QnQ: Quality and Quantity based Unified Approach for Secure and Trustworthy Mobile Crowdsensing

Shameek Bhattacharjee[1], Nirnay Ghosh[2], Vijay K. Shah[3] and Sajal K. Das[4]

[1]Dept. of Computer Science, Western Michigan University, USA, Email: shameek.bhattacharjee@wmich.edu
[2]iTrust Research Centre, Singapore University of Technology and Design, Email: nirnay_ghosh@sutd.edu.sg
[3]Dept. of Computer Science, University of Kentucky, USA, Email: vijay.shah@uky.edu
[4]Dept. of Computer Science, Missouri Univ. of Science & Technology, USA, Email: sdas@mst.edu

***Abstract*—A major challenge in mobile crowdsensing applications is the generation of false (or spam) contributions resulting from selfish and malicious behaviors of users, or wrong perception of an event. Such false contributions induce loss of revenue owing to undue incentivization, and also affect the operational reliability of the applications. To counter these problems, we propose an event-trust and user-reputation model, called QnQ, to segregate different user classes such as honest, selfish, or malicious. The resultant user reputation scores, are based on both *'quality'* (accuracy of contribution) and *'quantity'* (degree of participation) of their contributions. Specifically, QnQ exploits a rating feedback mechanism for evaluating an event-specific expected truthfulness, which is then transformed into a robust quality of information (QoI) metric to weaken various effects of selfish and malicious user behaviors. Eventually, the QoIs of various events in which a user has participated are aggregated to compute his reputation score, which in turn is used to judiciously disburse user incentives with a goal to reduce the incentive losses of the CS application provider. Subsequently, inspired by *cumulative prospect theory (CPT)*, we propose a risk tolerance and reputation aware trustworthy decision making scheme to determine whether an event should be published or not, thus improving the operational reliability of the application. To evaluate QnQ experimentally, we consider a vehicular crowdsensing application as a proof-of-concept. We compare QoI performance achieved by our model with Jøsang's belief model, reputation scoring with Dempster-Shafer based reputation model, and operational (decision) accuracy with expected utility theory. Experimental results demonstrate that QnQ is able to better capture subtle differences in user behaviors based on both quality and quantity, reduces incentive losses and significantly improves operational accuracy in presence of rogue contributions.**

***Index Terms*—Crowdsensing, Quality of information, Trust, Reputation, Dependable Systems, Participatory Sensing**

I. Introduction

Sophistication in mobile devices (e.g., smartphones, tablets) and their widespread adoption have given rise to a novel interactive sensing paradigm, known as *Participatory Sensing* [9]. A variant of participatory sensing system involving explicit human participation for sensing the environment, is termed as *Crowdsensing (CS) [14]*. In such systems, a crowd of citizens voluntarily submit certain observations termed as *contributions* (viz., report, image, audio) about some phenomena in their immediate environment to a CS server, which in turn fuses these contributions to conclude a summarized statistic (*or information*) and publishes for the benefit of the public at large.

An important category of CS applications is vehicular traffic monitoring and management [5]. In such applications, a user's contributions are equivalent to 'reports' about various traffic conditions that they might have observed. Based on certain correlations among such reports, the CS application decides whether a certain traffic 'event' has occurred, and publishes this 'information' as a broadcast notification on a user's smartphone application. Such information help to improve driving experiences through dynamic route planning and re-routing of traffic in busy cities. Two notable examples of real vehicular CS applications include Google's *Waze* (*www.waze.com*) and *Nericell* [26]. Other practical examples include *FourSquare* and *Yelp* which help users to find best destinations in their geographical proximity for food, entertainment, and other attractions or events of interest.

The real benefit of CS paradigm is that rich, fine grained and precise sensory observations can be obtained quickly without establishing dedicated infrastructure [32]. Thus, it reduces significant infrastructure overheads incurred due to sensor deployment, management, and periodic maintenance. However, a major drawback is its "open" nature (accessible to all) which may expose CS applications to false contributions [18], [37], which result in publishing erroneous information.

Most of the CS applications depend on various incentive mechanisms to motivate the users to keep contributing regularly, and thus preserve their viability [24]. It has been noted that in most of these mechanisms, the deciding factor of incentive is the user's *degree of participation* (i.e. "quantity" or how much they contribute). However, *selfish users* may take advantage of this loophole and intermittently generate false contributions to boost their participation for gaining undue incentives [32], thereby incurring revenue losses to the CS application. Furthermore, there could be *malicious users* who attempt to cripple the CS applications by generating a large number of bogus contributions in collusion [37]. Recently, such

colluding attack was launched against Waze in Israel, by which fake traffic jam reports were created to orchestrate traffic re-routing and unnecessary roadblocks [34]. Occasionally, false contributions may also be generated owing to wrong perception. Regardless of the motive, false contributions incur revenue loss due to unnecessary disbursement of incentives and also tarnishes the operational reliability of the CS application.

In a preliminary work, we studied a real data set from Waze [5], and established that the 'quantity' rather than 'quality' of contributions decides incentives (details presented in Section II-B). Here we argue that besides the quantity, there is also a simultaneous need for assessing *quality of information* (QoI) generated from user contributions. This *QoI is essentially a measure of trustworthiness* of the summary statistic and is equivalent to its *trust score*. Additionally, user reputation based on his level of truthful participation is required to determine: (i) if a user is honest, selfish, or malicious; (ii) the incentive received by the user; and (iii) acceptance of future reports from the user for decision making.

A. Motivation of this Work

Apart from other expensive methods (e.g., ground truth, sensor based) of evidence collection, a simple way to assess QoI is to allow other users in the proximity to provide a feedback rating (viz., positive, negative, or uncertain) for each published information [20], [32]. Based on such feedbacks (serving as evidence), the event trust and user reputation are quantified. In the literature, most existing trust and reputation models are based on Jøsang's belief model [20], Dempster-Shafer reputation [42] model, or their variants (see Sec. II for details).

A synthesis of existing works reveal that they only utilize the proportion of positive feedbacks in the QoI measure. However, we show that accurate QoI scoring should also include the effect of total number of feedbacks (i.e., feedback mass) that a published information has received. This step is important to weaken the ill effects of malicious ratings. Second, most existing works do not consider a dynamic discounting of uncertain feedbacks to ensure that the QoI measure is null invariant (i.e., not influenced by high uncertainty or orchestrated inconclusive feedbacks). Third, and most importantly, these models are not able to propose a reputation scoring model that unifies both degree of participation (quantity) as well the quality of each contribution. Fourth, existing models provide insufficient provisions for embedding heterogeneity among various CS providers in terms of economic behaviors such as risk tolerance attitude under possibilities of threat and uncertainty.

B. Contributions of the Paper

This paper proposes a model, called QnQ, for trust and reputation scoring in a CS system in presence of malicious and selfish users. First, we propose a QoI measure for every published information. Based on the feedbacks received over a particular published information (event), we calculate the Bayesian inference based belief, disbelief, and uncertainty masses. Thereafter, we model the expected truthfulness of the published information as a regression model using *generalized Richard's curve* and *Kohlsrausch relaxation function* as the weights to belief and uncertainty masses, respectively. This step weakens the effect of malicious feedbacks (such as ballot stuffing or bad mouthing) while also being null invariant against obfuscation stuffing attack in the sense that our model is not influenced by high uncertainty or orchestrated inconclusive feedbacks. Subsequently, we transform the expected truthfulness to a QoI (trustworthiness) measure that captures the odds of an event's occurrence. The transformation is achieved using *cumulative prospect theory (CPT)* inspired link function that captures varying risk tolerance attitudes (risk seeking, risk averse, risk neutral).

Second, we keep track of the QoI measure of all the published information contributed by each user via reports, and then calculate a raw user reputation score by aggregating them. The aggregated raw user reputation is normalized within an interval of [-1, +1] through a logistic distribution function. This normalized user reputation score is: (i) utilized for classification of users into honest, selfish and malicious; (ii) judicially disburse incentives based on both his degree of participation (quantity) and quality of those contributions; and (iii) utilized for robust decision making.

Third, we propose a CPT inspired two-level decision making scheme that exploits the reputation scores and other contextual information to improve accuracy of publishing true events while avoiding false (spam) events. In contrast with other works, a significant benefit of our scheme is that it can embed heterogeneity that might exist among various CS providers in terms of economic behaviors such as risk or loss aversion and avoid certain biases that negatively affect decision accuracy.

Finally, we conduct extensive performance to evaluate the proposed QnQ model using a vehicular crowdsensing system as a proof-of-concept. We use some real data from Waze and Epinions to parameterize the simulation environment. We demonstrate that our approach outperforms Jøsang's belief and Dempster-Shafer (D-S) based reputation models in terms of classification, incentivization, and scalability. Experimental results show that QnQ is able to give a reputation score, that rewards both quality and quantity and saves significantly on incentives in presence of dishonest users while maintaining fairness. Furthermore, we show that our cumulative prospect theoretic decision making scheme ensures better operational accuracy compared to expected utility theory (EUT) based models. We also present some recommendations for the system parameters that show how QnQ can be adapted to any CS application's requirements.

The rest of the paper is organized as follows. Section II summarizes the limitations of the existing literature. Section III describes the system and threat models. Section IV proposes the QnQ model for trust and reputation scores while Section V

extends the QnQ model for trustworthy decision making. Section VI presents results and performance. Section VII discusses parameter recommendations and extensions under various CS systems while Section VIII offers conclusions and future research directions.

II. Limitations of Existing Work

This section reviews the state-of-the-art research for QoI and user reputation scoring models and certain important limitations of existing literature for crowdsensing applications under selfish and malicious users.

