

Finding Reliable Source for Event Detection Using Evolutionary Method

Raushan Ara Dilruba^{1,2} and Mahmuda Naznin^{1(✉)}

¹ Department of CSE, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh

mahmudanaznin@cse.buet.ac.bd

² Department of CS, University of Calgary, Calgary, Canada
rdilruba@ucalgary.ca

Abstract. Participatory sensing is a phenomenon where participants use mobile phones or social media and feed data to detect an event. Since, data gathering is open to many participants, one of the major challenges of this type of networks is to identify truthfulness of the reported observations. Finding the reliable sources is a challenging task since the node or participant's reliability is unknown or even the probability of the reported event to be true is also unknown. In our paper, we study this challenge and observe that applying evolutionary method, we can identify reliable source nodes. We call our approach Population Based Reliability Estimation. We validate our claim by experimental results. We also compare our method with another widely used method. From experiments we find that our approach is more efficient.

Keywords: Evoloutinray approach · Reliability · Participatory sensing · Genetic Algorithm

1 Introduction

Participatory Sensing is a process of data collection and interpretation of an event by feeding interactive data via web or social media [28,31]. A *Participatory Sensor Network* consists of nodes or participants to collect data for a common project goal within its framework [1,3]. The nodes or participants use their personal mobile phones to sense various activities of their surrounding environment and submit sensed data through mobile network or social networking sites [3,24,25].

However, finding reliable sources in a participatory sensor network is very challenging task due to the big and continuous volume of sensing and communication data generated by the participant nodes and the availability of ubiquitous, real-time data sharing opportunities among nodes [2,12,16,17,20]. One conventional way to collect the reliable data is conducting self-reported surveys. However, conducting survey is a time consuming procedure. Popular data collection can be achieved by using mobile devices like smart phones, wearable sensing

devices or through social networks [4, 10, 23, 26]. In a participatory network, the users are considered as participatory sensors, and an event can be reported or detected by the users [21, 22]. The major challenge in this participatory sensing is to ascertain the truthfulness of the data and the sources. The reliability of the sources is questionable because the data collection is open to a very large population [36]. The *reliability* of the participants (or sources) denotes the probability that the participant reports correct observations. Reliability may be impaired because of the lack of human attention to the task, or because of the bad intention to deceive. Without knowing the reliability of sources, it is difficult to measure whether the reported observations or events are true or not [32]. Openness in data collection also leads to numerous questions about the quality, credibility, integrity, and trustworthiness of the collected information [5, 9, 14, 15]. It is very challenging to find whether the end user is correct, truthful and trustworthy. If the nodes are reliable, the credibility of the total system increases. Therefore, it is very important to find the reliable sensing sources to detect the events.

In this paper, we address the challenges of finding the node reliability in a participatory sensing system. Our paper is organized as follows. Section 2 provides the background study. Section 3 provides the problem domain in details. Section 4 provides the experimental results. Finally, Sect. 5 gives the conclusion of this research work.

2 Related Work

In this section, we discuss some research work on the reliability estimation of the nodes in a participatory sensing network. For the case of specific kinds of data such as location data, a variety of methods are used in order to verify the truthfulness of the location of a mobile device [19]. The key idea is that time-stamped location certificates signed by wireless infrastructure are issued to co-located mobile devices. A user can collect certificates and later provide those to a remote party as a verifiable proof of his or her location at a specific time. However, the major drawback of this approach is that the applicability of these infrastructure based approaches for mobile sensing is limited as cooperating infrastructure may not be present in remote or hostile environments.

In the context of participatory sensing, where raw sensor data is collected and transmitted, a basic approach for ensuring the integrity of the content has been proposed in [14], which guards whether the data produced by a sensor has been maliciously altered by the users. Trusted Platform Module (TPM) hardware [14], can be leveraged to provide this assurance. However, this method is expensive and not practical since each user must have predefined hardware framework.

The problem of trustworthiness has been studied for resolving multiple, conflicting information on the web in [36]. The earliest work in this regard are proposed in [7, 8]. A number of recent methods in [3, 18, 32, 33] also address this issue, in which a consistency model is constructed in order to measure the trust in user responses in a participatory sensing environment. The key idea of this system is that untrustworthy responses from users are more likely to be different from one another, whereas truthful methods are more likely to be consistent

with one another. This broad principle is used in order to model the likelihood of participant reliability in social sensing with the use of a Bayesian approach [32]. A system called Apollo [18] has been proposed in this context in order to find the truth from noisy social data streams. However, these types of methods are a bit time consuming and do not ensure source reliability. In case of collaborative attack this method may fail.

