

A Reputation Framework for Social Participatory Sensing Systems

Haleh Amintoosi · Salil S. Kanhere

Published online: 10 August 2013
© Springer Science+Business Media New York 2013

Abstract Social participatory sensing is a newly proposed paradigm that tries to address the limitations of participatory sensing by leveraging online social networks as an infrastructure. A critical issue in the success of this paradigm is to assure the trustworthiness of contributions provided by participants. In this paper, we propose an application-agnostic reputation framework for social participatory sensing systems. Our framework considers both the quality of contribution and the trustworthiness level of participant within the social network. These two aspects are then combined via a fuzzy inference system to arrive at a final trust rating for a contribution. A reputation score is also calculated for each participant as a resultant of the trust ratings assigned to him. We adopt the utilization of PageRank algorithm as the building block for our reputation module. Extensive simulations demonstrate the efficacy of our framework in achieving high overall trust and assigning accurate reputation scores.

Keywords Reputation framework · Trust · Participatory sensing · Online social networks · Data quality · Urban sensing · Fuzzy logic · Ranking algorithms

1 Introduction

The widespread prevalence of mobile computing devices such as sensor-rich smartphones has propelled the

emergence of a novel sensing paradigm, known as *participatory sensing* [1]. In participatory sensing, ordinary citizens volunteer to use their mobile phones for collecting sensor data from their nearby environment. The aim of such sensor gathering includes computing the aggregate statistics about a phenomenon, and increasing the global awareness of issues of interest. A plethora of applications have been proposed based on this revolutionary paradigm, ranging from personal health [2] and sharing prices of consumer goods [3] to monitoring various aspects of the urban environment such as air pollution [4].

For a participatory sensing system to be a success, one of the key challenges is the recruitment of sufficient well-suited participants, those who are suitable for tasks which require domain-specific knowledge or expertise [5]. Typically, in participatory sensing campaigns there is no explicit incentive for participation. Without adequate motivation, participants may not be willing enough to contribute, which in turn, reduces the data reliability. Another related challenge is the trustworthiness of the sensor data due which is a big concern due to the open nature of participatory sensing systems. Without confidence on the trustworthiness of sensor data, the obtained information will be of little use.

To address the above mentioned challenges, one proposed idea is to leverage online social networks as the underlying substrate for participatory sensing applications [6, 7]. This resulting paradigm, known as *social participatory sensing*, offers the following advantages. First, public profile information such as interests, expertise and education can facilitate the identification of well-suited participants. Second, social friendship relations act as an effective motivation to contribute to tasks created by friends, since people normally like to be helpful to their friends. Third, incentives in the form of e-coins [8] or reputation points will be devoted to well-behaved participants, and can be

H. Amintoosi (✉) · S. S. Kanhere
The University of New South Wales,
Sydney, Australia
e-mail: haleha@cse.unsw.edu.au

S. S. Kanhere
e-mail: salilk@cse.unsw.edu.au

publicly available in participants' profile. As an example of a real-world social participatory sensing platform, in Demirbas et al. [9], Twitter was used as the underlying social network substrate. The proposed system was tested in the context of two applications: weather radar and noise mapping. Their experiment resulted in a considerable smartphone-based participation from Twitter members even without an incentive structure. This clearly demonstrated the suitability of online social networks as a publish-subscribe infrastructure for tasking/utilizing smartphones and pave the way for ubiquitous crowd-sourced sensing and social collaboration applications.

The open nature of participatory sensing which allows everyone to contribute, while valuable for motivating participants, facilitates erroneous and untrusted data preparation. When combined with social network, new trust issues arise. For instance, following the devastation incurred due to Hurricane Sandy in the US in October 2012, social media was flooded with misinformation and fake photos.¹ While some of these were easy to identify as fake data (e.g., photo-shopped images of sharks swimming in New York streets), several other fake pictures and reports were initially thought to be true. In fact, the widespread use of social networks, along with fast and easy-to-use dissemination facilities such as re-sharing (a fake photo) or re-tweeting (a false event) make it difficult to identify the origin of the data and investigate its credibility. This clearly highlights the need for a trust system which is responsible for performing necessary validations both from the perspective of data trustworthiness and also the reliability of data contributors. In other words, it is important to know who and with what level of social trustworthiness produces the data and how much of the data can be trusted. While there exist works that address the issue of data trustworthiness in participatory sensing (see Section 2), they do not provide means to include social trust and as such cannot be readily adopted for social participatory sensing.

In this paper, we present an application agnostic framework to evaluate trust in social participatory sensing systems. Our system independently assesses the quality of the data and the trustworthiness of the participants and combines these metrics using fuzzy logic to arrive at a comprehensive trust rating for each contribution. These trust ratings are then used to calculate and update the reputation score of participants. By adopting a fuzzy approach, our proposed system is able to concretely quantify uncertain and imprecise information, such as trust, which is normally expressed by linguistic terms rather than numerical values. This paper is based on the basic idea proposed in our previous work [30]. Specifically, the following

new concepts and components have been added to this paper:

1. In our previous paper [30], we proposed equations for evaluating personal and social factors of trustworthiness of participant without indicating the input parameters for these equations. So, in our previous paper, we randomly assigned a value to trustworthiness of participant as a whole, and did not go through the evaluation of each and every effective parameter inside the factors. Here, we performed extensive investigations to set the parameters in a way that results in meaningful and valid values for each factor.
2. We have introduced the new concept of subjective rating (called Requester Evaluation (RE)) which allows the requester to evaluate the contribution to see how much the contribution satisfies his needs and desires.
3. We add a Reputation Module to calculate the reputation score for each participant. The reputation score can be used in different ways, depending on the functionality of the framework. In our framework, we assume that each node acts as a participant in one campaign and as a requester in another campaign. While serving as a requester, his reputation score serves as a weight for the evaluation that he assigns to each contribution. In other words, we assume that the requester's subjective evaluation has a weight, which is equal to the reputation score of the requester.

We undertake extensive simulations to demonstrate the effectiveness of our reputation framework and benchmark against the state-of-the-art. The results demonstrate that considering social relations makes trust evaluation more realistic, as it resembles human behaviour in establishing trustful social communications. We also show that our framework is able to quickly adapt to rapid changes in the participant's behaviour by prompt and correct detection and revocation of unreliable contributions and accurate update of participant's reputation score. Moreover, we find that leveraging fuzzy logic provides considerable flexibility in combining the underlying components which leads to better assessment of the trustworthiness of contributions. Our framework results in a considerable increase in the overall trust over a method which solely associates trust based on the quality of contribution. We should acknowledge that the focus of the work is on trust management for social participatory sensing systems and we do not consider security/privacy attacks in this paper. In fact, issues related to privacy preservations are set aside for our future work.

The rest of the paper is organised as follows. Related work is discussed in Section 2. We present the details of our framework in Section 3. Simulation results are discussed in Section 4. Finally, Section 5 concludes the paper.

¹<http://news.yahoo.com/10-fake-photos-hurricane-sandy-075500934.html>

2 Related work

To the best of our knowledge, the issue of reputation in social participatory sensing hasn't been addressed in prior work. As such, we discuss about related research focussing on reputation issues in participatory sensing systems, and in other related platforms.