A. Quality of Information (QoI) Scoring Models

The QoI scoring aims at assessing the 'veracity' of the information contributed by the users. The veracity assessment may be either on the individual reports or on the inferred information statistic. Broadly, the QoI is assessed by modeling evidence obtained using (i) ground truth, (ii) similarity based outlier detection, (iii) spatio-temporal provenance, (iv) prior reputation context, and (v) the rating feedback mechanism. Such evidence is then mathematically modeled into a QoI.

However, the availability of ground truth is not immediate, and often not guaranteed or feasible. Additionally, acquiring ground truth often requires deployment of dedicated infrastructure, or agents thus obviating the relevance and benefits of crowdsensing. Similarity based outlier detection [2], [18], [40] awards higher QoI to an event if most of the user's contributions agree in terms of event type, location, and time stamp. However, for fake events orchestrated by a group of rogue users, high degree of similarity among contributions is implicit, since honest users will not be reporting anything. Such QoI scoring therefore, fails under orchestrated fake events. Spatio-temporal provenance based schemes [38] assume existence of a prior and reliable reputation scores of each user and quantifies QoI based on prior reputation and similarity in reports. However, this does not address the cold start problem and variability in user behaviors.

Some real CS applications, viz., FourSquare, Waze, YikYak, Yelp, Ebay, etc. use a rating feedback mechanism, whereby other consumers/agents of the service provide positive, negative or neutral ratings on the published information. For example, in case of a 5-star rating system, the ratings 4 and 5 correspond to positive, 1 and 2 as negative and 3 as neutral. The estimation of QoI is achieved based on the feedbacks received. The benefits of using a feedback rating paradigm are that it is easy, fast, and less expensive. Moreover, it really exudes the essence of a true mobile crowd sensing paradigm and do not suffer from weaknesses of the other evidence modeling approaches. In most cases, QoI scoring is done using variants of Jøsang's belief model [20] that compute the QoI based on the ratio of positive feedback to the total feedback with some fixed weight to the ratio of uncertain feedbacks. Nevertheless, there exist threats such as ballot and obfuscation stuffing in rating feedback paradigms. We observe the following inherent weaknesses in Jøsang's belief models:

Confidence of the Feedback Community: Jøsang's belief model (details in Appendix A) fails to capture the differences in confidence of the feedback community, thereby making the resultant expected belief (QoI in our case) more vulnerable to manipulation by malicious raters who provide positive ratings to false events (*Ballot Stuffing attack)* and vice-versa (*Bad Mouthing attack*). This may influence the QoI score of false events in favor of the adversaries. As shown in Table I, each event is denoted as $E : \langle N, r, s, t\rangle$, where N is the total number of received ratings while r, s, t are positive, negative, and uncertain ratings, respectively. For event $E1$, 3 out of 7 feedbacks are good, whereas for event $E2$, 30 out of 70 are good. Jøsang's belief model generates almost the same QoI in both examples. From an adversary's perspective, it is easy to compromise or manipulate 3 good raters in $E1$ and maintain the same fraction of positive ratings as $E2$. However, it is harder to maintain the same fraction when the crowd is large (as in $E2$), in which case the adversary has to manipulate 30 raters. Hence, given the same fraction of positive feedbacks, any event with more feedbacks should be considered as more trustworthy. If this feature is not incorporated, the QoI becomes more vulnerable [8].

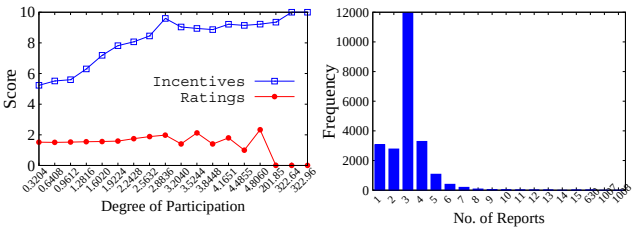
Table I: Limitations of Jøsang's Belief Model

Issues	Examples	Jøsang's QoI
Confidence of Community	E1:$\langle 7, 3, 2, 2\rangle$	0.55
	E2:$\langle 70, 30, 20, 20\rangle$	0.57
Not Null Invariant	E3:$\langle 105, 5, 0, 100\rangle$	0.51
	E4:$\langle 25, 5, 0, 20\rangle$	0.53

Not Null Invariant to Uncertainty: Jøsang's belief models do not offer *null invariance* property. This means that QoI of an event can achieve unwarranted increased trustworthiness due to high proportion of uncertain feedbacks, which may be either intentionally generated (*Obfuscation Stuffing attack*), or be a result of legitimate uncertainty. Either way, such event should not unduly increase the trust (QoI) score. For example, event $E3$ in Table I, has 100 uncertain feedbacks out of 105. However, it achieves almost the same QoI as event $E4$ which in contrast has only 20 uncertain ratings. For most services, it may be risky or unwise to give as high a QoI score to $E3$ as that of $E4$. Thus, the QoI scoring needs a mathematical provision for controlling the impact of high uncertainty on it.

B. User Reputation Scoring Models

Traditionally, reputation scoring models in crowdsensing use either Beta (for binary evidence) or Dirichlet distributions (for multinomial evidence) as theoretical basis for probabilistic and evidence based reputation modeling [27]. Jøsang's belief [20] and Dempster-Shafer (D-S) based trust models [42] provide the state-of-the-art approaches that exploit either of these distributions to model rating feedback based evidence into trust or reputation scores.



(a) Quality vs Quantity (b) Report Generation Frequency

Figure 1: Study on Waze Dataset

Recent works [2], [18], [27] have proposed a deterministic time varying reputation management system based on Gompertz functions. However, rather than evidence based scoring, they mainly investigate the evolution of trust over time and do not actively assume threats. Most of the recent works [2] [3], [32] [31] [13] [39] do not consider orchestrated dishonest reports, consequent economic implications attached with the reputation dynamics, cannot unify quality and quantity of participation, has no provision to handle uncertainty, and cannot thwart the effect of rogue ratings. To our understanding, some limitations of these models are as follows:

Sacrificing Participation for Quality: D-S model [42] (as described in Appendix B) does not fairly capture the degree of participation and quality together into the reputation score. Table II illustrates this limitation. Although users 1 and 2 have the same reputation, the latter with 52 additional good contributions ends up with a score almost similar to user 1. This is unfair and undermines the higher participation of users. If the same high reputation is attainable with lower contributions, the users will not be motivated to participate, and hence the existence of CS will be threatened.

Table II: Sacrificing Participation for Quality

User	Participation	Good	Bad	Dempster Score
1	9	9	0	0.99
2	61	61	0	1.00
3	20	18	2	0.99

Sacrificing Quality for Participation: In [5], we studied a real data set from Waze and identified that quality may be sacrificed for participation. Fig. 1b shows that the majority of the users have generated around three reports over the span of one week. However, there are a few users who have generated a very large number of reports (around 600 to 1000). Additionally, it is evident from Fig. 1a that the incentive of the users gradually increases with higher participation rate. Conversely, the ratings assigned to the users with high participation are very low and even drops to zero while maximum incentive is received. Thus, the reputation score of a user needs to *unify* both degree (quantity) and quality of participation.

Lack of Adaptive Risk Modeling: The general notion of trust and reputation has emphasized the need for incorporating the risk tolerance attitude of the defender. This is because two entities may perceive the same evidence differently because the former might have more to lose than the latter in case of an unfavorable breach of trust. Before entering into a relationship of dependence, the concerns over potential losses caused by possible breaches loom large, if evidence suggest possible threats or presence of considerable uncertainty. Unfortunately, existing trust and reputation scoring models poorly capture the economic behavioral aspect linked to risk perceptions. Our proposed model, on the other hand has provisions to adapt according to different risk attitudes.

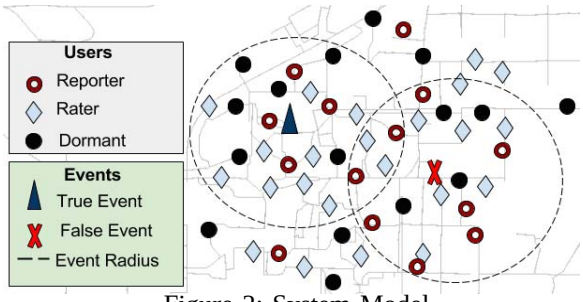


Figure 2: System Model

III. System and Threat Models

In this section, we present the system and threat models and the underlying assumptions.

A. System Model

Fig. 2 depicts the system model which consists of U users, each equipped with a smartphone and subscribed to a vehicular CS application. Two important aspects of this system are:

Report: A report is an alert furnished by a user after he perceives an incident (viz., accident, jam, road closure). However, due to the presence of selfish and malicious users, there may be reports generated for incidents that have never occurred.

Event: An event denoted by $k \in E = \{1, \cdots, |E|\}$ is a summarized information which is published after the CS application receives a predefined number of "similar" reports. Each k belongs to one out of a set of possible event types denoted by $\mathbb{E}$ indexed as $j \in \{1, \cdots, |\mathbb{E}|\}$. If reports from two different users indicate similarity in terms of location, time epoch (t), and event type, they indicate the 'same' event. As evident from Fig. 2, an event can be either true or false and has a boundary within which subscribed users are liable to participate. Event boundaries can be constructed through various geo-spatial clustering methods from GPS stamps to MGRS (Military Grid Reference System) conversion [18].

In the system model, there exists two types of users:

Reporter: A reporter is a user who has a propensity to generate reports and has reported at least one event. Any such user is liable to have a reputation score which reflects the overall quality of reports contributed, as well as the degree of participation. Furthermore, to remove biases from the feedbacks, a reporter is not allowed to rate a published event for which he himself has generated a report.