In [18], authors present a fuzzy approach where this system is able to quantify uncertain and imprecise information, such as trust, which is normally expressed by linguistic terms rather than numerical values. However, the linguistic term can create vague results. In [28,34], authors present a streaming approach to solve the truth estimation problem in crowd sourcing applications. They consider a category of crowd sourcing applications, the truth estimation problem. This is basically reliability finding problem. In fact fact-finding algorithms are used to solve this problem by iteratively assessing the credibility of sources and their claims in the absence of reputation scores. However, such methods operate on the entire dataset of reported observations in a batch fashion, which makes them less suited to applications where new observations arrive continuously. The problem is modelled as an Expectation Maximization (EM) problem to determine the odds of correctness of different observations [3]. Problem and accessing on-line information from various data sources are mentioned in [37]. However, all of these methods suffer from the collecting the ground truth data for unavoidable circumstances. Therefore, finding credible nodes of the detected event is still very challenging research problem.

3 Problem Domain

In this section, we define the system model, problem formulation and give the details of our methodology. At first we discuss some preliminaries relevant to our research problem. Let us consider a participatory sensing model where a group of M participants, $S_1 \dots S_M$, make individual observations about a set of N events $C_1 \dots C_N$. The probability that participant S_i reports a true event when the event is actually true is a_i and the probability that participant S_i reports a true event when the event is actually false is b_i . θ is the set of a_i and b_i , $\theta(a_i, b_i)$.

To handle this challenge, we apply Genetic Algorithm or Population Based Method where we can keep around a sample of candidate solutions rather than a single candidate solution to find the solution quickly [11]. We know, GA generate solutions to optimize the problems inspired by natural evolution, such as *inheritance, mutation, selection, and crossover* [11]. We call our method *Population Based Reliability Estimation (PBRE)* which uses a set of reliability for the population instead of single reliability. In our approach, we call this set of reliability as P and we use Genetic Algorithm to estimate the best possible reliable participants. We call the set of θ as P which is a set of reliability. z_j is the probability that the event or claim C_j is indeed authentic.

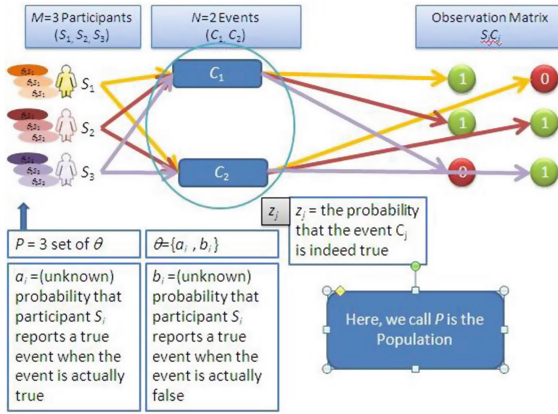


Fig. 1. Population based system model.

3.1 Population Based Method

In this section, we provide an outline of this method as follows.

Step 1: We initialize and build population in the following ways:

1. We initialize M, N
2. We take input SC matrix or *Source-Claim* matrix. Each entry of the matrix is either 0 or 1. Here, when the participant S_i reports an event C_j as false $S_i C_j = 0$, and when S_i reports an event C_j as true $S_i C_j = 1$. We assume, each observation and source of the matrix is independent of each other.
3. We initialize d , overall bias on event to be true (value may range from 0 to 1).
4. Finally, $P =$ The set of θ any value between 0 to 1.

Step 2: We calculate z_j as follows.

$p(z_j | X_j, \theta)$ is the conditional probability to be true, given the SC matrix X_j related to the j^{th} and the current estimate of θ .

Step 3: We compute fitness. *Computing Fitness* is described as follows. Then, we assess fitness of P , the set of reliability. We compare P with the best reliability. The *target reliability* or *target $_a_i$* is computed as follows.

$$target_a_i = \sum_{i=1}^M \left(\sum_{j=1}^N = \frac{S_i \times C_j}{N} \right)$$

For example, in an ideal case, when the probability of all events to be true, $z_j = 1$. Let us consider, there are 2 events and 3 participants which is illustrated in Fig. 2. Participant S_1 reports event C_1 as true, and reports the event S_2 as false. Therefore, $target_a_1 = \frac{S_1 C_1 + S_1 C_2}{2} = \frac{1+0}{2} = 0.5$.

Now, the objective is to select the best fit or the fittest a_i from P that helps to converge z_j . We take the fittest value from the initial set of values of a_i using the fitness function. We call this fittest value as *fit reliability* or *fit $_a_i$* .

Now, we define two types of fitness functions *Fit_Parent* and *Replace_Parent*.

