A Reputation System to Evaluate Participants for Participatory Sensing

Mohannad A. Alswailim*, Hossam S. Hassanein and Mohammad Zulkernine
School of Computing
Queen's University
Kingston, ON, Canada K7L 3N6
{mohannad | hossam | mzulker} @cs.queensu.ca

***Abstract*— Participatory sensing is an approach to data collection that offers individuals and groups the opportunity to participate in an application using their sensor devices. Receiving contributions from multiple individuals may result however, in corrupted and inaccurate sensed data. Therefore, it becomes important to be able to have enough knowledge about each participant in order to evaluate their contributions and trustworthiness. In this paper, we present a Reputation System to Evaluate Participants (RSEP). The RSEP, starts with grouping participants based on their contributions, and then selects the highest group value based on its participant reputation values. The RSEP filters the sensed data to separate out the most accurate contributions that enhance the purpose of the participatory sensing applications. Experimental results show that the proposed RSEP has a high accuracy level in evaluating and selecting participant contributions.**

***Index Terms*— participatory sensing; reputation system; data quality.**

I. INTRODUCTION

Participatory sensing applications [1] allow individuals and groups to participate in an application by sensing, collecting and sending the data to an application server. The server processes the collected sensor data and makes them available to the end-users. Existing smartphones allow participants to contribute in participatory sensing applications by using their embedded or peripheral sensors such as GPS, microphone, camera, ambient light, accelerometer and proximity to collect data. Smartphones, moreover, allow participants to send the collected sensor data through the Internet using LTE and Wi-Fi [2]. Participatory sensing enables a variety of applications that help communities, companies and governments in areas such as urban planning, community safety, public health, transportation and traffic monitoring [3].

Sensor devices have three sensing modes when collecting data that are at the participant involvement level. “Manual” mode requires the participant to perform the task for every sensing contribution. “Automatic” mode acts periodically set by the participant. “Opportunistic” mode takes an action when it satisfies the application's conditions [4].

Participatory sensing is not the only approach that carries those definitions, requirements and goals. There are other similar approaches such as public sensing [4], mobile sensing [5], crowd sensing and opportunistic sensing [6]. These terms are used interchangeably in the field of participatory sensing research. Sensing modes may differ between these similar approaches.

Participatory sensing applications are usually open to the public, and receive sensor data from multiple participants. This openness concept however, allows inaccurate and corrupted contributions which negatively affect the quality of the application services [7, 8]. Therefore, validating the accuracy of contributions is essential for participatory sensing applications. A way of evaluating contribution validity is by evaluating participant reputation values.

In this paper, we propose a Reputation System to Evaluate Participants (RSEP) and validates their contributions. The RSEP computes each participant reputation value based on their previous contributions results to help validate their current contribution. At the end of each round, the system rates all contributions and assigns scores to each participant based on their contributions validity.

The RSEP consists of two major phases: (1) selecting the most accurate contributions and (2) updating participant reputation values. Each phase must follow multiple steps to complete its tasks including various equations that compute individuals and groups reputation values.

We implement the RSEP on a real-world application dataset that collects sensor data from multiple participants [9]. They are a fleet of taxicabs in the city of Rome, Italy. They use their sensor devises to collect outside temperatures from different grids at different periods of the day. The application uses the collected temperatures for weather forecasting.

We perform empirical validations that show the efficiency of the RSEP in assessing the correctness of the contributions and in evaluating their participants. We compare the RSEP to an existing reputation system. The RSEP will enhance the services of the participatory sensing applications. It filters out the inaccurate contributions and passes only the accurate contributions to the application servers.

The remainder of this paper is organized as follows. In Section II, we discuss several related works. Section III details our proposed reputation system and its algorithms. In Section IV, we describe the experimental evaluation and setup, and discuss the evaluation results. Section V concludes our work.

* Mohannad A. Alswailim is also affiliated with Qassim University (QU), Qassim, Saudi Arabia.

978-1-5090-1328-9/16/$31.00 ©2016 IEEE

II. RELATED WORK

Reputation systems have been studied in various disciplines. Here, we are going to discuss related work in the field of participatory sensing as well as e-marketing.