Rater: A rater is a user who provides feedback on his perceived usefulness of an event as: *Useful* (α), *Not Useful* (β), and *Not Sure* (γ). For a published event, a rater is allowed to submit only one rating.

In our system model, we view the act of providing ratings as an obligation, and it is not rewarded. Hence, for majority of normal users, there is no selfish incentive to provide false ratings, although some false ratings could be motivated by malicious intent.

The design principle of the system model entails collection of as much feedbacks as possible to get a notion of truthfulness of the event in absence of ground truth. This may be achieved by pushing a pop-up rating query to all active users navigating through the event boundary (provided that that user has not reported that event under scrutiny). The rating query asks to click one of three possible options: *Useful*, *Not useful*, and *Not sure*, to gather a subjective judgment about the published event.

B. Threat Model and Assumptions

Dishonest reporters may be selfish or malicious. A selfish user is a legitimate user who generates true and false reports intermittently, with certain probabilities for maximizing his incentives. We consider two variations of selfish users. The first type reports more true events than false events and the other type reports more false events than true events. The rationale of two selfish user subtypes is explained in Section VII-B.

In contrast, malicious users are either actual user devices compromised and (or hardware emulators) controlled by an adversary that may intentionally act in collusion to cripple the CS platform. A part of these devices can be made to act in collusion as reporters to generate a false event while the rest are 'malicious' raters who may act in the following ways:
Ballot stuffing: A rater submits positive feedbacks to an incorrect (false) published event (generated by dishonest reporters).
Obfuscation stuffing: A rater submits uncertain feedbacks to a (false) published event (generated by dishonest reporters).
Bad mouthing: A rater submits negative feedbacks to a legitimate published event (generated by honest reporters).

Note that, hardware emulators can further generate numerous sybil interfaces to magnify the problem of false reports and ratings. However, such sybil interfaces could be identified by existing methods [37]. Hence, we assume that it is only the hardware emulators or compromised devices which pose a threat of colluding attacks. In general, the adversary has a fixed attack budget by which it can compromise or manage a limited number of reporters and raters. This is evident from [37], where the authors generated 1000+ sybil (virtual) interfaces to collude a Waze-like application, but were compelled to deploy only 10 emulators (physical systems) due to budgetary constraints.

Hence, the adversary uses this finite attack budget that effectively manipulates a fraction δ_{mal} of the reporters and raters in order to generate false reports and ratings. It will be a significant fraction for areas with limited number of legitimate users. However, in presence of a significant crowd of independent users, δ_{mal} will be low, and it will not be possible to sabotage the entire proportion of genuine feedbacks. Since crowdsensing paradigms are more prevalent in urban spaces, it may be assumed that for majority of times, substantial number of authentic raters are likely to be present in the vicinity of an event, thus reducing the proportion of false ratings to the total number of feedbacks. In most cases, with larger rater populations, the rating mechanism becomes less likely to get sabotaged. For any rating-based system, the number of raters is always higher compared to the number of reporters generating reports/reviews. Our study from the *Epinions* dataset [25] shows that the number of feedbacks is roughly three to four times the number of reviews (reports) for any item.

Table III: Notations

Symbol	Meaning
N	Total Number of Ratings
k	Published Event ID
α, β, γ	Rating Categories
b, d, u	Probability Masses for Rating Categories
w_b	Weighing Function for b
w_u	Weighing Function for u
A_b, A_u	Initial Asymptote for w_b, w_u
B_b, B_u	Growth Rate for w_b, w_u
ν	Tipping Point for w_b, w_u
φ	Kohlsrausch Relaxation Parameter for w_u
w_u^{max}	Maximum benefit of doubt towards u
τ_k	Expected Truthfulness of a k-th published event
N_{thres}	Rating Mass where discounting of u starts
θ_1	Gain Exponent for QoI value function
ϕ_1	Loss Exponent for QoI value function
Q_k	QoI Score of Event k
S_i	Aggregate Score of User i
R_i	Normalized Final Reputation Score of User i
j	Event Type ID
$\mathcal{C}_j$	Confidence on event type j
$v(\mathcal{C}_j)$	Value function of $\mathcal{C}_j$
$N_{agg}(j)$	Total ratings for the j-th event type
$R_{agg}(j)$	Total reputation for the users reporting j-th event type
ρ	Preference factor for $\mathcal{C}_j$
$U^+(z)$	Total set of active users in z-th region
θ_2	Gain Exponent of $v(\mathcal{C}_j)$
ϕ_2	Loss Exponent of $v(\mathcal{C}_j)$
π^+, π^-	Probability Weighing Functions
δ_1, δ_2	Steepness Exponent for π^+, π^-
p_j	Prior Likelihood of occurrence of j type

IV. QnQ: Proposed Reputation Scoring Model

Now we present the modules of the proposed reputation scoring model, called QnQ. The model is divided into 4 major phases. First, the posteriori probability masses phase calculates for each event published, the belief mass for each rating category using Bayesian inference. Second, the QoI scoring phase calculates QoI for each event through a non-linear weighted regression score (expected truthfulness) followed by a modified Tversky-Kahnemann link function. Third, the user reputation phase accumulates the QoI of events over a time window and associates them with the users who generated them to calculate an aggregate user reputation score. Fourth, the trustworthy decision making phase utilizes the user reputation scores, contextual evidence, etc., to make event publish decision more accurate and is presented in Section V.

A. Posteriori Probability Masses

The first step is to derive the expressions for the posteriori probability masses associated with rating feedbacks: *Useful*,

Not Useful, and *Not Sure*. The probability masses are estimated for each event k based on the available evidence (i.e., supporting each rating type), using a classical Bayesian approach. Let $\bar{\omega} = \{\omega_\alpha, \omega_\beta, \omega_\gamma\}$ be the three tuple probability parameter to be estimated. Here, $\omega_\alpha, \omega_\beta, \omega_\gamma$ are the unknown probabilities of observing a *Useful*, *Not Useful*, or *Not Sure* feedback, respectively. We denoted $H(\bar{\omega})$ as the hypothesis, such that it has three possibilities of either taking α , β or γ. Formally, $P(H(\bar{\omega}) = \alpha|\bar{\omega}) = \omega_\alpha$, $P(H(\bar{\omega}) = \beta|\bar{\omega}) = \omega_\beta$, $P(H(\bar{\omega}) = \gamma|\bar{\omega}) = \omega_\gamma$. Let F_α, F_β, and F_γ be the random variables denoting the number of feedbacks η_α, η_β, and η_γ, received for each feedback category, respectively, such that $N_k = \eta_\alpha + \eta_\beta + \eta_\gamma$. For simplicity, we drop k from all the notations. The *evidence* vector, denoted as $F(N) = \{F_\alpha, F_\beta, F_\gamma\}$, should be modeled as a multi-nomial distribution given by:

$$P(F(N)|\bar{\omega}) = \frac{N!}{\eta_\alpha!\eta_\beta!\eta_\gamma!}\omega_\alpha^{\eta_\alpha}\omega_\beta^{\eta_\beta}\omega_\gamma^{\eta_\gamma} \tag{1}$$

The posteriori hypothesis of positive outcome of any event based on the evidence vector and assumed prior is given as:

$$P(H(\bar{\omega}) = \alpha|F(N)) = \frac{P(H(\bar{\omega}) = \alpha, F(N))}{P(F(N))} \tag{2}$$

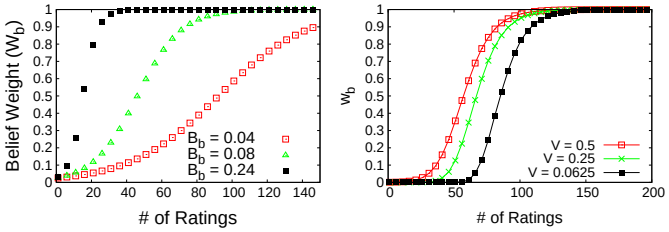
Similarly, the posteriori hypothesis of negative and uncertain outcomes can be represented by replacing α with β and γ respectively in Eqn. (2). Solving the above (see [6]), belief, disbelief, and uncertainty probability masses of an event are derived as follows: $P(H(\bar{\omega}) = \alpha|F(N)) = \frac{\eta_\alpha+1}{N+3} = b$, $P(H(\bar{\omega}) = \beta|F(N)) = \frac{\eta_\beta+1}{N+3} = d$, and $P(H(\bar{\omega}) = \gamma|F(N)) = \frac{\eta_\gamma+1}{N+3} = u$, respectively. These are the posteriori probability masses for *Useful*, *Not Useful*, and *Not Sure* feedbacks as perceived by the raters, respectively. Note that, when $\eta_\alpha = \eta_\beta = \eta_\gamma = 0$, all the possibilities are equiprobable under no information (i.e., non-informative prior).

B. Expected Truthfulness of an Event

Since trustworthiness is related to choice under uncertainty and risk, it is natural that trustworthiness of an event should account for uncertain evidence apart from the positive evidence [11], [20]. Thus, we propose w_b and w_u as the coefficients (or weights) of belief and uncertainty masses respectively, where the weights control the extent to which positive and uncertain probability masses contribute to the truthfulness score of k-th event. The problem is modeled similar to a weighted regression approach where probability masses are explanatory variables and the expected truthfulness is a response variable. We apply Richard's generalized curve [30] and Kohlrausch relaxation functions [4] to model w_b and w_u. The expected truthfulness for any published event k is:

$$\tau_k = (w_b).b + (w_u).u \tag{3}$$

where, $0 < \{w_b, w_u\} < 1$. Hence, $0 < \tau_k < 1$.