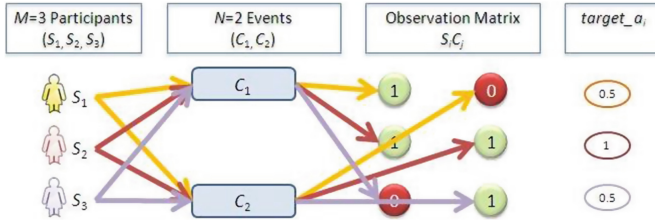


Fig. 2. Calculating the most reliable $target_{a_i}$

Type 1 : *Fit_Parent- Fit_Parent* selects fit_{a_i} from the set of a_i of S_i . Here, fit_{a_i} is the closest value to $target_{a_i}$. We describe the computation in Fig. 3.

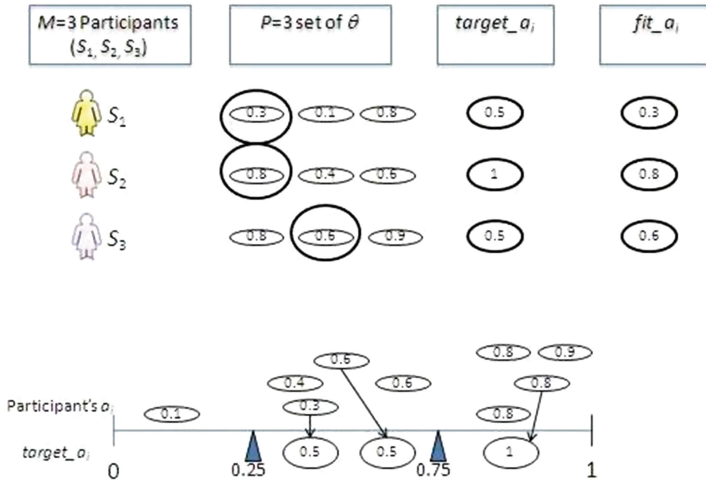


Fig. 3. Fit_Parent computation

Fit_Parent selects fit_{a_i} from the set of a_i of S_i . Here, fit_{a_i} is the closest value to $target_{a_i}$. We describe the computation in Fig. 3. For example, we initialize three sets of a_1 for participant S_1 e.i. 0.3, 0.1 and 0.8. Figure 3 is an illustrative example of Fit_Parent computation. We see that the $target_{a_1}$ is 0.5. Therefore, the closest a_1 e.i. fit_{a_1} is 0.3. Similarly, we calculate fitness for participant S_2 and S_3 which are $a_2 = 0.8$ and $a_3 = 0.6$ respectively.

Type 2 : *Replace_Parent-* Here, instead of selecting one fit_{a_i} from every participant S_i 's P , we select the full set of a_i which is the closest to set of $target_{a_i}$. Now, we give an illustrative example of Replace_Parent in Fig. 4.

For example, we initialize three sets of a_i for each participant S_1 e.i. $(a_{11}, a_{12}, a_{13}) = (0.3, 0.1, 0.8)$, for S_2 it is $(a_{21}, a_{22}, a_{23}) = (0.8, 0.4, 0.5)$ and for S_3 it is $(a_{31}, a_{32}, a_{33}) = (0.8, 0.5, 0.9)$. Now, we make another set taking the first a_i

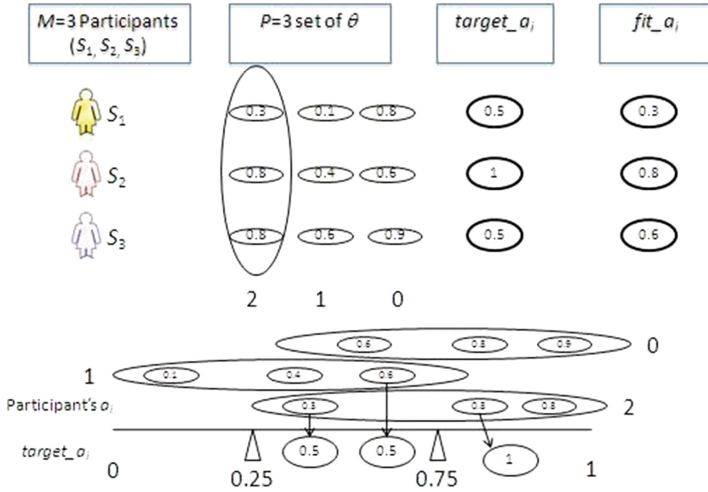


Fig. 4. Replace_Parent computation

from each S_i e.i. $(a_{11}, a_{21}, a_{31}) = (0.3, 0.8, 0.8)$ and similarly $(a_{12}, a_{22}, a_{32}) = (0.1, 0.4, 0.6)$ and $(a_{13}, a_{23}, a_{33}) = (0.8, 0.6, 0.9)$. Our $target_a_i = (0.5, 1, 0.5)$. Therefore, we find that there are two fit_a_i s in the first set, similarly one and no fit_a_i for the second and the third set. Finally, we take the first set as the set of fit_a_i .