Many online markets connect sellers with buyers for commercial purposes such as eBay[1], AliBaba[2] and Amazon.[3] These e-markets use reputation systems to allow buyers and sellers to rate each other based on their transaction satisfaction such as delivery time, quick payments, etc. For example, eBay has a reputation system where buyers can rate the sellers, and vice versa, by leaving a feedback rating of positive, neutral or negative after each transaction. The higher the positive feedback, the higher the reputation value. The eBay method of calculating a customer's reputation value can be obtained by dividing the positive feedback scores by the number of responded feedbacks in the last 12 months. This is considered long period and may not reflect the recent status of a customer.

In another aspect, reputation systems have also been studied in participatory sensing [10, 11, 12, 13]. Amintoosi and Kanhere [10] proposed a reputation system that is compatible with social networks to distribute participatory sensing tasks. The social network members could be requesters of a service or participants. The computations involved, in assessing participant contribution reputations, are influenced by the requester, who is able to add their evaluations to the contribution reputation. Yang et al. [11] suggested a reputation system that aims to improve data quality by classifying participants before starting the tasks. The requesters set a list of conditions that participants must satisfy. The system ranks participants who meet the requesters conditions based on the participant reputation values. Then, requesters select the desired rank level of participants. Restuccia and Das [12] projected a reputation system that maintains a list of trusted participants who always sense accurate data and are protected from external attacks. Thus, when other participants contribute in an area, the system checks if a trusted participant is nearby. Then, the system evaluates participant contributions based on the trusted participant contribution. Manzoor et al. [13] proposed a reputation system that relies on participants' previous contributions that have been evaluated based on the sensor data quality. The system assesses the contributions quality by passing them through a quality evaluator component. The quality evaluator's results and participant reputation history are the inputs to calculate the participant's current reputation. This reputation system uses similar inputs to compute participant reputation as our RSEP.

The above systems try to fulfil application requirements to maintain reliable participants and high quality contributions. These systems, however, consider the requirements that match their applications by applying strong assumptions that may not be feasible for other applications. Our RSEP is designed to be comprehensive and covers a wider range of application needs.

III. THE RSEP OVERVIEW

In this section, we provide the overview of the proposed reputation system for participatory sensing in Section A. Sections B and C discuss the algorithms of the RSEP and participant reputation value computation, respectively.

1 http://www.ebay.com/ 2 http://www.alibaba.com/ 3 http://www.amazon.com/

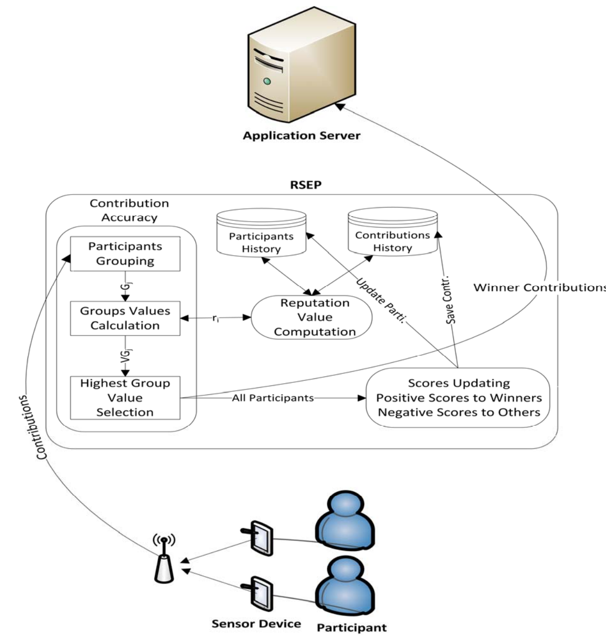


Fig. 1: RSEP Architecture for Participatory Sensing

A. Overview

Fig. 1 shows the data flow from participant handheld devices to the application server passing through the RSEP. Participants start sensing using their sensor devices and send the collected sensor data to the RSEP. It processes the received data and sends only the most accurate contributions to the participatory sensing application server. The RSEP consists of two major phases: (1) selecting the most accurate contributions and (2) updating participant reputation values.

In phase one, participant contributions go through three steps to evaluate them and to ease the selection of the most accurate contributions to be sent to the application server (see Algorithm 1, Section III.B). The first step of phase one starts with dividing participants into three groups. Second, it evaluates each group by calculating their values. To calculate a group value, the system requires the calculations of each participant reputation (Algorithm 2) in the group and each participant weight among other participants in the same group. Third, it selects the contributions of the highest group value and sends them to the application server.

In phase two, the RSEP applies the final step by updating the participant scores based on the accuracy of their contributions. It assigns positive scores to the participants of the selected group (highest group value) and negative scores to the remaining participants.