Reputation systems have long been used in diverse range of disciplines and applications. In online communities such as eBay² and Amazon,³ reputation scores are mainly built on community members' feedback about workers' activities in the system. Sometimes, this feedback is explicit as in eBay, that is, community members explicitly cast feedback on a worker's quality or contributions by, for instance, rating or ranking the content the worker has created. In other cases, feedback is cast implicitly, as in Wikipedia, when subsequent editors preserve the changes a particular worker has made [10].

Reputation systems have also been widely used in ad hoc wireless networks. Most of the works presented in this area [11] attempt to address the selfish routing problem by identifying misbehaving nodes. A Monitoring component observes the packet forwarding behaviour of nodes and the local trust ratings of the neighbor nodes are updated accordingly. A reputation score is then built upon such local ratings for each node. The paths are then evaluated according to the reputation score of nodes along the path and discarded if include misbehaving nodes. The concept of reputation management has also been studied in peer to peer (P2P) networks [12]. The reputation of a peer is computed based on the opinion of its direct transaction partners as well as some third-party peers. In fact, the reputation management system utilizes information such as community-based feedbacks about past experiences of peers to help making recommendation and judgment on quality and reliability of the transactions.

The problem of data trustworthiness in participatory sensing system has also been investigated. In a series of work such as [13, 14], a Trusted Platform Module (TPM) [15] is used, which is a micro-controller embedded in the mobile device and provides it with hardware-based cryptography as well as secure storage for sensitive credentials. However, TPM chips are yet to be widely adopted in mobile devices. A group of related work, on the other hand, do not rely on TPM. Ganerwal et al. [16] proposes a reputation-based framework which makes use of Beta reputation [17] to assign a reputation score to each sensor node in a wireless sensor network. A reputation framework for participatory sensing was proposed in Huang et al. [18], which monitors the short-term behaviour of participants

to assign them a reputation score by utilizing Gompertz function [19]. They have extended their work in Huang et al. [20] and proposed a reputation anonymization scheme which is aimed at preventing the privacy leakage due to the inherent relationship between reputation information. Christin et al. [21] also proposed an anonymity-preserving reputation scheme which utilizes blind signature to provide a secure transfer of reputation scores between pseudonyms. Wang et al. [22] also addresses the problem of trust without identity by proposing a framework to compute the trustworthiness of sensing reports based on anonymous user reputation levels. However, none of these works have considered participants' social accountability. As such, their system cannot be readily used in our context.

3 Fuzzy trust framework

In this section, we explain the proposed framework for evaluating trust and reputation in social participatory sensing systems. An overview of the architecture is presented in Section 3.1 followed by a detailed discussion of each component in Section 3.3.

3.1 Framework architecture

In order to describe the performance of our proposed framework, we first begin with an illustrative example. We assume of a social network member, who serves as the requester. The requester decides to leverage his social circle to perform a sensing task, e.g., taking geotagged photos of price tags of specific products, when they are out for their weekly shopping. When deciding on whether to rely on the received contributions, the requester would naturally take into account two aspects: (i) his personal trust perception of the participant, which would depend on various aspects such as the nature of friendship (close vs. distant), participant's awareness of task requirements, participant's location, etc and (ii) the quality of contribution. In other words, the requester in his mind computes a trust rating for each contribution based on these two aspects. Our proposed framework provides a means to obtain such trust ratings by mimicking an approach similar to the requester's perception of trustworthiness in a scalable and automated manner. This trust rating helps him to select trustable contributions. Moreover, the trust server provides a reputation score for each of the participating friends, according to the trustworthiness of their successive contributions.

Figure 1 illustrates the architecture of the proposed reputation framework. The social network serves as the underlying publish-subscribe substrate for recruiting friends as participants. In fact, the basic participatory sensing procedures

²<http://www.ebay.com/>

³<http://www.amazon.com/>

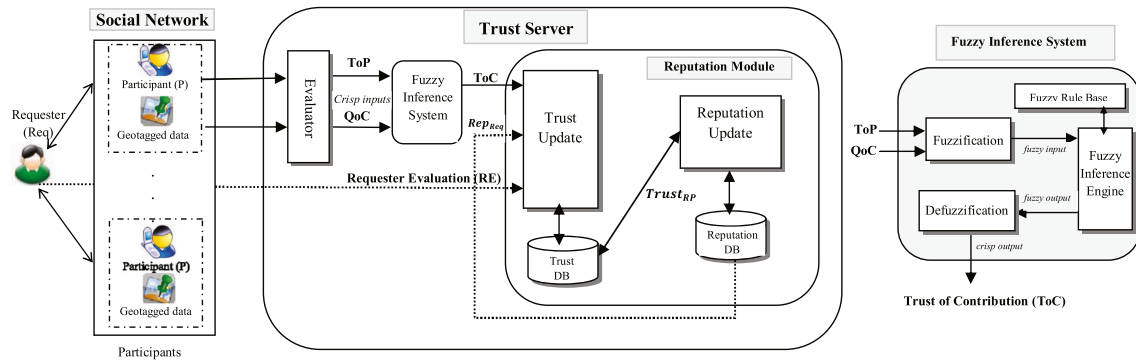


Fig. 1 Trust framework architecture

(i.e., task distribution and uploading contributions) are performed by utilizing the social network communication primitives. A person wishing to start a participatory sensing campaign acts as a requester and disseminates the tasks to his friends via email, private message or by writing as a post on their profiles (e.g., Facebook wall). Friends transfer their contributions via email or in the form of a message. We can also benefit from group construction facilities in Facebook or community memberships in Google Plus. The contributions received in response to a campaign are transferred (e.g., by using Facebook Graph API⁴) to a third party trust server, which incorporates the proposed fuzzy inference system and arrives at an objective trust rating for each contribution. This trust rating is used as a criterion to accept the contribution or revoke it, by comparing against a predefined threshold.

At the end of each campaign, a cumulative objective trust rating, referred to as $Trust_{RP}$ is automatically updated for each participant, which denotes the trustworthiness degree of Requester upon the Participant. $Trust_{RP}$ is dependent on the trustworthiness of the contribution that the participant has prepared for the requester.

For certain campaigns, depending on the nature of task, the requester may desire to add a subjective evaluation in order to indicate how much the contribution is compatible with his needs and expectations. In such a case, this subjective rating is combined with the system-computed rating to update $Trust_{RP}$.

At regular intervals, a reputation score is also calculated for each participant. The reputation score of each node depends on (i) the trust ratings that other nodes has assigned to him, and (ii) the reputation of those nodes. The reputation score is further used as a weight for participant’s evaluations, ratings or reviews. More details about trust update, subjective rating and reputation calculation are presented in Section 3.3.3.

⁴<http://developers.facebook.com/docs/reference/api/>

3.2 Definition of trust and reputation

Since trust and reputation are sometimes used interchangeably, we present a formal definition for these concepts.