Step 4: Breeding

Now, the objective is to generate a new $child_\theta$ from $parent_\theta$. We choose recombination technique [11] as breeding technique. This new values are called two children $anew_i$ and $bnew_i$, where,

$$anew_i = \alpha a_i + (1 - \alpha) b_i,$$

$$bnew_i = \beta b_i + (1 - \beta) a_i,$$

where, $\alpha =$ random value between 0 to 1, and

$\beta =$ random value between 0 to 1.

Step 5: Joining

We form the next generation parent by using new children. Joining equations are given as follows.

$$a_i = anew_i$$

$$b_i = bnew_i$$

Step 6: Error Percentage of Participant Reliability

We calculate the percentage of error of participant's reliability by dividing the total number of converged reliable nodes by the total number of reliable nodes.

The flow chart in Fig. 5 shows the summary of the procedure.

Now, we provide the formal algorithms of PBRE from Algorithms 1 to 7.

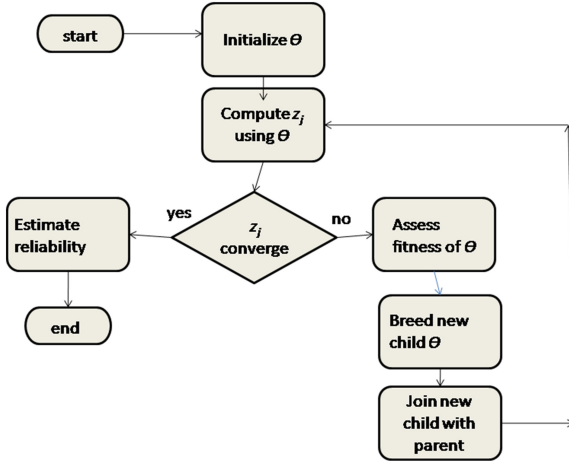


Fig. 5. The summary of the procedure.

Algorithm 1. Procedure PBRE

- 1: Take input: M, N, P, d , the observation matrix SC with random values either 0 or 1
 - 2: Initialize $\theta =$ random values between 0 and 1
 - 3: Initialize $fit_a(i)$ as NULL
 - 4: Initialize $zcount=0, z_j$ convergence metric
 - 5: Calculate $target_a(i) = \sum_{i=1}^M \sum_{j=1}^N \frac{SC(i,j)}{N}$
 - 6: **while** z_j does not converge **do**
 - 7: **for** $i=1:M$ **do**
 - 8: $a(i, P + 1) = fit_a(i)$ and add $fit_a(i)$ in the population of $a(i)$
 - 9: **end for**
 - 10: **for** $j=1:N$ **do**
 - 11: **for** $K=1:P+1$ **do**
 - 12: $z(j, K)$
 - 13: **if** $z(j, K) = d$ **then**
 - 14: $z_j =$ convergence counter
 - 15: **end if**
 - 16: **end for**
 - 17: **end for**
 - 18: Compute fitness using $Fit_Parent()$ or $Replace_Parent()$
 - 19: Apply $Breed()$, breeding
 - 20: Apply $Join()$, Joining
 - 21: Find reliability using $ReliabilityEstimation()$
 - 22: **end while**
-

Algorithm 2. Computing probability that the event z_j is true or false: procedure $z(j, K)$

- 1: Calculate a_t =conditional probability that participant's observation is true given θ and event $z=1$.
 - 2: $a_t(j, K) = \sum_{i=1}^M a(i, K)^{SC(i,j)} (1 - a(i, K))^{(1-SC(i,j))}$
 - 3: Calculate b_t =conditional probability that participant's observation is true given θ and event $z=0$.
 - 4: $b_t(j, K) = \sum_{i=1}^M b(i, K)^{SC(i,j)} (1 - b(i, K))^{(1-SC(i,j))}$
 - 5: $z(j, K) = \frac{a_t 1(j, K) \times d}{a_t 1(j, K) \times d + b_t 1(j, K) \times (1-d)}$
-

Algorithm 3. Procedure Fit_Parent()

- 1: This is to select closest a to $target_a$ as fit_a
 - 2: **for** $i = 1$ to M **do**
 - 3: **for** $K = 1$ to $P + 1$ **do**
 - 4: **if** ($0 \leq target_a(i) \leq 0.25$ AND $0 \leq a(i, K) \leq 0.25$) OR
 ($0.25 < target_a(i) \leq 0.75$ AND $0.25 < a(i, K) \leq 0.75$) OR
 ($0.75 < target_a(i) \leq 1$ AND $0.25 < a(i, K) \leq 1$) **then**
 - 5: $fit_a(i) = a(i)$
 - 6: **end if**
 - 7: **end for**
 - 8: **end for**
 - 9: **return** fit_a
-