There are two repositories at the RSEP to save the information collected from the results of the two phases. One saves participant history and the other saves contribution history. The participant history repository saves each participant's information. After each contribution, the system updates the participant information and computes the new reputation value. The contribution history repository saves all the contributions made by participants. After each contribution, the system records the contribution information.

The RSEP provides two modes of operations, *aging* and *newcomer*, to help make decisions in selecting accurate contributions. Aging allows the system to only consider the participant reputation value of the most recent contributions. In addition, it gives an opportunity for disreputable participants to re-establish their reputation status. By using contribution history repository, aging is able to only consider the last Z contributions, days or months in its calculation of participant reputation values.

In the case of a newcomer participant, the system assigns an initial reputation value of 0.5. If a newcomer receives a positive score in their first contribution, the reputation value will increase to 1 (100%). On the contrary, if the newcomer receives a negative score, the reputation value will drop to 0. Therefore, the RSEP proposes to continue assigning a reputation value of 0.5 for the first k contributions to avoid the fluctuation of the newcomer reputation value. The RSEP counts the positive and negative scores that a newcomer receives during the *k-th* contributions to be used in computing the actual reputation value in the (*k+1*)-*th* contribution.

B. The RSEP Algorithm

In Algorithm 1, the RSEP starts by gathering all participant contributions under certain conditions such as location and time, as system inputs. It then selects the most accurate contributions and updates participant scores as system outputs.

After gathering all contributions, the system clusters participants into three groups: G_j, $j = \{1, 2, 3\}$. Its clustering is based on the participant sensed data (s), as in lines 1 to 6, where clustering parameter $(cp) = 0.10$ *(10%)*.

Next, the system calculates each group value (V_{G_j}). This step requires computing participant reputation values (r) from Algorithm 2 in Section III.C. When participant reputation values are computed (line 7), the system calculates each participant's weight (w) among other participants in the same group, as in lines 8 to 12. The absolute total of all participant weights in the same group is one (100%). A participant weight is computed by dividing the reputation value of *i-th* participant (r_i), $i = \{1, \ldots, n\}$, over the summation of all participant reputation values in the same group G_j as follows:

$$w_i = \frac{r_i}{\sum_{i \varepsilon G_j} r_i} \quad (1)$$

Line 13 shows the last step of computing V_{G_j} that depends on Eq. 1.

$$V_{G_j} = \sum_{i \varepsilon G_j} w_i * r_i \quad (2)$$

The previous steps are applied on each of the three groups to get their V_{G_j}. Next step compares between the three group values and selects the highest V_{G_j} as the winner group, as shown in lines 14 to 18. To this point, the system is able to send the contributions of the winner G_j to the application server as the most accurate contributions.

Finally, the RSEP updates the reputation values of the participants for their contributions. Lines 19 to 26 show the steps of the updating task by assigning positive scores (ps) and negative scores (ns) to the participants based on their group values. Thus, each participant in the winner group receives a ps as an award for their accurate contributions. On the contrary, each participant in the other two groups receives an ns for their inaccurate contribution. These new ps and ns change the reputation value of each participant either up or down based on the assigned score as in Eq. 3. Let the reputation value of *i-th* participant (r_i) be 75% from a total of 30 positive scores (ps_i) and 10 negative scores (ns_i) from its previous contributions. If it receives a ps in a new contribution, r_i becomes 75.6%. In contrast, if it receives an ns in a new contribution, r_i becomes 73.2%.

Algorithm 1 Reputation System
Input: Periodic Participants Contributions
Output: Accurate Contributions; Scores Update

Participants Grouping (G_j)
1. **for** $i \leftarrow 1$ **to** n **do**
2. **if** $(\mu - cp) \leq (s_i) \leq (\mu + cp)$
3. **then** Add s_i to G_1
4. **else** **if** $(s_i) < (\mu - cp)$
5. **then** Add s_i to G_2
6. **else** Add s_i to G_3

Group Values Calculation (V_{G_j})
7. **Get** participant reputation values (r) *// Call Algorithm 2*
8. **for** $j \leftarrow 1$ **to** 3 **do**
9. **for** $i \leftarrow 1$ **to** $|G_j|$ **do**
10. $R_j \leftarrow \sum r_i$
11. **for** $i \leftarrow 1$ **to** $|G_j|$ **do**
12. $w_i \leftarrow \frac{r_i}{R_j}$
13. $V_{G_j} \leftarrow \sum w_i * r_i$