(a) Growth Rate:B_b (b) Effect of Various ν

Figure 3: Parameters of Richard's Equation

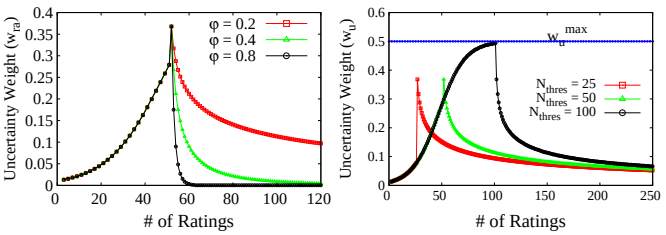
1) Design of Belief Coefficient: We mentioned earlier that expected truthfulness should also consider the volume of the feedbacks, i.e., how many feedbacks have been received for an event apart from the belief mass b. Intuitively, lesser N (total number of feedbacks/ratings) should have lower w_b, which in turn, contributes to a smaller expected truthfulness. However, w_b should gradually increase with N. Thus, to model this nature of w_b, we use a Generalized Richard's Equation normalized between 0 and 1 as:

$$w_b = \frac{1}{(1 + A_b e^{-B_b NR})^{1/\nu}} \tag{4}$$

where $A_b > 0$ but $A_b \neq \infty$ is the initial value of the coefficient, B_b is the rate of growth, and $\nu \neq 0$ is the parameter controlling the point where the curve enters into exponential growth.

Physical Significance of w_b : The w_b is modeled by the Richard's curve (refer to Fig. (3a)) and is motivated from deductive reasoning and learning studies in cognitive psychology [30], [11]. Intelligent humans are subconsciously rational enough to know that the possibility of a biases negatively affecting a belief inference is greater, if fewer number of sources say the same thing (b in this case), as opposed to the same endorsed by more sources. Hence, a Bayesian inference backed by more people/sources carries more weight than the same Bayesian inference backed by fewer people/sources. This phenomena is modeled through *incremental change processes* [30], [11], that are characterized by a slower initial phase followed by an *inflection point* where the learning rate exponentially peaks in the face of increasing evidences and finally saturates into a stationary phase where the learning rate approaches an upper asymptote. Such provisioning enables the CS application to better nullify the effects of ballot stuffing. More details on the advantage of using richards' curve is discussed in our preliminary work [8].

2) Design of Uncertainty Coefficient: In Eqn. (3), w_u controls the contribution of uncertainty mass to the effective truthfulness. Intuitively, uncertainty is high if an incident has just occurred, and the majority of users are uninformed. However, it gets reduced as more feedbacks are received. Thus, for smaller values of N, we should have an increasing function for w_u. As this is also similar to growth curve, we model by a Richard's function upper bounded at w_u^{max}. However, once N attains a threshold value, say $N = N_{thres}$, the coefficient should start to decrease. The value of N_{thres} and w_u^{max} depend on the



(a) Kohlrausch Factor φ (b) N_{thres}

Figure 4: Impact of Parameter Choices on w_u

empirical data of relevant application scenario and risk attitude (as later discussed in Section VII-D.

Typically, Kohlsrausch relaxation function [4] is used to model the property of a system that evolves towards equilibrium after sudden perturbation or a trigger. In our proposed model, after $N = N_{thres}$, this function is used to capture the discounting effect of uncertain ratings on trustworthiness. Its parameter φ, where, $0 < \varphi < 1$, controls the rate of discounting of w_u with N. The larger the value of φ, the more is the decrease (refer to Fig. 4a). The following equation gives the variation of w_u w.r.t the number of received feedbacks:

$$w_u = \begin{cases} \frac{w_u^{max}}{(1+A_u e^{-B_u NR})^{1/\nu}}, & \text{if } NR < N_{thres} \\ e^{-(NR-N_{thres})^{\varphi}}, & \text{if } NR \geq N_{thres} \end{cases} \quad (5)$$

where A_u and B_u are respectively the corresponding asymptote and growth parameters (as discussed in Eqn. (4)). Note that, $0 < w_u^{max} < 1$ is a fixed parameter controlling the maximum allowable benefit of doubt for an event. Choice of w_u^{max} should be guided by risk attitude or availability of trusted agents [32].

Physical Significance of w_u: The concept of trust cannot exist without a certain level of acceptance of uncertainty [10]. Especially for trusting some decisions that tend to be objective, people tend to give some 'benefit of doubt' if uncertainty is reported from a small number of people. But if the same uncertainty mass occurs even as more people/sources have participated, the effect of that uncertainty does not contribute to the increase of trust, since the risk perception is magnified [11]. The uncertainty involves a trigger point (or *knot point*), around which there is a relatively brisk reorientation of the existing state of 'benefit of doubt' into a qualitatively different state of *discounting* the benefit of doubt. Such phenomenon in developmental learning theory is known as *transformational change processes* [30], which fit into a family of spline curves and these phase transitions are modeled by multiple equations around the knot point [10]. The nature of w_u mimics such effects on the modeling of uncertainty evidence. Appendix D shows how this step increases resilience to obfuscation attacks.

C. QoI of Published Event

In Eqn. (3), τ_k is the expectation that the published event k has actually happened. Now, the CS system needs to determine the odds of k-th event being true or false which we model as the QoI. When the response/predictor variables are categorical (true/false, yes/no, etc.), the error distribution is non-normal. We need a *link* function to provide the relationship between the predictor variable and the mean of the distribution defining the QoI. Normally, under risk neutral case, a *logit* link function is used as in our preliminary work [8]. However, when it comes to trust relationships under risk and uncertainty given economic objectives, another factor to be considered is the risk tolerance attitude. Logit link function is inappropriate as it does not have a provision to embed such attitude while making decisions. Therefore, we propose the use of cumulative prospect theory (CPT) inspired link function to embed such risk tolerance. CPT [23] [36] is a descriptive model of how a decision maker perceives/interprets risky prospects that may lead to losses and gains. CPT properties relevant to our work are given below:

1. Reference Point: A decision maker judges a prospect based on the potential gains or losses with respect to a *reference point*, which acts as a neutral boundary about which gains or losses of an outcome are visualized. In our case, $\tau = 0.5$ is the neutral point of the outcome variable.

2. Asymmetrical Value Function: A decision maker is by default risk or loss averse, and thus he strongly prefers avoiding losses than achieving gains. As a result, the value function is *S-shaped* and *asymmetrical*. Mathematically, it is *concave* for gains, *convex* for losses, and steeper for losses than for gains.

3. Principle of Diminishing Sensitivity: A decision maker tends to over-react to smaller deviation from the reference point and the sensitivity decreases at the boundary points.

The link between τ_k and the QoI of the event Q_k is established by the following modified value functions from CPT:

$$Q_k = \begin{cases} (\tau_k)^{\theta_1}, & \text{if } \tau_k \geq 0.5 \\ -\lambda_1(0.5-\tau_k)^{\phi_1}, & \text{if } \tau_k < 0.5 \end{cases} \quad (6)$$

where Q_k has the value in the interval $[-\lambda_1, 1]$, $\lambda_1 > 1$, $\theta_1 > 0$, and $0 < \phi_1 < 1$. The variations of QoI with respect to different parameters are depicted in Fig. 5.

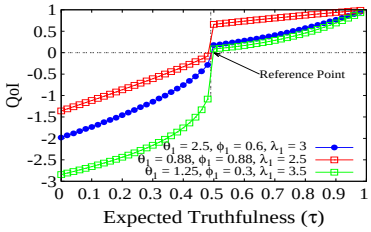


Figure 5: Risk Aware Modified Link Function

As evident from Eqn. (6), Q_k has two value functions on either side of the reference point, ($\tau_k = 0.5$). The exponents θ_1 and ϕ_1 enable the CS administrator to control the rate of change of QoI above and below the reference point respectively. The values taken may vary according to the risk attitude. For instance, if the administrator wants to prevent loss of revenue and business goodwill of the application, he exhibits risk averse attitude through a very gradual increment of QoI (with $\theta_1 > 1$)

w.r.t to increasing expected truthfulness (in Fig. 5, blue and green curves: *convex* nature) starting from the neutral point. However, if the application can afford a 'less' risk averse approach by setting $\theta_1 < 1$ (red curve: *concave* nature). The choice of θ_1 in Eqn. (6) can embed the contextual risk tolerance of various PS providers accordingly.

In contrast, the nature of QoI below the reference point is always *convex* and its steepness depends on the exponent ϕ_1. This is because, the CS administrator will always show loss aversion attitude and perceive steeper loss of quality with $\tau < 0.5$. The loss penalty parameter λ_1 determines the lowest value which the QoI can attain. Lower QoI will incur penalties on the defaulters and the administrator will be satisfied if it encourages rogue users to churn out of the system. This will implicitly prevent loss of both revenue and operational reliability of the application.