Algorithm 4. Procedure Replace_Parent()

- 1: This is to select closest set of a to set of $target_a$ as fit_a
 - 2: **for** $i = 1$ to M **do**
 - 3: **for** $K = 1$ to $P + 1$ **do**
 - 4: **if** ($0 \leq target_a(i) \leq 0.25$ AND $0 \leq a(i, K) \leq 0.25$) OR
 ($0.25 < target_a(i) \leq 0.75$ AND $0.25 < a(i, K) \leq 0.75$) OR
 ($0.75 < target_a(i) \leq 1$ AND $0.25 < a(i, K) \leq 1$) **then**
 - 5: $count(K) ++$
 - 6: **end if**
 - 7: **end for**
 - 8: **end for**
 - 9: $best = 0$
 - 10: **for** $K = 1$ to $P + 1$ **do**
 - 11: **if** $count(K) > best$ **then**
 - 12: $best = count(K)$
 - 13: $L = K$
 - 14: **end if**
 - 15: **end for**
 - 16: **for** $i = 1$ to M **do**
 - 17: $fit_a(i) = a(i, L)$
 - 18: **end for**
 - 19: **return** fit_a
-

Algorithm 5. Procedure `breed()`

```

1: for  $i=1$  to  $M$  do
2:   for  $i=1$  to  $P$  do
3:      $t(i, K) = \alpha \times a(i, K) + (1 - \alpha) \times b(i, K)$ ,  $\alpha =$  between 0 to 1
4:      $s(i, K) = \beta \times b(i, K) + (1 - \beta) \times a(i, K)$ ,  $\beta =$  between 0 to 1
5:   end for
6: end for
7: return  $t, s$ 

```

Algorithm 6. Procedure `join()`

```

1: Replace new children with parents
2: for  $i = 1$  to  $M$  do
3:   for  $K=1$  to  $P$  do
4:      $a(i, K) = t(i, K)$ 
5:      $b(i, K) = s(i, K)$ 
6:   end for
7: end for
8: return  $a, b$ 

```

Algorithm 7. Procedure `ReliabilityEstimation()`

```

1: for  $i=1$  to  $M$  do
2:   if  $(0 < \text{target\_a}(i) < 0.25 \text{ AND } 0 < \text{fit\_a}(i, K) < 0.25) \text{ OR } (0.25 < \text{target\_a}(i) < 0.75 \text{ AND } 0.25 < \text{fit\_a}(i, K) < 0.75) \text{ OR } (0.75 < \text{target\_a}(i) < 1 \text{ AND } 0.25 < \text{fit\_a}(i, K) < 1)$  then
3:      $\text{truecount}++$ , count of correct reliability estimation
4:   end if
5: end for
6:  $\text{error} = (1 - \frac{\text{truecount}}{M}) \times 100$ , percentage of error reliability estimation

```

4 Experimental Results

In this section, we present the experimental results to show the effectiveness of *PBRE* method. We also compare our findings with the findings of another relevant algorithm Expectation Maximization [35]. The simulation of PBRE runs on 1.58 GHz *Intel Core 2 Duo Processor* with 2 GB memory. Simulation is done on synthetic data sets where SC matrix is generated randomly. SC matrix contains data of 1, 0 and if we used real time data set it would carry the similar property. The performance metrics used to evaluate the methods are described as follows:

1. The *Error Percentage of Participant's Reliability* denotes the estimation of reliability of a participant to a converged event z .
2. The *Convergence Rate* denotes how quickly participant can report the correct event. It is computed by the participant's reliability divided by the total iteration needed to converge.

Now, we give the table of simulation parameters used for testing in Table 1.

Table 1. Simulation parameters

Parameters	Value
Participant number, M	30–900
Event number, N	2–10
Observation Matrix, SC	0,1
Probability of a reported TRUE event as TRUE, a	(0 to 1)
Probability of reported FALSE event as TRUE, b	(0 to 1)
Probability of the event C is actually TRUE, d	0.7
α	(0 to 1)
β	(0 to 1)

We carry out experiments using simulation to evaluate the performance of the proposed *PBRE* scheme in terms of estimation accuracy of the probability that a participant is right or a measured variable is true compared to another existing reference method Expectation Maximization (EM) Method. We take the average of ten simulation runs. Variance we found is negligible. We consider two types of scenario. In a dense network, the number of participant nodes M is high.

Varying different parameters value, the error percentage was calculated. We now give the details as follows.

4.1 For Variable Number of Participants

We compare the estimation accuracy of PBRE (Fit_Parent and Replace_Parent) and Expectation Maximization(EM) scheme by varying the number of participants in the system.