Highest Group Value Selection (HGV)
14. **if** $V_{G_2} \leq V_{G_1} \geq V_{G_3}$
15. **then** $HGV \leftarrow V_{G_1}$
16. **else** **if** $(V_{G_2} \geq V_{G_3})$
17. **then** $HGV \leftarrow V_{G_2}$
18. **else** $HGV \leftarrow V_{G_3}$

Send s_i of HGV to the Application Server

Positive (ps) and Negative (ns) Scores Updating
19. **for** $j \leftarrow 1$ **to** 3 **do**
20. **if** $V_{G_j} = HGV$
21. **then**
22. **for** $i \leftarrow 1$ **to** $|G_j|$ **do**
23. $ps_i \leftarrow ps_i + 1$
24. **else**
25. **for** $i \leftarrow 1$ **to** $|G_j|$ **do**
26. $ns_i \leftarrow ns_i + 1$

C. Reputation Value Computation Algorithm

Participant reputation values reflect their trustworthiness which plays a major role in allowing the system to make a selection decision. Algorithm 2 shows the steps to calculate a participant reputation value (r). It can be calculated under the following two conditions. (1) If a participant is a newcomer (nc)

Algorithm 2 Participant Reputation Values Computation

Input: Participants ID
Output: Participant Reputation Values (r)

1. **for** $i \leftarrow 1$ **to** n **do**
2.   **if** $X_i < k$ *// Newcomers mode*
3.   **then** $r_i \leftarrow nc$
4.   **else** **if** $X_i < Z$ *// Aging mode*
5.   **then** $Y \leftarrow X_i$
6. $PS_i \leftarrow ps_i$
7.   **else** $Y \leftarrow Z$
8.   **for** $b \leftarrow X_i - (Z-1)$ **to** X_i **do**
9. $PS_i \leftarrow \sum ps_b$
10. $r_i \leftarrow \frac{PS_i}{Y}$

(lines 2 and 3) the system assigns an initial $r = nc$, $nc = 0.50$ *(50%)* as starting value, then the participant builds its reputation value through the future contributions. (2) If a participant is not a newcomer (lines 4 to 9), the system compares and takes the minimum number of contributions of X or Z, where X is the participant's total number of contributions and Z is the number of contributions that need to be considered for aging mode. If X is the minimum (lines 4 to 6), the total positive score of *i-th* participant (PS_i) equals the total positive scores in the previous contributions. On the contrary, if Z is the minimum (lines 7 to 9), PS_i is the total positive scores in the last Z contributions only.

Finally, line 10 computes the reputation value of *i-th* participant (r_i) using the previous inputs of PS_i and Y as follows:

$$r_i = \frac{PS_i}{Y} \tag{3}$$

IV. Experimental evaluation

In this section, we discuss our experimental evaluation results of the RSEP. In Section A, we review the implementation setup and evaluation environment. Section B explains the dataset we use in the implementation. Finally, the evaluation metrics to assess the accuracy of the RSEP results are discussed in Section C.

A. Evaluation Environment

Our implementation uses a real-world dataset that consists of a fleet of taxicabs as participants (see Section IV.B for details). The participant collects sensor data, attaches date, time and location, and then sends the report to the RSEP. The RSEP continues receiving contributions from the participants for each grid until the end of the current period. The RSEP then starts processing all the contributions from that period as discussed in Section III. It selects the most accurate contributions and sends them to the application server. Finally, the RSEP updates the participant scores and saves them for the next participation.

B. Dataset

We implement our work on adapted real-world data that is publicly available on Crawdad archive[4] [9] at Dartmouth College. The original dataset [14] collects the GPS position of taxicabs moving around the city of Rome, Italy. Each driver is equipped with an Android OS tablet device running an app that uses the GPS sensor to update their current position.

We limit our analysis to cover a large area of the center of Rome, where the density of taxicabs is high. We consider an area of approximately 22.5x22.5 [km x km] whose bounds are given by the coordinate pairs (41.9951, 12.3648) and (41.7887, 12.6283). The area has been analyzed using 3x3 grids. A grid cell covers a square area of approximately 7.5x7.5 [km x km].

For all the presented results, we analyze the sensing of 289 taxicabs over 4 days. We split each day into four periods, six hours each. The periods are as follows: early morning [00:00 – 05:59], late morning [06:00 – 11:59], afternoon [12:00 – 17:59] and evening [18:00 – 23:59].