Definition 1 We use the term “trust” to represent the level of confidence about the reliability of a participant. In other words, trust is a pair-wise concept, which defines the belief of one node in the reliability, truth, ability, or strength of another node.

Definition 2 The reputation of a participant, denoted as ρ , is the overall quality or character as seen or judged by people in general. In other words, reputation is a community-wide opinion generally held about someone.

3.3 Framework components

This section provides a detailed explanation of the framework components. In particular we focus on the trust sever, fuzzy inference system and reputation module.

3.3.1 Trust server

The trust server is responsible for maintaining and evaluating a comprehensive trust rating for each contribution and calculating a reputation score for each participant. As discussed in Section 1, there are two aspects that need to be considered: (1) Quality of Contribution (QoC) and (2) Trust of Participant (ToP). The server maintains a trust database, which contains the required information about participants and the history of their past contributions. When a contribution is received by the trust server, the effective parameters that contribute to the two aforementioned components are evaluated by the Evaluator and then combined to arrive at a single quantitative value in the range of [0,1] for each. The two measures serve as inputs for the fuzzy inference system, which computes the trustworthiness of contribution.

In the following, we present a brief discussion about the underlying parameters and the evaluation methods.

Quality of Contribution (QoC) In participatory sensing, contributions can be of any modality such as images or sounds. The quality of the data is affected not only by fidelity of the embedded sensor but also the sensing action initiated by the participant. The in-built sensors in mobile devices can vary significantly in precision. Moreover, they may not be correctly calibrated or even worse not functioning correctly, thus providing erroneous data. Participants may also use the sensors improperly while collecting data, (e.g., not focussing on the target when capturing images). Moreover, human-as-sensor applications such as weather radar in Demirbas et al. [9] are exposed to variability in the data quality due to subjectivity. For example, while one person may consider such weather as hot, it may be comfortable for another. In order to quantify QoC, a group of parameters must be evaluated such as: relevance to the campaign (e.g., groceries in the above example), ability in determining a particular feature (e.g., price tag), fulfilment of task requirements (e.g., specified diet restrictions), etc. There already exists research that has proposed methods for evaluating the quality of data in participatory sensing. Examples include image processing algorithms proposed in Reddy et al. [2] and outlier detection [23] for sound-based sensing tasks. Rather than reinventing the wheel, our system relies on these state-of-the-art methods for determining the QoC. QoC is in the range of [0,1].

Trust of participant (ToP) ToP is a combination of personal and social factors. Personal factors consist of the following parameters:

Expertise(E): It is defined as the measure of a participant's knowledge and is particularly important in tasks that require domain expertise. Greater credence is placed in contributions made by a participant who has expertise in the campaign. We propose to use expert finding systems for evaluating expertise. These systems employ social networks analysis and natural language processing (text mining, text classification, and semantic text similarity methods) to analyse explicit information such as public profile data and group memberships as well as implicit information such as textual posts to extract user interests and fields of expertise [24]. In particular, Dmoz⁵ open directory project is used for expertise classification. Expertise evaluation is done by incorporating text similarity analysis to find a match between the task keywords (e.g., vegetarian) and participant's expertise. We assume

that the set TE contains the Task's required Expertise and PE is the set of Participant's Expertises. In this case, the expertise score of each participant is defined as Eq. 1:

$$E = \frac{n(TE \cap PE)}{n(TE)} \quad (1)$$

where $n(A)$ is the number of elements in set A.

Timeliness(T): Timeliness measures how promptly a participant performs prescribed tasks. It depends on the contribution response time (t) and the task deadline (d). To evaluate this parameter, inverse Gompertz function defined as $T(t) = 1 - e^{-be^{-ct}}$ is used because of its compatibility with timeliness evolution: timeliness score is highest when the contribution is received immediately after the task release time. The score begins to decrease as the response time increases, reaching the minimum value when the response is received just before the deadline. In the original inverse Gompertz function, the lower asymptote is zero; it means that the curve approaches to zero in infinity. In our case, timeliness rate will only be zero if contribution is received after the deadline; otherwise, a value between x and 1 is assigned to it. It means that the lowest timeliness rating will be x if contribution is received before the deadline, and is zero if received after the deadline. So, we modify the function as Eq. 2 to (T):

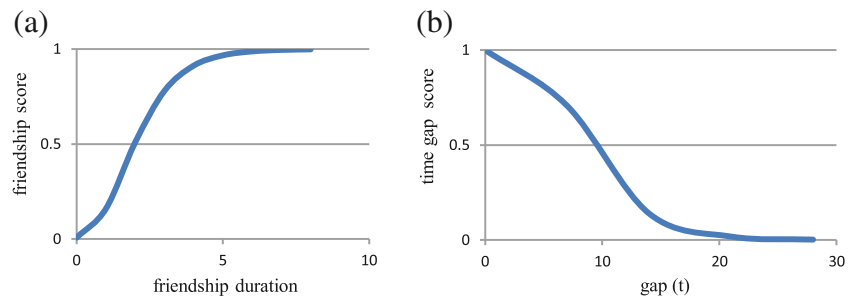
$$T(t) = \begin{cases} 1 - \left[(1-x)e^{-be^{-ct}} \right] & \text{if } t < d \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Locality(L): Another significant parameter is locality, which is a measure of the participant's familiarity with the region where the task is to be performed. We argue that contributions received from people with high locality to the tasking region are more trustable than those received from participants who are not local, since the first group is more acquainted with and has better understanding of that region. According to the experimental results presented in Alt et al. [25], people tend to perform tasks that are near to their home or work place (places that they are considered 'local' to them). This implies that if we log the location of participants' contributions, we can estimate their locality. A participant's locality would be highest at locations from where they make maximum number of contributions. In order to evaluate locality, we assume that the sensing area has been divided to n regions, and a vector V with the length equal to n is defined for each participant, where, $V(i)$ is number of samples collected in region i . In this case, locality of a participant to region i is calculated by Eq. 3:

$$L(i) = \frac{V(i)}{\sum_{i=0}^{n-1} V(i)} \quad (3)$$

⁵<http://www.dmoz.org>

Fig. 2 **a** Gompertz function for friendship score. **b** Inverse Gompertz function for time gap score



Next, we explain the social factors that affect ToP:

Friendship duration (F): In real as well as virtual communications, long lasting friendship relations normally translate to greater trust between two friends. So, friendship duration which is an estimation of friendship length is a prominent parameter in trust development. We use the Gompertz function depicted in Fig. 2a to quantify friendship duration, since its shape is a perfect match for how friendships evolve. Slow growth at start resembles the friendship gestation stage. This is followed by a period of accumulation where the relationship strengthens culminating in a steady stage. As such, the friendship duration is evaluated according to Eq. 4, in which, b and c are system-defined constants and t is the time in years.

$$F(t) = e^{-be^{-ct}} \tag{4}$$

Interaction time gap (I): In every friendship relation, interactions happen in form of sending requests and receiving responses. Interaction time gap, measures the time between the consequent interactions and is a good indicator of the strength of friendship ties. If two individuals interact frequently, then it implies that they share a strong relationship, which translates to greater trust. We propose to use the inverse Gompertz function depicted in Fig. 2b to quantify the interaction time gap, since a smaller time gap indicates stronger relationship, which leads to high social trust and vice-versa. So, the interaction time gap is evaluated according to Eq. 5, in which, b and c are system-defined constants and t is the gap (in days) between the current time and the Latest Interaction (LI) time.

$$I(t) = 1 - e^{-be^{-ct}} \tag{5}$$

The aforementioned parameters are combined by the Evaluator to arrive at a single value for ToP, as depicted in Eq. 6,

$$ToP = w_1 \times E + w_2 \times T + w_3 \times L + w_4 \times F + w_5 \times I \tag{6}$$

where w_i is the weight of each parameter, and $\sum_{i=1}^5 (w_i)$ equals to 1. The adjustment of the weights depends on the

nature of the task. For example, in location-based tasks, w_3 is set to be considerably high to give more impression to Locality parameter. Similarly, for tasks where real-time information is important, a higher weight may be associated with Timeliness (w_2). ToP is in the range of $[0,1]$.