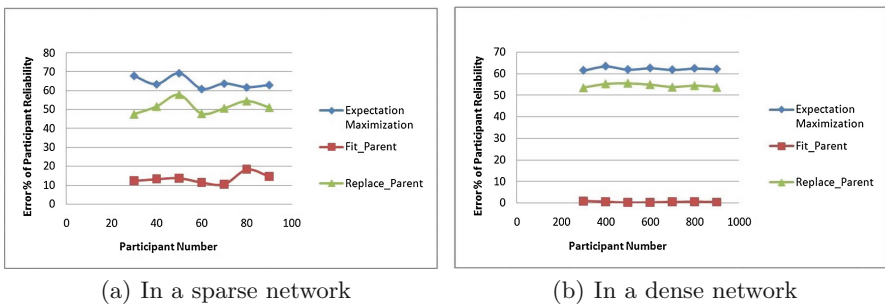


Fig. 6. Error estimation for participant number

In Fig. 6(a), the number of participants is varied from 30 to 90. Two events and two sets of reliability per person are considered. We observe that, PBRE has a lower estimation error in participant reliability compared to EM scheme. Between two schemes of PBRE, Fit_Parent and Replace_Parent, Fit_Parent has much lower estimation error. This is because Fit_Parent takes only the fit values whereas Replace_Parent takes the fit set of values.

We run experiments for the increased number of participants from 300 to 900. The number in the set of reliability per person is 15. Event number is same as before e.i. two. Now, in Fig. 6(b), we observe that the error percentage decreases for Fit_Parent to 1 % which is compared to 10 to 15 % in Fig. 6 for participants with 4 sets of reliability per person. The reason behind this decline is due to the increased number in the set of reliability.

4.2 For Variable Number of Events

Here, we compare the results by varying the number of events from 2 to 10 for two cases.

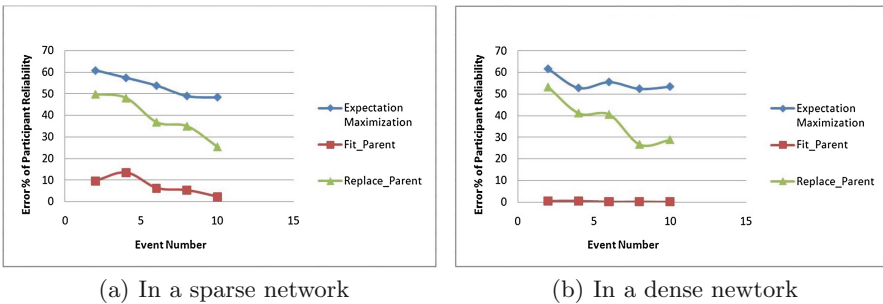


Fig. 7. Error estimation for the events

In Fig. 7(a), experiments are run for a sparse network of 50 participants, 2–10 events and 4 set of reliability per person. Here also, PBRE shows better results than EM because, when the event number increases, $target_{a_i}$ decreases (Line 6, Procedure PBRE). Therefore, there are more matches of a_i as fit_{a_i} to $target_{a_i}$. We run experiments for the increased number of participants to 600 in Fig. 7(b). Here also, PBRE shows better results than EM because, when the event number increases, $target_{a_i}$ decreases (Line 6, Procedure PBRE). Therefore, there are more matches of a_i as fit_{a_i} to $target_{a_i}$.

4.3 Convergence Rate

We study the convergence vs. estimation accuracy of PBRE and EM scheme by varying the number of participants from 30 to 80. Event number is fixed

at 2 and the set of reliability per person is at 4. In Fig. 8, we observe that the convergence rate for PBRE is a bit lower than the EM. Since PBRE has the lower error percentage of reliability than EM, it iterates more than EM to converge. Here, the convergence rate for Fit_Parent, Replace_Parent and EM are 0–2, 3.5–4.5 and 8–10 respectively.

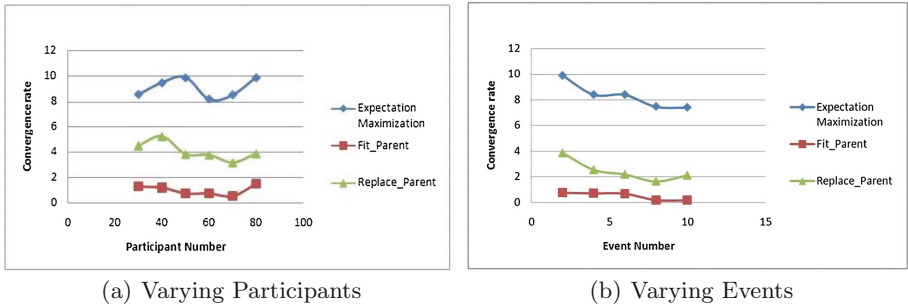


Fig. 8. Convergence rate.