We modify the original data by assuming that each taxicab is equipped with a peripheral temperature sensor. When the taxicab is moving, the sensor collects the outside temperature of its current grid and records the period of day. In each grid, we assume that each taxicab is allowed to participate once during each period of a day.

We generate a temperature value for every active taxicab in a certain period by applying Gaussian distribution. To fill out the parameters of Gaussian function, we assign a mean (μ) and standard deviation (δ) for every run. The mean value (μ) corresponds to the ground truth temperature for every period in every grid on every day. We use The Weather Network[5] statistics to assign the correct value ground truth to the specific period and grid location. For every taxicab, we assign a fixed error range (δ) that remains the same in all of its contributions. To do so, we randomly classify participant taxicabs into three classes. First class, called "honest," consists of taxicabs that usually sense accurate temperatures within a 10% error range from the ground truth. The population of honest class is 145 taxicabs (50% of all participant taxicabs). Second class, called "dishonest," consists of taxicabs that usually sense inaccurate temperatures between 10% and 30% error range from the ground truth. The population of the dishonest class is 72 taxicabs (25%). Third class, called "misleading," consists of the remainder of the participant taxicabs that is 72 (25%) that can report accurate or inaccurate temperatures. The data generator function makes a random decision to generate accurate or inaccurate temperatures for each taxicab among the misleading class. The latter class plays a major role in the results of the reputation system since the accuracy of their contributions is not consistent. The dataset assigns for each taxicab a sensed temperature contribution based on its fixed error range and the ground truth of the day, period and grid of its location.

C. Experiment Results

Collecting sensor data by the participants and sending this data to the application server are major tasks in participatory sensing systems. However, assessing those participants and selecting the accurate contributions are essential tasks for the application services. Hence, the RSEP aims to apply these essential tasks efficiently. To evaluate the RSEP results, we use two metrics: *False Positive (FP)* and *False Negative (FN)* rates.

FP means that a participant is classified as dishonest before running the system but the results show that its reputation value is above the *threshold*. On the contrary, FN means that a participant is classified as honest before running the system but

4 http://crawdad.org/queensu/crowd_temperature/20151120

5 http://www.theweathernetwork.com/

the results show that its reputation value is below the threshold. The threshold is a percentage that is applied to the participant reputation values to distinguish honest participants from dishonest participants. Threshold value can be set based on the application requirements. FP and FN have an inverse correlation that depends on the threshold. The higher the threshold, the lower the FP, and the higher the FN.

FP and FN rates are to measure the accuracy of the RSEP in assessing participants. As discussed in Section IV.B, we randomly classified participant taxicabs into honest, dishonest and misleading. We generated the temperature degrees for the participant taxicabs with given error range based on their classes. Now, after running the system on all valid contributions and rewarding the accurate contributions with positive scores and the inaccurate contributions with negative scores, we check how accurate the RSEP is in assessing participants.

To evaluate the RSEP results based on FP and FN rates, we create three sets of participants: set_1 and set_2 consist of 50 and 150 randomly selected participant taxicabs, respectively, while set_3 consists of all 289 participant taxicabs. We set the metrics to three different thresholds: 60%, 70% and 80% to show the accuracy rates of assessing participants in different settings. We execute the metrics by applying each threshold on all three sets and get the FP and FN rates. To compute FP and FN rates, we apply the following:

$$FP\ rate = \frac{D}{|subset|} \tag{4}$$

$$FN\ rate = \frac{H}{set_m - |subset|} \tag{5}$$

Here, D is the number of dishonest participants whose reputation values are above the threshold. H is the number of honest participants whose reputation values are below the threshold. *Subset* is total number of participants whose reputation values are above the threshold in set_m, $m = \{1, 2, 3\}$, as in Table 1.

With a threshold of 60%, set_1 has a *subset* of 37 participant taxicabs above the threshold and 13 participant taxicabs below the threshold. By executing FP and FN rates in Eq. 4 and Eq. 5, we find that FP rate is 3% and FN rate is 0%. The FP rate of 3% is due to 1 participant taxicab that has a reputation value above the threshold after running the system, while it was classified among the dishonest class before running the system. On the contrary, no single participant taxicab is classified as honest and the system shows that it is dishonest. In set_2, we get a *subset* of 110 participant taxicabs above the threshold and 40 participant taxicabs below the threshold. The FP rate is 6% and FN rate is 2%. In set_3, a *subset* of 211 participant taxicabs above the threshold with FP and FN rates of 7% and 1%, respectively.