D. QoI-based User Reputation Score

For any reporter i, we match the reports he had generated with the estimated QoI value of the corresponding events. We sum up Q_k for every unique event reported by reporter i, to calculate the aggregate reputation score S_i.

$$S_i = \sum_{k=1}^{|E|} Q_k I(k,i) \tag{7}$$

$$\text{where,} \quad I(k,i) = \begin{cases} 1, & \text{If } i \text{ reported event } k \\ 0, & \text{Otherwise} \end{cases} \tag{8}$$

E. Normalized User Reputation Score

The aggregated reputation score S_i obtained from Eqn. (7) is a real number in the interval $[-\infty, +\infty]$. In order to make it intuitive and consistent with the definition of trust metrics, we use the logistic distribution function to map its values in the interval $[-1, +1]$. Therefore, the final reputation score (R_i) of a reporter i is given as:

$$R_i = \begin{cases} +\left(\frac{1}{1+e^{-\frac{S_i-\mu_s^+}{C^+}}}\right), & \text{if } S_i > 0 \\ -\left(\frac{1}{1+e^{-\frac{|S_i|-|\mu_s^-|}{C^-}}}\right), & \text{if } S_i < 0 \\ 0, & \text{if } S_i = 0 \end{cases} \tag{9}$$

where μ_s^+ and μ_s^- are the mean reputation scores for reporters with positive and negative S_i respectively. Similarly, $C^+ = \frac{\sqrt{3}\sigma_{s^+}}{\pi}$ and $C^- = \frac{\sqrt{3}\sigma_{s^-}}{\pi}$ where σ_{s^+} and σ_{s^-} are the standard deviations for reporters with positive and negative S_i respectively. At t-th epoch, R_i^t, denotes the steady state reputation at the t-th epoch. The reputation may be calculated at the end of a predefined time window of multiple epochs.

V. Trustworthy Decision Making Scheme

The work until now was dedicated to build a reasonably genuine user base via computation of QoI through feedbacks received against published events. Once such an user base is formed, the event publishing step itself could be made more dependable and accurate. In this section, we show how the QnQ reputation can be applied for trustworthy and dependable 'event publishing' decisions. As discussed in Section III, a vehicular CS (like Waze) application may receive reports that indicate either of the following event types, viz., jam, accident, road closure, weather hazard from a potential event boundary, say z. Such an event boundary can be constructed through a geospatial grid clustering of received GPS stamped reports [18]. Intuitively, we should only consider reports from users with $R_i > 0$ for the decision making in z.

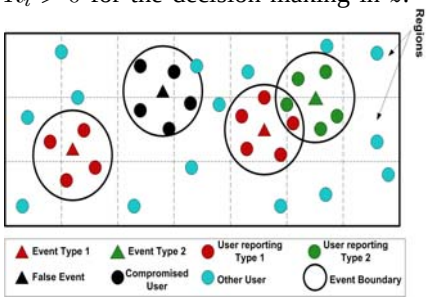


Figure 6: Decision Making Scenarios

However, accurate decisions are still non-trivial in absence of ground-truth due to the following possibilities: First, potential event boundary constructed from clustering step are not disjoint as shown in Fig. 6. This can produce a mixture of relevant and irrelevant reports for a given event. Second, uncertainty looms regarding the actual behavior of the new users (with no reputation) registered in the current time epoch. Third, some users can build higher reputation first, and then start orchestrating fake events. Fourth, although prior spatio-temporal likelihood is often considered, we believe that event occurrence is still not deterministic. Hence, rarer events may not be favored and never get published, while malicious users can orchestrate a fake event in a location with high prior likelihood, both triggering wrong decisions.

Wrong decisions ('publishing' fake events or 'not publishing' real events) will incur losses, while correct decisions (publishing events that occurred and not publishing when no event occurs) will accrue gains. Therefore, the decision making is confronted with the following four utility prospects: (i) *'publish' given event type j 'did not occur'* ($\mathcal{P}|\overline{e}$), incurring a loss of l_1; (ii) *'not publish' a given event type j that actually occurred* ($\overline{\mathcal{P}}|e$), incurring a loss l_2; (iii) *'publish' given event type j actually occurred* ($\mathcal{P}|e$), which incurs a gain of g_1; and (iv) *'not publish' given event type j did not occur* ($\overline{\mathcal{P}}|\overline{e}$), that incurs a gain of g_2. The gains are positive utilities ($g_1, g_2 > 0$) while losses are negative utilities ($l_1, l_2 < 0$), such that $g_1 > g_2 > l_1 \geq l_2$, is the ordering of the utility prospects in terms of decreasing profits. Since, the real goal is

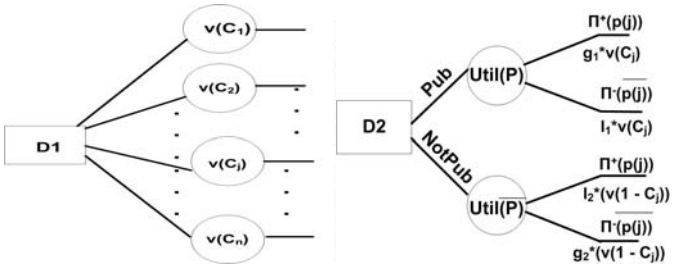
to publish an event when it occurs, g_1 is strictly greater than g_2. If an event has not occurred, but the CS system has received supporting reports and still it successfully prevent itself from publishing the false information, then it will be considered as a finite positive gain $g_2 > 0$. In the loss front, l_2 could be greater than l_1, if missing a true event is causing more loss than publishing a fake event and vice-versa. However, for our work, we considered them equally bad and assumed $l_1 = l_2$. To solve this, we propose a two-level decision process that maximizes the gain corresponding to correct decisions while accounting for the losses for wrong decision depending on the application's risk policy.

Decision Levels: The application needs to determine in near real-time: ($D1$) *what* event type to publish (i.e., the most likely event type that has occurred), and ($D2$) *whether* to publish an event (i.e., if sufficient evidence exists to suggest that the most likely event has actually taken place).

The first decision ($D1$) is required since reports belonging to more than one event type may be received at the same time epoch in the event boundary z. Therefore, the first decision is to decide in runtime the most likely event type (winner event). To achieve this, a confidence value for each reported event type j is computed based on the relative quantity and quality support for each event type j.

The second decision ($D2$) is required because, even if there is a clear likely event type, there may not be strong overall evidence to suggest that publishing this event will result in a benefit or gain. This is particularly true for preventing orchestrated fake events, because honest reporters will not report anything in the absence of any event. Such decision problems are modeled as a decision tree [28].

Solution Methodology: We need to compute the final utilities of both 'publish' $util(\mathcal{P})$ and 'not publish' decisions $util(\overline{\mathcal{P}})$ and the event is published if $util(\mathcal{P}) > util(\overline{\mathcal{P}})$. Most decision trees are classically solved by the *expected utility theory (EUT)*. However, recent research in behavioral economics and decision theory showed strong empirical evidence decision making under risks and uncertainty does not follow EUT. In fact, cumulative prospect theory (CPT) [23] [36] in recent decades have become a stronger and realistic descriptive model for decision making under risk and uncertainty particularly by humans. While, CPT was originally proposed as descriptive model for human decision making, we propose a modified application of the original theory which fits an automated decision making system. The goal of such modification is to preserve the advantages of CPT while simultaneously avoiding certain biases humans suffer from, that result in occasional irrational choices. As an example, we show that our model is more likely to publish true events with low prior likelihood of occurrence, and avoid fake events at locations which have a high prior likelihood in contrast to existing approaches. To the best of our knowledge, such an efforts have not been made in automated trustworthy decision making for CS applications.



(a) Level 1 (b) Level 2

Figure 7: Two Level Decision Scheme

The decision trees for the proposed two-level decision scheme are given in Fig. 7a and Fig. 7b, respectively. For the first decision level $D1$, we calculate the raw confidence of each type j, adjust it with the risk attitude to a utility value. The second decision level $D2$ adds another step that handles the effect of prior likelihood of events for the decision making. Then the final utilities of $util(\mathcal{P})$ and $util(\overline{\mathcal{P}})$ are computed. The different steps of the proposed decision scheme are explained below:

1) Raw Confidence of an Event Type: Let the CS application receive reports from different users from a common event boundary z. Till the last time epoch, the reputation scores of the users present within z are known. These reports may or may not endorse the same event type. Let each particular event type j receive $N_{agg}(j)$ number of reports. Additionally, let the aggregated reputation score of all users reporting the j^{th} event type be denoted as $R_{agg}(j)$. Let $|U^+(z)|$ be the total number of users currently present in z, each of them has reputation $R_i > 0$. Thus, $\sum_{i \in U^+(z)} R_i$ denotes the total reputation of all active users $i \in U^+(z)$. Therefore, for the j^{th} event, $\frac{N_{agg}(j)}{|U^+(z)|}$ and $\frac{R_{agg}(j)}{\sum_{i \in U^+(z)} R_i}$ are the two evidences that denote relative support for 'quantity' and 'quality' respectively. Mathematically, we model the overall confidence on any event type j ($\mathcal{C}_j$) using a weighted sum of quantity and quality supporting each event type j.

$$\mathcal{C}_j = \rho . \frac{N_{agg}(j)}{|U^+(z)|} + (1 - \rho) . \frac{R_{agg}(j)}{\sum_{i \in U^+(z)} R_i} \tag{10}$$

where, $0 \leq \rho \leq 1$ is the 'preference factor' associated with the evidence types, $N_{agg}(j)/|U^+)(z)|$ is the participation fraction of event j. The preference factor ρ is controlled by contextual information, such as spatio-temporal information of a given type of event or risk policy. For example, if the event occurs in a prior known inherently crowded location, the CS administrator may give higher weightage to the quantity support $\frac{N_{agg}(j)}{|U^+(z)|}$. Conversely, for inherently sparsely crowded locations where participation is lower, higher importance may be given to the quality support $\frac{R_{agg}(j)}{\sum_{i \in U^+(z)} R_i}$. In the absence of any such information, $\rho = 0.5$.