We then examine the results by varying number of events from 2 to 10. The Participant number is fixed at 50 and set of reliability per person is at 4. In Fig. 8(b), we find that the convergence rate for PBRE is lower than the EM. This is because of the similar fact happening in Fig. 8(a). Here, convergence rate for Fit_Parent, Replace_Parent and EM are 0–1, 2–4 and 7.5–10 respectively. We also observe that the rate in Fig. 8(a) is lower than the rate in Fig. 8(b) for increased number of events which is natural since the number of events more it takes more time to converge.

5 Conclusion

In this paper, we study the challenge of finding the node reliability in a participatory sensor network. We propose Population Based Reliability Estimation (PBRE). We computed conditional probability of event to be true with the given set of reliability. We use Genetic Algorithm to estimate the reliability by iterating fitness assessment, breeding and joining. We vary the number of participants, the number of events. The metrics for performance measurement are the error percentage of participants' reliable reports and the convergence rate. We compare the results with the result of another relevant and popular method Expectation Maximization for the truth finding. We find that our approach provides better results.

In future, we would like to provide some hybrid approach to have better results. Besides, we have a bias about ground truth more than 50% probability. We would like to explore the impact of the uncertainty on our method.

References

1. Aggarwal, C., Abdelzaher, T.: Social sensing. In: *Managing and Mining Sensor Data*, pp. 237–297. Springer (2013)
2. Amintoosi, H., Allahbakhsh, M., Kanhere, S., Torshiz, M.N.: Trust assessment in social participatory networks. In: *International eConference on Computer and Knowledge Engineering (ICCKE)*, pp. 437–442 (2013)
3. Burke, J., Estrin, D., Hansen, M., Parker, A., Ramanathan, N., Reddy, S., Srivastava, M.B.: Participatory sensing. In: *World-Sensor-Web: Mobile Device Centric Sensor Networks and Applications*, pp. 117–134 (2006)
4. Choudhury, T., Philipose, M., Wyatt, D., Lester, J.: Towards activity databases: using sensors and statistical models to summarize peoples lives. *IEEE Data Eng. Bull.* **29**(1), 49–58 (2006)
5. Cox, L.P.: Truth in crowd sourcing. *IEEE Secur. Priv.* **9**(5), 74–76 (2011)
6. Deng, L., Cox, L.P.: Livecompare: Grocery bargain hunting through participatory sensing. In: *The Workshop on Mobile Computing Systems and Applications (Hot-Mobile)*, NY, USA, pp. 1–6. ACM (2009)
7. Dong, X.L., Berti-Equille, L., Srivastava, D.: Integrating conflicting data: the role of source dependence. *VLDB Endow.* **2**(1), 550–561 (2009)
8. Dong, X.L., Berti-Equille, L., Srivastava, D.: Truth discovery and copying detection in a dynamic world. *VLDB Endow.* **2**(1), 562–573 (2009)
9. Dua, A., Bulusu, N., Feng, W.-C., Hu, W.: Towards trustworthy participatory sensing. In: *USENIX Conference on Hot Topics in Security (HotSec)*, CA, USA (2009)
10. Eagle, N., Pentland, A.S., Lazer, D.: Inferring friendship network structure by using mobile phone data. *Natl. Acad. Sci.* **106**(36), 15274–15278 (2009)
11. Eiben, A.E., Smith, J.E.: *Introduction to Evolutionary Computing*. Springer, Heidelberg (2003)
12. Eisenman, S.B., Miluzzo, E., Lane, N.D., Peterson, R.A., Ahn, G.-S., Campbell, A.T.: The bikenet mobile sensing system for cyclist experience mapping. In: *International Conference on Embedded Networked Sensor Systems (SenSys)*, pp. 87–101. ACM (2007)
13. Estrin, D.L.: Participatory sensing: applications and architecture. In: *ACM MobiSys*, pp. 3–4 (2010)
14. Gilbert, P., Cox, L.P., Jung, J., Wetherall, D.: Toward trustworthy mobile sensing. In: *Computing Systems and Applications*, NY, USA, pp. 31–36 (2010)
15. Gilbert, P., Jung, J., Lee, K., Qin, H., Sharkey, D., Sheth, A., Cox, L.P.: YouProve: authenticity and fidelity in mobile sensing. In: *International Conference on Embedded Networked Sensor Systems (SenSys)*, NY, USA, pp. 176–189 (2011)
16. Hicks, J., Ramanathan, N., Kim, D., Monibi, M., Selsky, J., Hansen, M., Estrin, D.: Wellness: an open mobile system for activity and experience sampling. In: *Wireless Health (WH)*, pp. 34–43. ACM (2010)
17. Lane, N., Choudhury, T., Campbell, A.: Bewell: a smart phone application to monitor, model and promote well-being. In: *International ICST Conference on Pervasive Computing Technologies for Healthcare* (2011)