Table 1 also shows the FP and FN rates when setting the threshold to 70%. We get a *subset* of 35, 101 and 195 participant taxicabs above the threshold with FP rates of 3%, 4% and 5%, and FN rates of 7%, 2% and 1%, for set_1, set_2 and set_3, respectively.

With a threshold of 80%, we get a *subset* of 28, 84 and 168 participant taxicabs above the threshold with FP rates of 0%, 0% and 2%, and FN rates of 9%, 5% and 5%, for set_1, set_2 and set_3, respectively.

Threshold	Parameters	Set_1 (50 Taxicabs)	Set_2 (150 Taxicabs)	Set_3 (289 Taxicabs)
60%	Subset	37	110	211
	D	1	7	15
	H	0	1	1
	FP	3%	6%	7%
	FN	0%	2%	1%
70%	Subset	35	101	195
	D	1	4	10
	H	1	1	1
	FP	3%	4%	5%
	FN	7%	2%	1%
80%	Subset	28	84	168
	D	0	0	4
	H	2	3	6
	FP	0%	0%	2%
	FN	9%	5%	5%

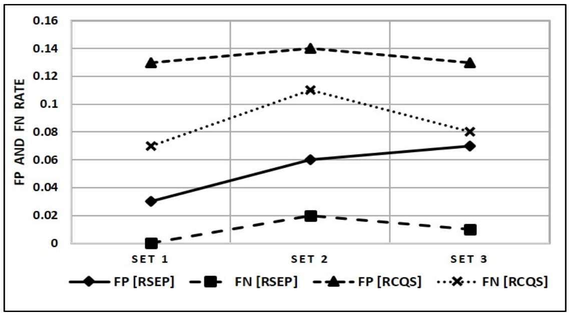
Table 1: FP and FN rates and their parameters with thresholds settings of 60%, 70% and 80%, and three sets of participants set_1, set_2 and set_3 for RSEP

In order to understand the advantage that participatory sensing applications gain from using the RSEP, we implement the Reputation and Contribution Quality System (RCQS) [13] to compare our system to another existing system. We focus on this particular system because it mainly depends on the current and previous participant contributions quality when it computes the reputation value, unlike other systems where they give the service requesters or users leverage to affect on the reputation value computation. We implement the RCQS on our dataset to get a reasonable comparison.

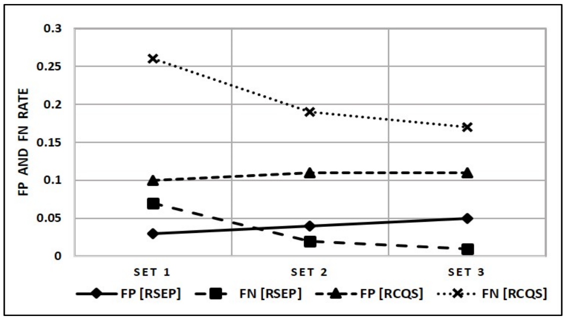
For the RCQS scheme, we measure the steps of assessing participants and evaluating their reputations. The inputs for the system in a period are a prediction value that is derived from the participant contribution and a ground truth value that is received from the end user.

These inputs pass through three main components that analyze the possible data errors, evaluate the contribution quality and finally evaluate participant reputation as the system output. Last component computes the reputation value based on two factors, data quality level that is received from the previous component and participant history. Next, we apply the FP and FN rate metrics on those final participant reputation values to evaluate the system accuracy in assessing participants.

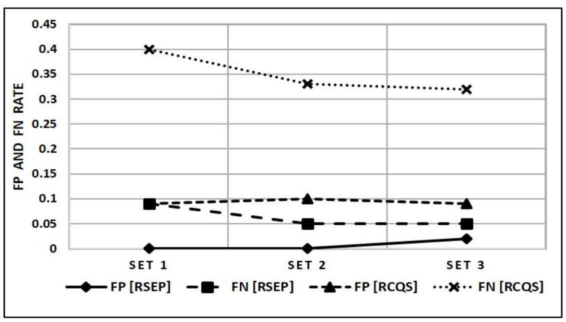
Fig. 2 illustrates the FP and FN rates for the RSEP and RCQS with the same three thresholds of 60%, 70% and 80% in a, b and c, respectively. The lower the rate, the better the accuracy in assessing participants. Therefore, the results in all three thresholds show that the RSEP has a better performance than RCQS in both metrics. In Fig. 2.a, for example, the FP rates in the RSEP is 3%, 6% and 7%, while they are 13%, 14% and 13% in RCQS for set_1, set_2 and set_3, respectively. FN rates are 0%, 2% and 1% in the RSEP and 7%, 11% and 8% in RCQS for set_1, set_2 and set_3, respectively.