2) Value Function of Event Confidence: This step embeds the risk aversion policy into the event confidence. Now, we need to determine the value of publishing event type j. Since $0 \leq \mathcal{C}_j \leq 1$, the midpoint where $\mathcal{C}_j = 0.5$ is perceived

as the neutral reference point from the perspective of utility value. Hence, all $\mathcal{C}_j > 0.5$ and $\mathcal{C}_j < 0.5$ instills the notion of gains and losses respectively and accordingly perceived as less risky or more risky. Since a CS application is never absolutely certain about the outcome, it can tune the interpretation of $\mathcal{C}_j$ according to risk aversion policy or changing context. In order to replicate a risk attitude while making decisions under uncertainty, we again use the CPT value functions (CPT property-2 in Section IV-C) for estimating the values:

$$v(\mathcal{C}_j) = \begin{cases} (\mathcal{C}_j)^{\theta_2}, & \text{if } \mathcal{C}_j \geq 0.5 \\ -\lambda_2.(0.5 - \mathcal{C}_j)^{\phi_2}, & \text{if } \mathcal{C}_j < 0.5 \end{cases} \tag{11}$$

The $v(\mathcal{C}_j)$ ranges $[-\lambda_2, 1]$, while the exponents $\theta_2 > 0$ and $0 < \phi_2 < 1$ control the rates of growth of gains and loss regions, respectively. The loss penalty parameter λ_2 determines the lowest value which $v(\mathcal{C}_j)$ can attain. All $\lambda_2 > 1$ imply risk aversion.

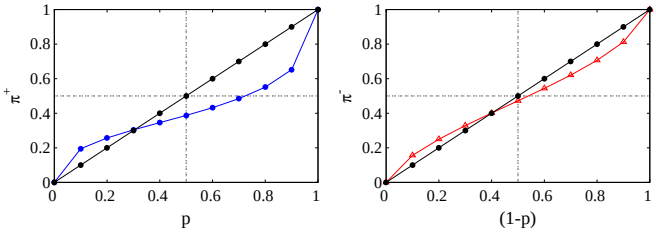
3) First Level Decision ($D1$): At level $D1$ (refer to Fig.7a), the CS administrator needs to find the event type which yields the maximum value. The event type with largest value is the winner and is selected as a candidate for publishing. Thus:

$$v_{max} = \max_{\forall j \in \mathbb{E}}(v(\mathcal{C}_j)) \tag{12}$$

In this level, if more than one event types yield the maximum value, or if the largest value function is negative, then all the types are discarded and no event is published. The reason is that the available evidence, in terms of the number of reports and the aggregated reputation score supporting j^{th} event, is not sufficient to convincingly choose one particular winner event type, and the uncertainty of publishing *what* still persists. However, if we find only a specific event type with maximum value, we move onto the second level. Formally, we define the first level decision process as:

$$D1 = \begin{cases} \text{Select } j, & \text{iff } v_{max} = v(\mathcal{C}_j) \\ & \text{and } v_{max} > 0 \\ \text{No } j \text{ is selected}, & \text{Otherwise} \end{cases} \tag{13}$$

4) Handling Bias of Prior Likelihood of Event Types: Prior likelihood of any j occurring at a particular region and time epoch biases decision making. In the second decision $D2$, the CS administrator decides *whether* to publish a winner event type j. However, if no j is chosen, $D2$ decision level is not invoked. If we denote the occurrence likelihood of the event type j to be the *positive outcome*, then its complement (i.e., non-occurrence) will be the *negative outcome* given a publish decision. According to CPT, utility of a prospect is obtained by multiplying its value by a decision weight to obtain its utility [36]. These weights measure the impact of events on desirability of the prospects, and not merely on the perceived likelihood and in general the decision probability weighing functions should be concave near 0 and convex near 1. Instead of one probability weighing function, we use



(a) $\pi^+(p_j)$: Occurrence (b) $\pi^-(\overline{p_j})$: Non-Occurrence

Figure 8: Probability weighing functions

different decision probability weighing functions and their steepness of concavity and convexity are different for positive (prior likelihood of occurrence) and negative outcomes (complement of occurrence). Following equations give the weights for positive and negative outcomes:

$$\pi^+(p_j) = \frac{p_j^{\delta_1}}{(p_j^{\delta_1} + (1 - p_j)^{\delta_1})^{\frac{1}{\delta_1}}}, \tag{14}$$

$$\pi^-(\overline{p_j}) = \frac{(\overline{p_j})^{\delta_2}}{((\overline{p_j})^{\delta_2} + (1 - \overline{p_j})^{\delta_2})^{\frac{1}{\delta_2}}} \tag{15}$$

where p_j and $\overline{p_j}$ are likelihoods of the occurrences of event j and its complement respectively, and $0 < \delta_1, \delta_2 < 1$ controls the steepness of the decision weighing functions $\pi^+(p_j)$ and $\pi^-(\overline{p_j})$ respectively. The steepness determines how large π^+ and π^- will be for smaller values of p_j and $\overline{p_j}$, and vice-versa (see Figs. 8a and 8b). The weighing functions for positive and negative outcomes are closer, although the former is more curved than the latter (i.e., $\delta_1 < \delta_2$), such that strong prior likelihoods underweigh the historical bias on the decision function.

5) Final Utilities of Prospects: The sum of utilities to be gained from publishing j is given as: $util(\mathcal{P}) = g_1 * v(\mathcal{C}_j) * \pi^+(p_j) + l_1 * v(\mathcal{C}_j) * \pi^-(\overline{p_j})$. Similarly, the sum of utilities to be gained from not publishing it is: $util(\overline{\mathcal{P}}) = l_2 * v(1 - \mathcal{C}_j) * \pi^+(p_j) + g_2 * v(1 - \mathcal{C}_j) * \pi^-(\overline{p_j})$. Here, $v(1 - \mathcal{C}_j)$ evaluates the value of discarding the confidence ($\mathcal{C}_j$) generated from the evidences received in the current time epoch.

6) Second Level Decision ($D2$): We formally define the second level decision as:

$$D2 = \begin{cases} \text{Publish } j, & \text{if } (util(\mathcal{P}) - util(\overline{\mathcal{P}})) > 0 \\ \text{Not Publish}, & \text{Otherwise} \end{cases} \tag{16}$$

VI. Experimental Study

In this section, we evaluate the performance of QnQ and compare them with Jøsang's belief model [20] for expected truthfulness (QoI), D-S model [42] for reputation scoring. We also compare the performance of the proposed trustworthy decision scheme against EUT.

A. Simulation Settings and Datasets

We simulated a realistic environment for vehicular crowdsensing system by extracting important simulation parameters from the Waze data set [5] and Epinions dataset [25]. The Waze data comprises of reports for four major traffic event types: *jam (ja), accident (ac), weather hazard (wh)*, and *road closure (rc)*. It has approximately 22,910 users, 71,505 reports, spanning across 10 geographical regions adjacent to Boston, USA. In each region, prior probabilities of occurrences (likelihood) of different event types have been computed from the dataset, as summarized in Table IV.

Table IV: Event Probabilities from Waze Dataset

Region	P(ja)	P(ac)	P(wh)	P(rc)
1	0.48	0	0.52	0
2	0.75	0.01	0.2	0.02
3	0.56	0.008	0.19	0.23
4	0.66	0.008	0.33	0
5	0.47	0	0.53	0
6	0.86	0.02	0.12	0
7	0.79	0.01	0.19	0
8	0.74	0.01	0.25	0
9	0.45	0.02	0.53	0
10	0.45	0.02	0.52	0

For simulation, we consider a city area of 20 X 20 sq. miles as the region of interest. This area is partitioned into ten rectangular grids to replicate regions from the dataset. The system is initialized with $U = 2400$ number of active users, among which $U_{rp} = 800$ are reporters and $U_{rt} = 1600$ are raters. We extracted a realistic expected ratio of reporters to raters by studying an Epinions dataset collected from [25] since the Waze data did not offer this information. We assume the presence of 520 dishonest devices out of which 120 (i.e., 15% of total reporters) are used for generating false reports and 400 (i.e., 25% of total raters) are used for false ratings. These devices have been distributed uniformly in the simulated city area at the start of the simulation. The total simulation time is slotted into $T = 240$ number of epochs, each of which is of duration 30 minutes.

We consider an event to have a fixed radius (5 miles) within which all reporters and raters are liable to report or rate. Each event has a tunable lifetime within which reports and feedbacks are accepted. For example, if an event occurred in epoch t and the duration of its lifetime is two epochs, then it can be reported and rated until epoch $t + 2$. The probability of event type j in a particular region is extracted from Table IV.

We consider random paths along which a user moves with speeds of 20-50 miles/epoch. We parameterize the number of raters and ratings to account for all possible realistic combinations. However, we considered that users progressively leave the region of interest mimicking a dense location becoming sparse over time to capture effects of crowd movement.

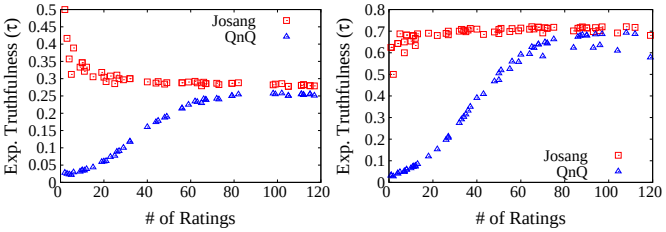
For the reporters, we emulate honest, selfish, and malicious behaviors in the following ways. 20% of the reporters are programmed as selfish, while 15% act as malicious and the rest act honestly. Given that an event has occurred, an honest reporter reports 99% of the time and has minuscule probability of generating a false report (simulating occasional wrong perception). Malicious reporters within a randomly generated location (chosen for false event) collude to generate fake reports of a fictitious event with high probability $\approx 100\%$. One class of selfish users reports more true events (about 60%) than false events, while the other class reports fewer true events (about 40%) than false events.