18. Le, H.K., Pasternack, J., Ahmadi, H., Gupta, M., Sun, Y., Abdelzaher, T.F., Han, J., Roth, D., Szymanski, B.K., Adali, S.: Apollo: towards fact finding in participatory sensing. In: ACM IPSN, pp. 129–130 (2011)
19. Lenders, V., Koukumidis, E., Zhang, P., Martonosi, M.: Location-based trust for mobile user-generated content: applications, challenges and implementations. In: ACM Workshop on Mobile Computing Systems and Applications (HotMobile), NY, USA, pp. 60–64 (2008)
20. Lu, H., Frauendorfer, D., Rabbi, M., Mast, M.S., Chittaranjan, G.T., Campbell, A.T., Gatica-Perez, D., Choudhury, T. Stress sense: detecting stress in unconstrained acoustic environments using smart phones. In: ACM Conference on Ubiquitous Computing (UbiComp), pp. 351–360 (2012)
21. Madan, A., Cebrin, M., Moturu, S.T., Farrahi, K., Pentland, A.: Sensing the health state of a community. *IEEE Pervasive Comput.* **11**(4), 36–45 (2012)
22. Madan, A., Moturu, S.T., Lazer, D., Pentland, A.S.: Social sensing: obesity, unhealthy eating and exercise in face-to-face networks. In: *Wireless Health (WH)*, pp. 104–110 (2010)
23. Olgun, D.O., Gloor, P.A., Pentland, A.: Wearable sensors for pervasive healthcare management. In: *Pervasive Health*, pp. 1–4 (2009)
24. Oliveira, M.P.G., Medeiros, E.B., Davis, C.A.: Planning the acoustic urban environment: a gis centered approach. In: *GIS*, pp. 128–133 (1999)
25. Reddy, S., Shilton, K., Denisov, G., Cenizal, C., Estrin, D., Srivastava, M.: Bike-tastic: sensing and mapping for better biking. In: *ACM Conference on Human Computer Interaction (CHI)*, pp. 1817–1820 (2010)
26. Roy, D., et al.: The human speechome project. In: Vogt, P., Sugita, Y., Tuci, E., Nehaniv, C.L. (eds.) *EELC 2006. LNCS (LNAI)*, vol. 4211, pp. 192–196. Springer, Heidelberg (2006)
27. Ryder, J., Longstaff, B., Reddy, S., Estrin, D.: Ambulation: A tool for monitoring mobility patterns over time using mobile phones. In: *CSE*, vol. 4, pp. 927–931. *IEEE* (2009)
28. Srivastava, M., Abdelzaher, T., Szymanski, B.: Human-centric sensing. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **370**(1958), 176–197 (2012)
29. Tang, L.A., Yu, X., Kim, S., Han, J., Hung, C.-C., Peng, W.C.: Tru-alarm: Trustworthiness analysis of sensor networks in cyber-physical systems. In: *International Conference of Data Management (ICDM)*, pp. 1079–1084 (2010)
30. Tilak, S.: Real-world deployments of participatory sensing applications: current trends and future directions. *ISRN Sens. Netw.* **2013**, 8 (2013)
31. Wang, C., Burris, M.A., Ping, X.Y.: Chinese village women as visual anthropologists: a participatory approach to reaching policy makers. *Soc. Sci. Med.* **42**(10), 1391–1400 (1996)
32. Wang, D., Abdelzaher, T., Ahmadi, H., Pasternack, J., Roth, D., Gupta, M., Han, J., Fatemieh, O., Le, H., Aggarwal, C.: On bayesian interpretation of fact-finding in information networks. In: *Information Fusion (FUSION)*, pp. 1–8 (2011)
33. Wang, D., Abdelzaher, T., Kaplan, L., Aggarwal, C.: On quantifying the accuracy of maximum likelihood estimation of participant reliability in social sensing (2011)
34. Wang, D., Abdelzaher, T., Kaplan, L., Aggarwal, C.: Recursive fact-finding: a streaming approach to truth estimation in crowd sourcing applications. In: *International Conference on Distributed Computing Systems (ICDCS)* (2013)

35. Wang, D., Kaplan, L., Le, H., Abdelzaher, T.: On truth discovery in social sensing: a maximum likelihood estimation approach. In: ACM International Conference on Information Processing in Sensor Networks (IPSN), pp. 233–244 (2012)
36. Yin, X., Han, J., Yu, P.S.: Truth discovery with multiple conflicting information providers on the web. In: International Conference on Knowledge Discovery and Data Mining (KDD), New York, USA, pp. 1048–1052 (2007)
37. Yin, X., Tan, W.: Semi-supervised truth discovery. In: ACM International Conference on World Wide Web, New York, USA, pp. 217–226 (2011)