

Figure 2. Zone Size vs. Difference with fixed device pool

a zone size of 30), however certain contexts may require greater accuracy or low device pools.

5.3 Result Comparisons

As a baseline approach for status value truth discovery, we use the state-of-the-art truth discovery scheme in our simulations, i.e., CRH (conflict resolution on heterogeneous data) [8], which does not take any actions to break sensor security during the whole procedure. A (p, ⌊p\2⌋)-threshold Paillier cryptosystem is used in simulations (http://cs.utdallas.edu/dspl/cgi-bin/pailliertoolbox/). The status value truth discovery is implemented by following the Paillier Threshold Encryption Toolbox [1].

We also consider Voting framework for comparison [9]. Voting is used to eliminate conflicts for decision-making based on collected data, which is used to conduct majority voting so that information with the highest number of occurrences, mean, or median is regarded as the correct answer. In Voting, it is assumed that all the sensors are equally reliable, and thus the votes from different sensors are uniformly weighted. We adopted a state-of-the-art network based voting algorithm from [9].

The experiments used to test PPTD [1] and the simulations of BTD, which were previously discussed, are not the same. The results of the two cannot be compared directly. However, inferences can have made from the results of the PPTD experiments, the BTD simulations and also the analysis of CRH [6], a crowd sensing framework used within PPTD. Figure 3 shows the comparison results between BTD, PPTD, Voting, and other works.

Figure 3 shows a minimum error rate of 0.70-0.71. Regardless of the rounding parameter L [1], when the mean of absolute error (measured by the mean of absolute distance between the estimated results and ground truths) used in PPTD is combined with CRH (blue line) the error is 0.70-0.71. Figure 3 shows, as long as there is at least one reliable source, that BTD (black line) has an error rate of 0. This shows that PPTD, without any of the privacy preserving additives produces an error of approximately 0.71. Looking back at Section 5.2, our simulations show that BTD has an error of 0.05 during its “worst case”, a high weight variance +/- 5%, scenario while using the enhanced method. However, using a lower weight variance can yield a much lower error of around 0.001.

As stated previously, we cannot directly compare the results of the BTD simulations and the PPTD experiments. The significance of the errors produced by PPTD still gives evidence to the improvement in data accuracy that BTD provides. It is important to note that BTD also takes steps to ensure data integrity by protecting against users with intent to alter the estimated ground truth.

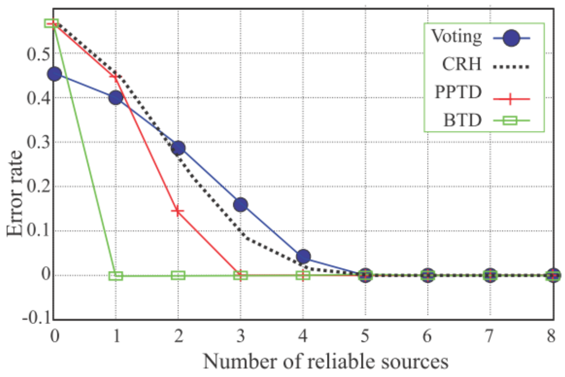

Figure 3. Comparisons: Error Rate vs. reliable sources.

6. CONCLUSIONS

In this paper, we proposed balanced truth discovery (BTD) framework to maintain the balance between data primary and integrity in IoT. BTD improves upon previous frameworks by satisfying three demands of mobile crowd sensing in IoT: preservation of privacy for the device user, data integrity for the data-collection party, and low-cost computation on the user device end. The possibility of modification allows for the framework to be improved and molded to serve a particular context. Simulations running the BTD framework show that accurate data within a few thousands of a percentage can be achieved whilst preserving the privacy of the device user.

7. ACKNOWLEDGMENTS

This work is supported in part by the Fordham University faculty startup research grant and Ames Fund, in part by the Promotion Program for Young and Middle-aged Teacher in Science and Technology Research of Huaqiao University (No.ZQN-PY308), and in part by the National Natural Science Foundation of China under Grant Numbers 61402543.

8. REFERENCES

[1] C. Miao, W. Jiang, L. Su, Y. Li, S. Guo, Z. Qin, H. Xiao, J. Gao, and K. Ren. Cloud-Enabled Privacy-Preserving Truth Discovery in Crowd Sensing Systems. in Proc. of ACM SenSys, 2015.

[2] R. Kravets, H. Alkaff, A. Campbell, K. Karahalios, and K. Nahrstedt. CrowdWatch: Enabling In-Network Crowd-sourcing. In Proc. of MCC, 2013

[3] M. Bhuiyan and J. Wu. Trustworthy and Protected Data Collection for Event Detection Using Networked Sensing Systems. IEEE Sarnoff, 2016.

[4] Q. Li, Y. Li, J. Gao, B. Zhao, W. Fan, and J. Han. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. in Proc. of ACM SIGMOD, 2014.

[5] F. Ma, Y. Li, Q. Li, M. Qiu, J. Gao, S. Zhi, L. Su, B. Zhao, H. Ji, and J. Han. Faitcrowd: Fine grained truth discovery for crowdsourced data aggregation. In Proc. of ACM SIGKDD, 2015.

[6] 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*, 8(4):425–436, 2014.

[7] S. Wang, D. Wang, L. Su, L. Kaplan, and T. F. Abdelzaher. Towards cyber-physical systems in social spaces: The data reliability challenge. In Proc. of IEEE RTSS 2014.

[8] O. Helgason and S. Kouyoumdjieva, “Enabling multiple controllable radios in omnet++ nodes,” in Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques, 2011, pp. 398–401.

[9] H.-T. Pai and Y. S. Han, “Power-Efficient Direct-Voting Assurance for Data Fusion in Wireless Sensor Networks,” *IEEE Transactions on Computers*, vol. 57, no. 2, pp. 261–273, 2008.

[10] M. Z. A. Bhuiyan, G. Wang, and K. R. Choo, "Secured Data Collection for a Cloud-Enabled Structural Health Monitoring System," In Proc. of IEEE HPCC 2016.

[11] M. Z. A. Bhuiyan and J. Wu, "Event Detection through Differential Pattern Mining in Internet of Things," In Proc. of IEEE MASS 2016.