3.3.2 Fuzzy inference system

Our proposed framework employs fuzzy logic to calculate a comprehensive trust rating for each contribution, referred to as the Trust of Contribution (ToC). We cover all possible combinations of trust aspects and address them by leveraging fuzzy logic in mimicking the human decision-making process. The inputs to the fuzzy inference system are the crisp values of QoC and ToP. In the following, we describe the fuzzy inference system components.

Fuzzifier: The fuzzifier converts the crisp values of input parameters into a linguistic variable according to their membership functions. In other words, it determines the degree to which these inputs belong to each of the corresponding fuzzy sets. The fuzzy sets for QoC, ToP and ToC are defined as:

$$\begin{aligned} T(QoC) &= T(ToP) = \{Low, Med1, Med2, High\} \\ T(ToC) &= \{VL, L, M, H, VH\}. \end{aligned}$$

For any set X , a membership function on X is any function from X to the real unit interval $[0,1]$. The membership function which represents a fuzzy set A is usually denoted by μ_A . The membership degree $\mu_A(x)$ quantifies the grade of membership of the element x to the fuzzy set A . The value 0 means that x is not a member of the fuzzy set; the value 1 means that x is fully a member of the fuzzy set. The values between 0 and 1 characterize fuzzy members, which belong to the fuzzy set only partially.

Figure 3a represents the membership function of QoC and ToP and Fig. 3b depicts the ToC membership function. We used trapezoidal shaped membership functions since they provide adequate representation of the expert

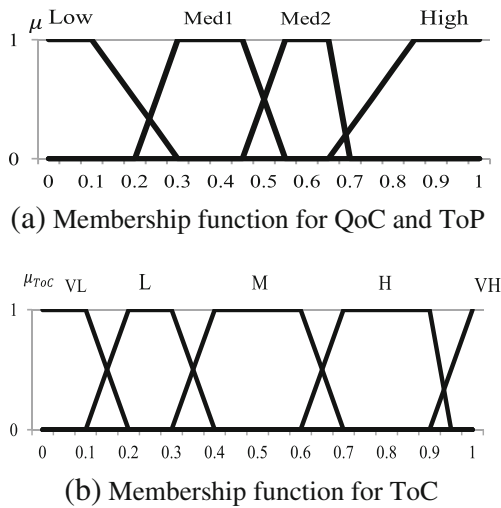


Fig. 3 Membership functions of input and output linguistic variables

knowledge, and at the same time, significantly simplify the process of computation.

Inference Engine: The role of inference engine is to convert fuzzy inputs (QoC and ToP) to the fuzzy output (ToC) by leveraging If-Then type fuzzy rules. The combination of the above mentioned fuzzy sets create $4 \times 4 = 16$ different states which have been addressed by 16 fuzzy rules as shown in Table 1. Fuzzy rules help in describing how we balance the various trust aspects. The rule base design has been done *manually*, based on the experience and beliefs on how the system should work. For example, as rule no. 1 denotes, we manually set ToC with very low (VL) value for those contributions with Low ToC and ToP, so that it results in the revocation of these contributions. To define the output zone, we used *max-min* composition method as:

$$\mu_{T(ToC)}(ToC) = \max_{X \in T(ToP), Y \in T(QoC)} [\min(\mu_X(ToP), \mu_Y(QoC))].$$

The result of the inference engine is the ToC which is a linguistic fuzzy value.

Table 1 Fuzzy rule base for defining ToC according to QoC and ToP

Rule no.	if QoC	and ToP	Then ToC	Rule no.	if QoC	and ToP	Then ToC
1	Low	Low	VL	9	Med2	Low	M
2	Low	Med1	L	10	Med2	Med1	H
3	Low	Med2	L	11	Med2	Med2	H
4	Low	High	M	12	Med2	High	H
5	Med1	Low	L	14	High	Low	H
6	Med1	Med1	L	14	High	Med1	H
7	Med1	Med2	M	15	High	Med2	VH
8	Med1	High	M	16	High	High	VH

Defuzzifier: A defuzzifier converts the ToC fuzzy value to a crisp value in the range of [0, 1]. We employed the Centre of Gravity (COG) [26] defuzzification method, which computes the center of gravity of the area under ToC membership function. COG is perhaps the most commonly used and popular defuzzification technique with the advantage of quick and highly accurate computations.

3.3.3 Reputation module

Once the ToC is defined for a contribution, the corresponding requester-participant mutual trust is updated, which is then used to calculate/update the participant’s reputation score. In the following, we describe these steps in details:

As mentioned before, depending on the nature of the task, the requester may desire to add a subjective rating to participants’ contribution. Typically, in a participatory sensing system, there are two types of tasks. Type 1 tasks are those where subjective rating is important. This is particularly relevant for campaigns where it is difficult for the requester to express his real needs, desires or restrictions via task definition. Subjective rating is also useful when the requester does not have enough knowledge about the task and needs an expert review to confirm the validity of the contributions. For example, assume a requester with a strict gluten-free diet who asks his friends to take photos from the price tag and ingredients of gluten-free products. The term gluten-free is generally used to indicate a supposedly harmless level of gluten rather than a complete absence. For those with serious celiac disease, the maximum safe level of gluten in a finished product is even lower than the amount that exists in normal gluten-free products. So, a double check with product ingredients is essential to be performed either by the requester himself or by a nutritionist to assure that it is safe to be consumed.

On the other hand, type 2 tasks such as Earphone application [4] (in which, participants are recruited to gather noise samples), do not benefit from subjective rating because the requester may not be in the best position to evaluate the

quality due to not being aware of the ground truth. In such cases, the requester gives the authority to the trust server and relies on the objective evaluation of the system, which automatically assigns a rating to the contributions by leveraging methods such as majority consensus [18].

In order to support both kinds of tasks, we denote the subjective rating as Requester Evaluation (RE) which implies the trustworthiness of contribution from the requester’s point of view. Although RE value can be in any range, in our simulation in Section 4, we assume that RE has a value in the range of $(ToC - \mu, ToC + \mu)$, where $\mu = 1 - \rho_{Req}$ and ρ_{Req} is the requester’s reputation score. For a requester with high reputation score, the value of μ is small, resulting in RE close to ToC. It means that a requester with high reputation score is likely to assign a rating, which is close to the system-computed rating.