For the raters, the compromised raters give positive ratings to false events and negative ratings to true events, while the honest raters provide genuine ratings with 5% legitimate uncertainty. Note that a user reporting a particular event is prevented from rating it. The percentages of compromised raters corresponding to an event varies with the variation in the population size. Since 400 out of 1600 raters are compromised, *on average* the fake rating percentage for true and false events is about 25%. We have discussed its effects in the scalability analysis (see Section VI-F). Let the parameters of our system take the following values: $A_b = A_u = 20$, $\nu = 0.25$, $\varphi = 0.2$, $N_{thres} = 60$ and $w_u^{max} = 0.5$. Only the parameters B_b and B_u are adjusted during runtime of QoI scoring. The growth rate parameter is adjusted to $B_b = B_u = 0.08$ if low feedbacks are received for a particular event. For higher feedbacks received, we keep $B_b = B_u = 0.04$. The value function parameters considered for simulation are $\theta_1 = 2.5$, $\phi_1 = 0.6$, and $\lambda_1 = 3$. An analytical study of effect of varying these parameters is provided in Appendix D.

B. Expected Truthfulness (QoI) of Events

Fig. 9a illustrates a comparison between the expected truthfulness (QoI score) achieved by QnQ vs. Jøsang's belief model for a false event. We observe that QnQ refrains from giving an undue high QoI score, unlike Jøsang's model for low ratings. As higher number of ratings are received, the confidence of the crowd and the uncertainty discounting is taken into account to converge to the true value, preventing malicious raters to harness an advantage. This is however not true for Jøsang's model, and false events end up getting higher scores even if the number of ratings were smaller. In contrast, Fig. 9b shows the the QoI score comparison for a true event. For QnQ, the QoI converges to the true value only after sufficient number of ratings are received, while for Jøsang's model this aspect does not matter. This is essential to prevent potential sabotaging by an organized minority of rogue raters.

Note that QnQ will always assign low QoI to events receiving low feedbacks. When the number of ratings are limited, there could be two possible options: (i) the published event may not be significant enough and does not draw attention of majority of raters, resulting in low QoI and (ii) the place has an inherently low population, implying N is not very high. The parameters A_b, A_u, B_b, B_u and ν could be tuned to achieve higher QoI score at comparatively lower number of ratings to adapt to contextual requirements (explained in Section VII-D).



(a) False Event (b) True Event

Figure 9: QoI Score Comparison

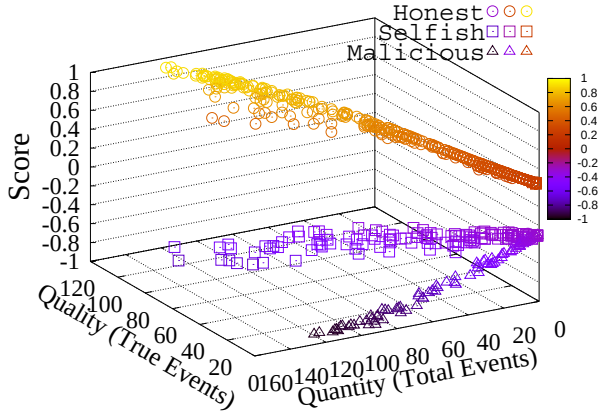


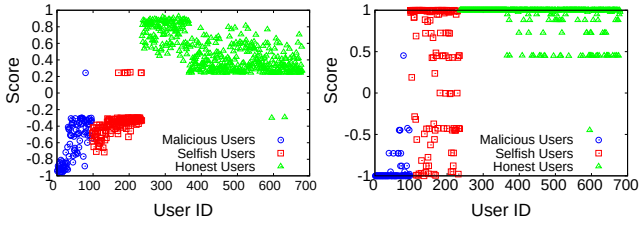
Figure 10: User Reputation: Quality and Quantity

C. User Reputation Scores

We evaluate the performance of reputation scoring with a goal to unify quality and quantity, achieve fair classification of user behaviors, and compare with existing work.

1) Unifying Quality and Quantity: Fig. 10 shows how QnQ is able to reflect both quantity (i.e., total number of events participated) and quality (i.e., the number of events found to be true) of participation in the resultant user reputation score. The first observation is that three distinct user groups emerged. The lowest group corresponds to malicious, the middle group to selfish, and the top group to honest users. Another key observation is that selfish and malicious users cannot increase their reputation by boosting up only participation. Since selfish users intermittently contribute true and false events, their scores are higher than malicious but lower than honest users.

2) Classification of Users with Fairness: We considered two different types of selfish users: (i) those who report more true events than false events, and (ii) those who report more false events than true events. Intuitively, selfish users with higher number of genuine contributions should have higher scores than others from the same class. However, it is noteworthy that the penalty factor of reporting a fake event is higher ($\lambda_1 = 2$) than the reward for generating a true one. Hence, the majority of both categories of selfish users end up with negative score. This aspect is evident in Fig. 11a. However, only few selfish users (around six out of 160) owing to their participation in true events (with higher QoI) manage to have positive scores. Likewise, very few honest and malicious users end up having negative and positive scores, respectively.



(a) Proposed QnQ (b) D-S

Figure 11: Comparison: Reputation based User Classification

These are the outliers to our user behavior classification.

Table V compares the reputation scores of various user classes and their outliers. Here n_i is the number of events for which user i has generated reports. Honest user #1 has very low event participation compared to that of honest user #2, and hence has a lower score. Although selfish user #1 has reported more true events than #2, both have reported a large number of false events, leading to negative scores. Evidently, malicious users (malicious #1 and #2) reporting majority of false events have negative scores.

3) Comparison with Dempster Shafer (D-S) Model: Our model exhibits better performance in terms of accuracy and fairness than D-S based reputation score as shown in Fig. 11b, where many selfish users end up with very high scores.

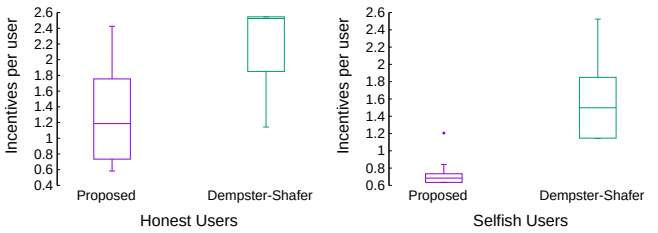
Table V: Comparative Reputation Scores

Type	n_i	True #	False #	Score
Honest #1	3	3	0	0.245
Honest #2	100	99	1	0.842
Selfish #1	41	26	15	-0.346
Selfish #2	37	13	24	-0.462
Malicious #1	4	0	4	-0.299
Malicious #2	102	2	100	-0.919

D. Reducing Incentive Losses

QoI-aware incentive mechanisms account for quality of each sensing report before making incentive/reward assignments. The QoI metrics can be broadly classified into two categories: (i) reputation scoring based micropayments [19], (ii) satisfaction index-based involving data quality in terms of sampling rate, accuracy, similarity, and timeliness [29], [35]. In particular, [29] proposes an Expectation Maximization (EM) algorithm to estimate "effort matrix" for the participants, which captures the goodness of reports in terms of temporal proximity of the reported data with the time interval of ground truth occurrence. A scalar function maps the effort matrix to a QoI score which forms the basis of a reward mechanism that achieves both individual rationality and profit maximization.

Beside this, game-theoretic (auction-based) incentive mechanisms exist but some of their limitations include: (a) Rationality of agents: Consider human users to be perfectly rational agents and absence of malicious participants in the sensing task [17], [41]; (b) System/Computational inefficiency: Consider incentivization as a maximum coverage problem which is essentially NP-hard [44]. Therefore, the system and



(a) Honest (b) Selfish

Figure 12: Protecting Undue Incentive Leakage

computational efficiency of the mechanism are not guaranteed; (c) Static tasks and users: Few works have assumed incentive task and the number of users in the system to be static [12], [15]. However, for real mobile crowdsensing applications, such assumptions may not hold.

Based on the above discussions, we argue the incentive mechanism in mobile crowdsensing applications should have the following features: (i) computationally inexpensive, (ii) dynamic, and (iii) maintain fairness. Dynamism entails that over time the reward for users should change and fairness is guaranteed if participants are incentivized based on both the changing quality and quantity of contributions. Incidentally, the reputation score generated by QnQ encompasses these attributes. Furthermore, consideration of a simpler reputation score based reward function will ensure a lightweight incentive mechanism, suitable for real-time systems like vehicular crowdsensing applications. Thus, we adopt the reputation based incentive function presented in [32] as our choice. In [32], the incentive received by user i at the end of t^{th} time epoch is:

$$I_i^t = \frac{R_i^t}{\sum_{k=1}^{U^+} R_k^t} . \frac{B.U^+}{U} \tag{17}$$

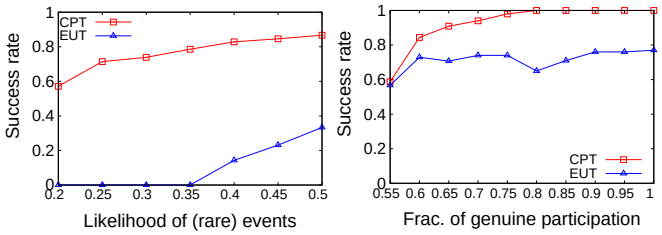
where R_i^t is the reputation score of user i as computed after t^{th} time epoch, U^+ is the number of users in the system with positive reputation score, B is the total incentive budget allocated in time epoch t, and U is the total number of users in the system. The fraction $\frac{R_i^j}{\sum_{k=1}^{U^+} R_k^j}$ acts as a discounting factor to the maximum possible incentive $\frac{B.U^+}{U}$ any user can gain. Thus, the user with relative reputation on the higher side will yield less discount and ends up getting handsome reward and vice versa. Fig. 12a shows that QnQ offers a larger variation of incentives disbursed to the honest users according to the variations in quality and quantity. However, the D-S model gives higher incentives since it only awards quality but not quantity. Hence, users with lower participation also end up with a high score and hence a higher incentive. In contrast, Fig. 12b shows that mean QnQ-based incentives for selfish users is 50% that of honest ones and is three times smaller than that yielded by D-S based reputation model. Unlike D-S model, QnQ can distinguish between honest and selfish behaviors, and penalize the latter with low rewards thus preventing loss of revenue due to false contributions.