a. Threshold of 60%



b. Threshold of 70%



c. Threshold of 80%

Fig. 2: FP and FN rates with three threshold settings of 60%, 70% and 80% for RSEP and RCQS

The different factors used in RCQS during the reputation computation causes having higher rates. Combining the two factors, data quality level and reputation history, in the computation without assigning a reward mechanism produces a fluctuation of reputation values during a few periods. Consequently, some honest participants may have lower reputation values than what they deserve because of only a few inaccurate contributions, and vice versa.

V. Conclusion

Participatory sensing applications need to verify participant contributions to select useful collected sensor data and enhance the application services. We propose a Reputation System to Evaluate Participants (RSEP), to achieve the participatory sensing needs. The RSEP goes through two phases. One selects the most accurate contributions and the other updates participant reputation values. We showed the steps to fulfil the first phase by dividing participants into three groups based on their collected sensor data. Then, the RSEP evaluates each group using its participant reputation values and selects the contributions of the highest group value. The second phase assigned positive and negative scores to participants based on their contributions accuracy. These new scores change the reputation values of the participants in the next contribution.

Our evaluations, based on a real-world dataset, showed that the accuracy level of assessing participants using the RSEP is high, as compared to an existing reputation system. We used the false positive and false negative rate metrics to evaluate both systems with three threshold settings on three sets of random participants.

Acknowledgment

This research is supported by a grant from the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant number: STPGP 479248.

References

[1] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava, "Participatory Sensing," in *Proceedings of the ACM International Workshop on World-Sensor-Web*, pp. 117–134, 2006.

[2] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, "A survey of mobile phone sensing," *IEEE Communications Magazine*, vol. 48, no. 9, pp. 140–150, 2010.

[3] K. Herrmann, D. Fischer, and D. Philipp, "Resource management for public sensing," in *Proceedings of the 26th AAAI Conference on Artificial Intelligence Workshop*, 2012.

[4] D. Christin, A. Reinhardt, S. S. Kanhere, and M. Hollick, "A survey on privacy in mobile participatory sensing applications," *Journal of Systems and Software*, vol. 84, pp. 1928– 946, 2011.

[5] D. Estrin, "Participatory sensing: applications and architecture," *IEEE Internet Computing*, vol. 14, no. 1, pp. 12–42, 2010.

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

[7] P. Gilbert, L. P. Cox, J. Jung, and D. Wetherall, "Toward trustworthy mobile sensing," in *Proceedings of the 11th ACM Workshop on Mobile Computing Systems and Applications*, (HotMobile), pp. 31–36, 2010.

[8] H. Mousa, S. B. Mokhtar, O. Hasan, O. Younes, M. Hadhoud, and L. Brunie, "Trust management and reputation systems in mobile participatory sensing applications: A survey," *Computer Networks*, vol. 90, pp. 49 – 73, 2015.

[9] M. A. Alswailim, H. S. Hassanein, and M. Zulkernine, "CRAWDAD dataset queensu/crowd_temperature," 2015.

[10] H. Amintoosi and S. Kanhere, "A reputation framework for social participatory sensing systems," *Mobile Networks and Applications*, vol. 19, no. 1, pp. 88–100, 2014.

[11] H. Yang, J. Zhang, and P. Roe, "Using reputation management in participatory sensing for data classification," *Procedia Computer Science*, vol. 5, pp. 190 – 197, 2011.

[12] F. Restuccia and S. Das, "Fides: A trust-based framework for secure user incentivization in participatory sensing," in *IEEE 15th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 1–10, 2014.

[13] A. Manzoor, M. Asplund, M. Bouroche, S. Clarke, and V. Cahill, "Trust evaluation for participatory sensing," in *Mobile and Ubiquitous Systems: Computing, Networking, and Services*, pp. 176–187, Springer, 2013.

[14] L. Bracciale, M. Bonola, P. Loreti, G. Bianchi, R. Amici, and A. Rabuffi, "CRAWDAD dataset roma/taxi," 2014.