In the absence of subjective ratings, the requester simply relies on the objective ratings assigned by the trust server. In this case, μ is simply set to zero, resulting in $RE = ToC$.

Based on the ToC assigned to each contribution, the trust of requester upon the corresponding participant ($Trust_{RP}$) is updated. In fact, we adopt a reward/penalty policy for this update. A participant with ToC values greater than a predefined threshold1(Th_1) is rewarded, and the amount of $|ToC - \rho_{Req} * RE|$ is added to $Trust_{RP}$. Similarly, a participant with ToC less than a predefined threshold2(Th_2) is penalized, and the amount of $|ToC - \rho_{Req} * RE|$ is reduced from $Trust_{RP}$. This can be summarized in Eq. 7. In our simulations in Section 4, we set $(Th_1) = 0.7$ and $(Th_2) = 0.3$.

$$Trust_{RP} = \begin{cases} Trust_{RP} + |ToC - \rho_{Req} * RE| & \text{if } ToC > Th_1 \\ Trust_{RP} - |ToC - \rho_{Req} * RE| & \text{if } ToC < Th_2 \end{cases} \tag{7}$$

Note that in this equation, we use the requester’s reputation score (ρ_{Req}) as a weight for his evaluation (RE), since we believe an evaluation from a requester with high reputation score is more trustworthy than an evaluation from a low reputable requester.

This process is repeated for all participants at the end of each sensing campaign, and $Trust_{RP}$ is updated for all of them.

After every n campaigns, $Trust_{RP}$ values upon each active participant act as inputs for reputation module, which updates the participant’s reputation score accordingly.

While there are already different crowdsourcing applications of online reputation systems [27] such as eBay, Epinions⁶ and Amazon, we use Web Page ranking algorithms as the basis for computing reputation scores. We draw parallels between the rank of a page in a set of web

pages and the reputation score of a member in a social network. Moreover, the weights of links from different pages to a specific page are considered to be equivalent to the trust ratings of one member as determined by the other members of the social network.

Having a set of objects, a ranking algorithm calculates a relative importance of all objects in the set and makes an ordered list according to the importance. Web page ranking algorithms such as PageRank [28] calculate and assign a rank to a web page by analysing the web graph. Roughly speaking, PageRank ranks a page according to how many other pages are pointing at it. This can be described as a reputation system, because the collection of hyperlinks to a given page can be seen as public information that can be combined to derive a reputation score. A single hyperlink to a given web page can be seen as a trust rating of that web page.

In PageRank, the rank of page P , denoted by $\rho(P)$ is defined as: $\rho(P) = \frac{\sum_{P_i \rightarrow P} (\rho(P_i))}{L(P)}$ in which, P_i is the set of all pages which have an outgoing link to page P , and $L(P_i)$ is the number of outgoing links from page P_i .

In the original PageRank algorithm, it is assumed that all the outgoing links have equal weights. This is not always true, since not all outgoing links from a web page are equally important. So, we adopted the extension offered in Kamvar et al. [29] which modifies the above equation as Eq. 8,

$$\rho(P) = \sum_{P_i \rightarrow P} \frac{w_i}{\sum_{P_i \rightarrow P_j} w_j} \rho(P_i) \tag{8}$$

in which, w_i is the weight of the outgoing link, and the sum of weights of outgoing links is equal to 1.

We explain this further by presenting an illustrative example. Consider the graph in Fig. 4 in which, P_1, P_2, P_3 and P_4 are the social network members. Links represent

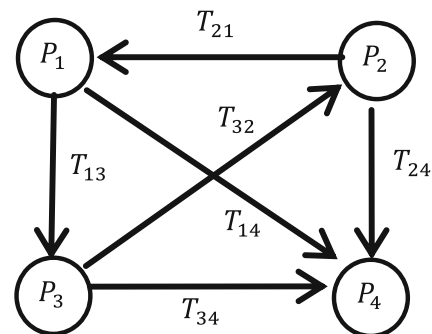


Fig. 4 A sample social graph of 4 members with mutual trust ratings

⁶<http://www.epinions.com/>

friendship relations with weights equal to the mutual trust between the pairs. In this case, according to Eq. 8:

$$\begin{aligned}\rho(P_1) &= T_{21} \times \rho(P_2) \\ \rho(P_2) &= T_{32} \times \rho(P_3) \\ \rho(P_3) &= T_{13} \times \rho(P_1) \\ \rho(P_4) &= T_{14} \times \rho(P_1) + T_{24} \times \rho(P_2) + T_{34} \times \rho(P_3)\end{aligned}$$

As can be seen in the above expressions, reputation calculation is an iterative process and continues until convergence is obtained. In our simulation in Section 4, we assume that the convergence occurs when $|\rho_k(P_i) - \rho_{k-1}(P_i)| \leq 10^{-10}$ for all P_i .

It should be noted that the reputation score calculation is independent of the campaign specification. As shown in Eq. 8, the reputation score of each participant depends on the trust rating of requester upon the participant ($Trust_{RP}$), which itself, according to Eq. 7, depends on two factors: (i) trustworthiness of contribution (ToC) and (ii) requester's evaluation (RE). All these parameters are related to the contribution itself and do not depend on the campaign.

Regarding the reputation update interval, one may argue that updating the reputation score after every campaign will allow the system to better reflect the behavioural changes of participants. However, as mentioned in Section 3.3.3, the reputation module utilises PageRank algorithm to calculate and update reputation scores. The PageRank algorithm is a recursive algorithm wherein, the recursion continues until convergence happens. This can be quite time consuming, especially if the number of participants is large, which is typically the case in social networks. In other words, there is a trade off between accurately reflecting the behavioural changes of the participants and the associated computational complexity of updating the reputation scores. The update period n , thus can be a system parameter that can be configured by the system designer.

To summarize, once a campaign is launched, participants begin to send a series of contributions. For each contribution, the Evaluator computes a value for QoC and ToP. These values are fed to fuzzy inference engine which calculates ToC for that contribution. The trust of requester upon each participant ($Trust_{RP}$) is updated according to his ToC. The server utilizes $Trust_{RP}$ and ρ_{Req} to update the reputation score of each participant.

4 Experimental evaluation

This section presents simulation-based evaluation of the proposed trust system. The simulation setup is outlined in Section 4.1 and the results are in Section 4.2.

4.1 Simulation setup

To undertake the preliminary evaluations outlined herein, we chose to conduct simulations, since real experiments in social participatory sensing are difficult to organise. Simulations afford a controlled environment where we can carefully vary certain parameters and observe the impact on the system performance. We developed a custom Java simulator for this purpose.

In order to investigate the performance of our proposed framework under heavy load conditions and obtain stable and reliable results, we assumed high connectivity of social graph and large number of campaigns. In other words, we simulate an online social network where 100 members participate in 5000 campaigns, producing one contribution for each. We assume that each member is connected to all others, similar to a social community. So, there are totally 10000 friendship relations. All members can serve both as requesters who launch sensing campaigns and as participants who contribute data to these sensing campaigns.

In our previous work [30], we assumed of categorizing participants according to the trade-offs between ToP and QoC. We wanted to observe how accurate the system assigns trust ratings to contributions in case of different ToP and QoC levels. Moreover, we artificially created scenarios where participants begin producing contributions with low QoC, which results in a decrease in ToC. We wanted to see if the system is able to quickly detect this transition and revoke low trustable contributions in an accurate and robust manner.

In this paper, instead of categorizing the participants according to ToP and QoC, we designed the categories according to the trade-offs between personal factors and social factors inside ToP, and simply assumed that QoC has a value in the range of $(ToP - \mu, ToP + \mu)$. In fact, we are going to observe how the system reacts to behavioural changes of participants and how much it is successful to update the reputation scores in case of such fluctuations. As mentioned in Section 3.3, ToP parameters can be divided into two groups: social factors which include Friendship duration and Interaction time gap, and personal factors which include Timeliness, Expertise and Locality. In the real-world, there are often situations where a friend with a high rating of social factors (i.e., a very close friend with whom one has repeated interactions) has a low rating for personal factors for a period of time (i.e., does not have related expertise or does not produce timely contributions). In other words, we may have participants who have high social trust, but low personal trust, and vice versa. We have thus 4 different states based on the combination of different levels of personal and social trusts.

Specifically, we assumed that 60 members (out of 100) belong to Category A whereas the remaining 40 belong to Category B, adding the assumption that category A members have high personal trust, while category B members have low personal trust. We also assume that for each member P_A in category A, all other members score P_A with high social trust, and for each member P_B in category B, all other members score P_B with low social trust.

When P_A serves as requester, other members form two subcategories:

- A-1: which includes 59 members from category A, excluding P_A . They have high personal trust and score P_A with high social trust.
- A-2: which includes 40 members from category B. They have low personal trust and score P_A with high social trust.

Similarly, when P_B serves as requester, other members form two subcategories:

- B-1: which includes 60 members from category A. They have high personal trust and score P_B with low social trust.
- B-2: which includes 39 workers from category B, excluding P_B . They have low personal trust and score P_B with low social trust.

It is but natural that not all friends in a social network would contribute data to sensing campaigns. As such, we assume that 10 % of the members in category A and 50 % of the members in category B do not upload any data. The rationale for assuming unequal percentages is that the first group are close friends and hence a higher percentage would be willing to contribute, whereas the second group are not so and have less willingness to contribute.

Whenever a task is launched, one of the participants is selected to be the requester. Without loss of generality we assume that tasks are launched in sequential order by the social network members, i.e., member 1 launches the first campaign, member 2 launches the second campaign and so on.

ToP parameter settings In the following, we will discuss the initialisation of the various parameters introduced in Section 3.3. For a set of parameters, the assumptions and parameter settings are based on the statistics and results that have been proposed in other experiments. For the rest, we set the parameters in a way that allows us to configure user groups with different behavioural traits.

In order to set the Expertise value for a participant, we assume that there are a total of 6 expertise areas defined and that each task needs at most 3 expertise areas ($n(TE) = 3$). To calculate the Expertise score for each participant,

we assign a value to $n(PE)$ based on his category, as shown in Table 2. The expertise score E is then calculated using (1).

For Timeliness, we first set the response time (rt) for each participant. In order to initialise the response time (rt), we used the statistics presented in Alt et al. [25]. In this paper, the authors performed a real participatory sensing experiment and found that with the deadline of 1 day, 40 % of the tasks were solved within the first 3 hours, 70 % within 15 hours, and 90 % within 20 hours. We have used the general trend from their observations to set the timeliness parameter in our simulation settings, where the deadline is 1 week. As seen in Table 2 (Section 4.1, page 11), for a participant P_A belonging to category A, with probability of 0.4, rt is at most one day, with the *probability* = 0.65, rt is at most half of a week, and with the probability of 0.9, rt is at most one week (Note that the greatest probability is 0.9, since with the probability of 0.1 (10 %), P_A does not attend in sensing campaign). rt then acts as the input value for Eq. 2 which results in Timeliness score T for participant. Other input parameters for Eq. 2 have been set as $x = 0.3$, $b = 6$, $c = 0.6$, and $d = 7days$.

For Locality, we assume that there are a total of 25 regions and that each participant is local to 3 regions (i.e., Locality score L for these 3 regions is 1). We also assume that when a participant has the maximum Locality score to a region, he has a relatively high locality to its surrounding regions. So, Locality score L is assigned to the surrounding regions up to 3 levels of neighborhood, i.e., N_1 , N_2 and N_3 , based on participant's category, as shown in Table 2.

For Friendship duration, as mentioned in Section 3.3, the input parameter (t) is the time (in years) passed from the beginning of friendship establishment. The initial value of t is set according to the participant's category, as shown in Table 2 and a constant value of 0.02 is added to t upon each participation. t is then serves as the input value for Eq. 4 which computes the Friendship duration score F for the participant. Other input parameters for Eq. 4 have been set as $b = 5$ and $c = 1$.

Finally, for the Interaction time gap, as mentioned in Section 3.3, the input parameter t is the gap (in days) between the current time and the Latest Interaction(LI) time. We set LI based on the category of each participant, as shown in Table 2, and calculate t accordingly. t is then fed to Eq. 5 which calculates the Interaction time gap score I for the participant. Other input parameters for Eq. 5 have been set as $b = 10$ and $c = 0.2$.

Once all of the aforementioned parameters are computed, ToP is calculated by simply averaging them. In other words, we simply assume that $w_i = 1/5$ in Eq. 6. QoC is then assigned a value in a range of $(ToP - \mu, ToP + \mu)$ with $\mu = 0.1$.

Table 2 ToP parameter settings

Category A		Category B	
param	value	param	value
n(PE)	4	n(PE)	2
rt	$\begin{cases} (0, 1] & \text{prob} = 0.4 \\ (1, 7/2] & \text{prob} = 0.65 \\ (7/2, 7] & \text{prob} = 0.9 \end{cases}$	rt	$\begin{cases} (0, 1] & \text{prob} = 0.1 \\ (1, 7/2] & \text{prob} = 0.3 \\ (7/2, 7] & \text{prob} = 0.5 \end{cases}$
N_1	random(0,1)	N_1	random(0,0.5)
N_2	random(0,0.9)	N_2	0
N_3	random(0,0.8)	N_3	0
t	rand[4,5]	t	rand[0,1]
LI	$\begin{cases} (0, d] & \text{prob} = 0.8 \\ 0 & \text{prob} = 0.2 \end{cases}$	LI	$\begin{cases} (0, d] & \text{prob} = 0.2 \\ 0 & \text{prob} = 0.8 \end{cases}$

ToC is then calculated and $Trust_{RP}$ is updated according to Eq. 7. At intervals, reputation score is also updated for participants. We set the reputation interval to be after every 100 campaigns ($n = 100$).

In the first scenario, we assume that ToPs follow the category settings throughout the entire simulation. In the second scenario, we assume that ToP parameters change for a group of participants which results in a transition from one category to another (details in Section 4.2).

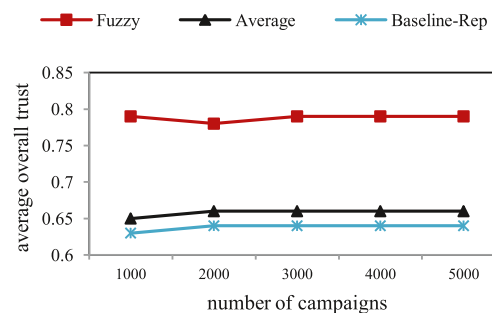
As mentioned in Section 3.1, a ToC rating is calculated for each contribution and those with ToC lower than a predefined threshold are revoked from further calculations. The ToCs for the non-revoked contributions are then combined to form an overall trust for that campaign. In other words, $OverallTrust = \frac{\sum_{i=1}^n ToC}{n}$ in which, n is the number of non-revoked contributions. The revocation threshold is set to 0.5. We consider the overall trust as the evaluation metric. The greater the overall trust the better the ability of the system to revoke untrusted contributions. Overall trust has a value in the range of [0, 1]. We also calculate the reputation scores for all participants to see whether they reflect the behaviour of participants in normal and transition settings. Reputation score value is a number in the range of [0, 1] with initial value of 0.5 for each participant.

We compare the performance of our framework against the following systems: (1) Baseline-Rep: which follows the approach in Huang et al. [18] by calculating a reputation score for each participant according to the QoC of his successive contributions. This reputation score is used as a weight for QoC. In other words, $ToC = \sqrt{Rep * QoC}$ (2) Average: which includes ToP but computes the ToC simply as an average of ToP and QoC (3) Fuzzy: our proposed framework.

4.2 Simulation results

We first present the simulation results for the first scenario. Figure 5 depicts the evolution of the average overall trust as a function of the number of campaigns. As shown in the figure, our fuzzy trust method outperforms the other methods. This confirms its success in mimicking the human trust establishing process by correct settings of fuzzy rules. In particular, we have set the rules in a way that results in early detection and severe punishment of untrusted contributions and also put greater emphasis on highly trusted contributions. The former has been done by assigning a very low (VL) value to ToC in case of low ToP and QoC (i.e., Rule no. 1 in Table 1), whereas the latter has been obtained through assigning very high (VH) value to ToC in case of high QoC and above average ToP (i.e., Rule no. 15 and 16 in Table 1).

Figure 6 depicts the evolution of overall trust for 1000 contributions with Fuzzy method. As can be seen in this figure, at each interval containing 100 contributions, two different levels of overall trust are achieved. Remembering the order of requesters which is equal to members' order, higher level of overall trust is obtained when the requester is from category A. So, participants are located either in subcategory A-1 or A-2. This will result in either high ToC values (when participants are from category A-1) or

**Fig. 5** Evolution of average overall trust for all methods, Scenario 1

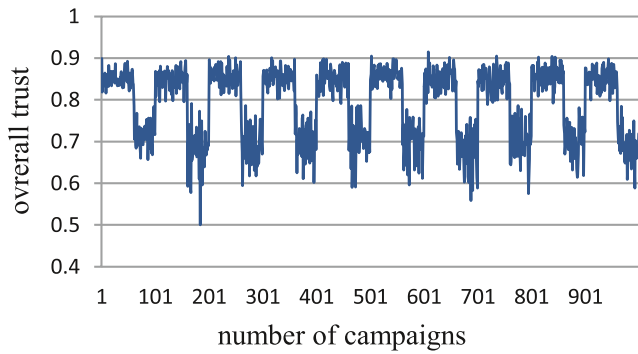


Fig. 6 Evolution of overall trust, Fuzzy method, Scenario 1

medium ToC values (when participants are from category A-2), which in turn, results in high overall trust. Similarly, lower level of overall trust is obtained when the requester is from category B. So workers are located either in category B-1 or B-2. This will lead to either medium ToC values (when participants are from category B-1) or low ToC values (when participants are from category B-2), which results in low overall trust. This variation is repeated regularly at each interval of 100 contributions.

Figure 7 presents the reputation of 100 participants after attending in 5000 sensing campaigns. As mentioned before, the initial value of reputation score for all participants is 0.5. Category A participants who have high ToPs, produce contributions with high ToC and hence, they get rewarded. This reward results in $trust_{RP}$ increase upon them, which in turn, increases their reputation score. On the contrary, for category B participants with low ToPs, ToCs will also be low, and hence, they are penalized, which results in the reduction of their reputation score. To summarize, our system continually tracks the contributions made over a series of campaigns and detects participants' behaviour, which is accurately reflected in the evolution of the reputation scores.

Next, we present results for the second scenario, wherein, the behaviour of participants change for a period of time,

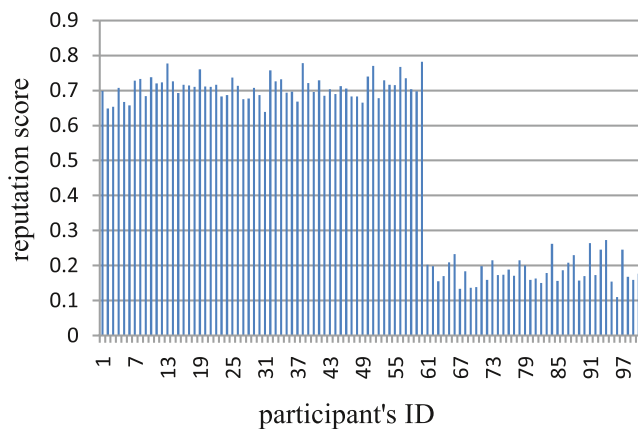


Fig. 7 Reputation score for all members, Fuzzy method, Scenario 1

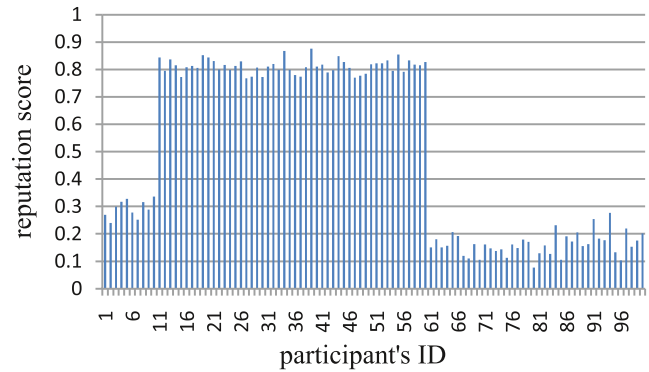


Fig. 8 Reputation score for all members at Campaign 4000th, Fuzzy method, Scenario 2

which results in a transition from one category to another. This scenario allows us to observe the performance of the schemes in the presence of noise. For example, consider a participant P_A who is in category A, changes his behaviour for a period of time and behaves in a different manner which results in decrease of his personal and (hence) social trust. For example P_A no longer provides timely contributions or does not care enough about the requirements of the task. This behavioural change results in a decrease in his personal trust, and consequently, others score him low with social trust. In other words, a participant may encounter a transition from category A to category B. In this scenario, we assume that 10 from 60 participants of category A transition to category B (e.g., a reduction in their personal and social factor values is created) in the period between 1000th and 4000th campaigns.

Figure 8 shows the reputation score of 100 participants at the end of transition period (i.e., after attending in 4000 campaigns). As can be seen in this figure, the reputation of first ten participants who encounter such transition has a considerable decrease in comparison with others not encountering such transition. This again demonstrate the

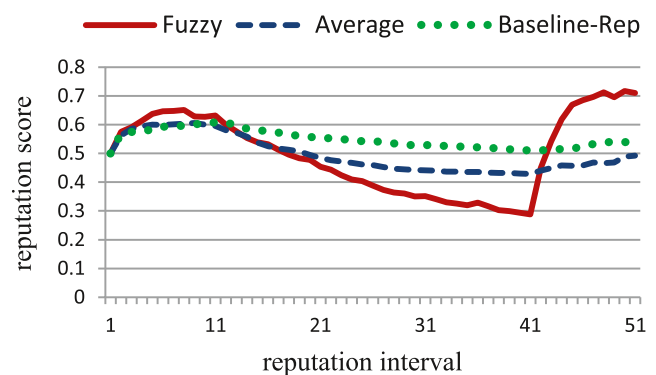


Fig. 9 Evolution of reputation score for participant no. 9 in all methods, Scenario 2

ability of our reputation module to adjust the reputation scores as a reflection of behavioural changes of participants.

Finally, Fig. 9 shows the reputation score evolution of participant no. 9 encountering such transition between 10th and 40th reputation intervals (between 1000th and 4000th campaigns). As can be observed, our proposed method shows an explicit and considerable reaction to this behavioural change, as compared with other methods. There is a decrease in reputation score due to dishonest behaviour during the transition period. At the end of transition period, transition encountered participant resumes his normal behaviour which results in a considerable increase in his reputation score.

5 Conclusion an future work

In this paper, we proposed an application agnostic reputation framework for social participatory sensing system. Our system independently assesses the quality of the data and the trustworthiness of the participants and combines these metrics using fuzzy inference engine to arrive at a comprehensive trust rating for each contribution. The system is then assigns a reputation score to participants by leveraging the concepts utilised in PageRank algorithm. Simulations demonstrated that our scheme increases the overall trust by over 15 % as compared to other methods, and assigns reputation scores to participants. In this paper, the focus is on trust management for social participatory sensing systems. As our future work, we are going to consider security/privacy attacks and propose a privacy preserving framework for social participatory sensing systems to address these attacks.

References

- Burke J, Estrin D et al (2006) Participatory sensing. In: WSW workshop, ACM SenSys'06
- Reddy S., Parker A et al (2007) Image browsing, processing, and clustering for participatory sensing: lessons from a dietsense prototype. In: ACM EmNets'07
- Dong Y, Kanhere SS, Chou CT, Liu RP (2011) Automatic image capturing and processing for petrolwatch. In: ICON'11
- Rana RK, Chou CT, Kanhere SS, Bulusu N, Hu W (2010) Ear-phone: an end-to-end participatory urban noise mapping. In: ACM/IEEE IPSN'10
- Reddy S, Estrin D, Srivastava M (2010) Recruitment framework for participatory sensing data collections. In: Pervasive'10
- Krontiris I, Freiling F (2010) Integrating people-centric sensing with social networks: a privacy research agenda. In: IEEE PERCOM'10
- Krontiris I, Freiling F (2010) Urban sensing through social networks: the tension between participation and privacy. In: ITWDC'10
- Camenisch J, Hohenberger S, Lysyanskaya A (2006) Balancing accountability and privacy using e-cash. In: SCN'06
- Demirbas M, Bayir MA, Akcora CG et al (2010) Crowd-sourced sensing and collaboration using twitter. In: IEEE WoWMoM'10
- Allahbakhsh M, Benatallah B, Ignjatovic A, Motahari-Nezhad HR, Bertino E, Dustdar S (2013) Quality control in crowdsourcing systems: issues and directions. *IEEE Internet Comput* 17(2):76–81
- Louta M, Kraounakis S, Michalakis A (2010) A survey on reputation-based cooperation enforcement schemes in wireless ad hoc networks. In: WINSYS'10
- Selvaraj C, Anand Sh (2012) A survey on security issues of reputation management systems for peer-to-peer networks. *Comput Sci Rev* 6(4):145–160
- Dua A, Bulusu N, Feng WC, Hu W (2009) Towards trustworthy participatory sensing. In: HotSec'09
- Gilbert P, Jung J et al (2011) YouProve: authenticity and fidelity in mobile sensing. In: ACM SenSys'11
- Trusted computing group. <https://www.trustedcomputinggroup.org/home>
- Ganeriwal S et al (2008) Reputation-based framework for high integrity sensor networks. In: ACM TOSN'08, vol 4, p 3, pp 1–37
- Commerce BE, Jsang A, Ismail R (2002) The beta reputation system. In: Bled electronic commerce conference
- Huang KL, Kanhere SS, Hu W (2011) On the need for a reputation system in mobile phone based sensing. *Ad Hoc Networks*
- Kenney J, Keeping E (1962) *Mathematics of statistics, part 1*. Van Nostrand, Princeton
- Huang KL, Kanhere SS, Hu W A Privacy-preserving reputation system for participatory sensing. *IEEE LCN'12*
- Christin D, Kanhere SS et al (2013) IncogniSense: an anonymity-preserving reputation framework for participatory sensing applications. In: *Pervasive and mobile Computing'13*
- Wang X et al (2013) ARTSense: anonymous reputation and trust in participatory sensing. In: *IEEE Infocom'13*
- Papadimitriou S, Kitagawa H, Gibbons P, Faloutsos C (2003) Loci: fast outlier detection using the local correlation integral. In: *IEEE ICDE'03*
- Alkouz A, Luca EWD, Albayrak S (2011) Latent semantic social graph model for expert discovery in facebook. In: *IICS'11*
- Alt F, Shirazi AS et al (2010) Location-based crowdsourcing: extending crowdsourcing to the real world. In: *ACM NordiCHI'10*
- Leekwijck W, Kerre EE (1999) Defuzzification: criteria and classification. *Fuzzy Sets Syst* 108(2):159–178
- Jøsang A, Ismail R, Boyd C (2007) A survey of trust and reputation systems for online service provision. *J Decis Support Syst* 43(2):18–644
- Brin S, Page L et al (1998) The PageRank citation ranking: bringing order to the web. Technical report, Stanford Digital Library Technologies Project
- Kamvar S, Haveliwala T et al (2003) Exploiting the block structure of the web for computing PageRank. Technical Report Stanford University
- Amintoosi H, Kanhere SS (2012) A trust framework for social participatory sensing systems, in press. In: *MobiQuitous 2012, China*