E. Trustworthy Decision Making Accuracy

As mentioned, the simulator was run for 240 time epochs to generate a history of occurrences of events, and a set of eligible reporters with reputation score greater than 0. Following this, we again run the simulator for another 240 epochs to evaluate the operational accuracy of the proposed decision scheme.

In practice, eligible reporters may get compromised at the current time epoch (zero-day attack) or experience wrong perception of a true event. Moreover, given that an eligible reporter generates a false event, he generates the correct event type with probability 1.

We evaluate the performance of our CPT-inspired decision scheme against EUT-based model by computing (i) *success rate*, i.e., the fraction of true events successfully published among of all true events that actually occurred (ii) *detection rate*, i.e., fraction of true events published among all published events. The objective of our decision scheme is to ensure that rare events with sufficient quality and quantity support has a higher chance of getting published.



(a) Success Rate vs p_j (b) Success Rate vs $F_g(U^+)$

Figure 13: Performance Evaluation of Decision Scheme

The values of various parameters are considered as follows: $\theta_2 = \phi_2 = 0.88$, $\lambda_2 = 2.25$, $\rho = 0.5$, $g_1 = 2$, $g_2 = 1$, and $l_1 = l_2 = -1$. For performance analysis of CPT versus EUT, we analyze the success rates against the two following parameters: (i) p_j as the prior likelihood of occurrence of an event (rare) and (ii) $F_g(U^+)$ as the fraction of users reporting correctly among all eligible users (with $R_i > 0$) (termed as fraction of genuine participation).

Publishing Low Likelihood Events: Fig. 13a shows that the success rates of the proposed CPT-based model is significantly better than EUT for publishing true events whose prior likelihood of occurrences are very low ($p_j < 0.5$). The reason is CPT uplifts the likelihood of occurrences of rare events (less than 0.3), and thereby increases the publishing utilities. This ensures that the rare events do not remain unpublished if it generates higher confidence in the current time epoch. To realize this scenario, we have considered the probability of reporting the accurate event type for a true event as 0.75.

Fraction of Genuine Participation $F_g(U^+)$: As shown in Fig. 13b, CPT yields notable improvement in success rate over EUT for any fraction of genuine participation greater than 0.5. This is because, unlike EUT, CPT produces an enhanced value for the confidence of true events, and thereby increases the number of published true events. Consequently,

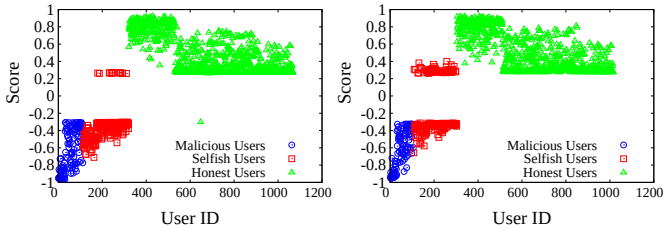
the success rate for CPT gradually reaches 1 with increase in the fraction of genuine participation. Similarly, a comparison of the detection rates of our CPT inspired decision scheme is shown to outperform EUT in Appendix C.

F. Scalability and Robustness of Performance

Unlike all prior plots with total number of users $U = 2400$, let us we consider $U = 1200$ and $U = 3600$ (fake devices inclusive), under the same population 520 dishonest devices. For $U = 1200$ (low population scenario), malicious users form about 43% of the total population, and there is presence of a very low proportion of genuine raters (only 45% of all raters). Although the chances of such a scenario is rare, it is still possible and some conservative systems may want to understand the performance limits of the defense model. Now we examine the scalability in the context of risk aversion and risk neutral attitudes. Fig. 15a shows the reputation scores for a risk averse system for $U = 1200$. Note that, we still succeed to keep all malicious and selfish users in the lower reputation tier with negative scores. Interestingly, our model misclassifies all the honest users too due to the presence of very low proportion of genuine raters (only 45% of all raters). Since our system follows a protective risk averse approach, this sacrifice is significant. For the risk neutral approach, Fig. 15b shows that we still manage to put all malicious and selfish users in the lower reputation tier but the misclassification of honest into the malicious tier is much smaller. However, as and when the crowd increases (under $U = 3600$), the reputation of all honest users are improved for both risk averse and risk neutral approaches, reinforcing the significance of the crowd.

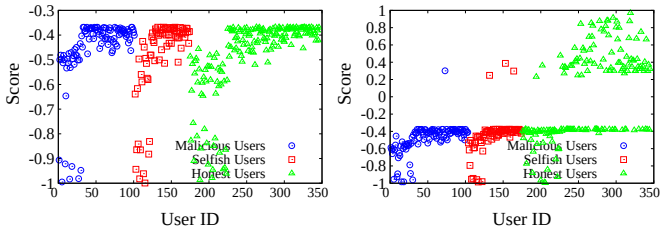
Fig. 14a is the reputation score distribution for the risk averse system (with value function as link function) while Fig. 14b, shows the same for a risk neutral system (with classical logit link function (used in our preliminary work)). Here, we see that the risk aversion embedded by the value function is better at keeping selfish users regardless of their subtype in the lower reputation tier, while for risk neutral systems, all selfish users with more true events than false events are in the positive side on the reputation scale. For conservative systems, the system may want to keep all kind of selfish users from being considered for any sort of decision making. Thus, it is evident that the risk tolerance attitude is not only related to losses, gains and uncertainty but also to the scalability aspect. We intend to study in our future work, how a system can perform better with fewer misclassification of honest users than the current case when the population is low, system is risk averse with bad mouthing attacks.

Classification accuracy is expressed in terms of whether an honest user and dishonest user is accurately inferred or not. If a legitimate selfish and malicious user is classified as honest then it is a missed detection while if an honest user is classified as anything else it is a false alarm. Missed detections and false alarms are an index of classification accuracy that is affected by varying population sizes and attack budgets which



(a) Risk Averse (b) Risk Neutral

Figure 14: Scalability with $U = 3600$

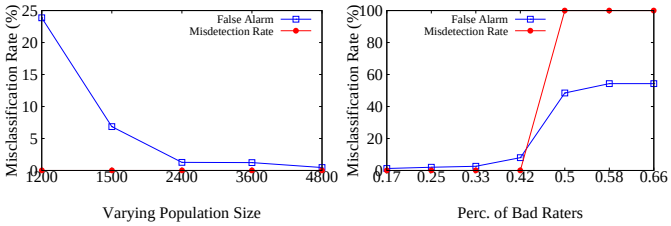


(a) Risk Averse (b) Risk Neutral

Figure 15: Scalability with $U = 1200$

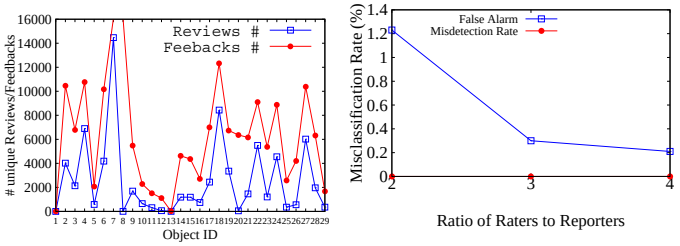
we study through Figs. 16a, 16b and 17b. Fig. 16a, shows the classification accuracy when the attack budget is fixed (520 fake devices) and the rating population varies, showing that larger crowd populations are more robust. On the other hand, Fig. 16b, shows the effect of varying attack budgets under a fixed population size of 3600. It is evident from Fig. 16b, that attackers need to control a sizeable number of the crowd (around 52%) for the classification to fail completely. However, 52% for a crowd of 3600 is about 1800 devices that attacker needs to control which is very expensive.

Additionally, in most crowdsensing systems, the volume of ratings are always higher than reporters. We verified this claim from Epinions dataset across different items in the dataset (See Fig. 17a above). Many other works have studied Yelp and Amazon datasets and found similar observations. In most paradigms including Waze, there is no incentive to provide ratings on reviews/reports, but the volume of ratings remain high due to the relative ease of the pop-up feedback. Fig. 17b, clearly shows that missed detection rates are minimal and false alarm rates become miniscule when the raters are about 4 times the reporters. In general as the ratio of raters to reporters increases the performance only improves. To show conservative results we have assumed a much lesser rater to reporter ratio for most results.



(a) Fixed Attack Budget (b) Varying Attack Budget

Figure 16: Robustness against Various Attack Budgets and Rating Sample Sizes



(a) Real Rating Distribution (b) Varying Reporter-Rater Ratios

Figure 17: Additional Illustrations

VII. Discussions

In this section, we present various discussions on possible extensions of QnQ and parameter recommendations under varying assumptions of rating systems, risk attitudes, availability of trusted agents.

A. Extension to Multinomial or Real Valued Rating

The QnQ framework can be easily extended under crowdsensing systems with more than 3 rating levels or have real valued ratings. For a 5-point rating mechanism viz., Bad, Mediocre, Average, Good, Excellent (proposed in [22]), the levels bad and mediocre may correspond to the disbelief mass, while good and excellent form the belief mass, and average correspond to the uncertainty mass. In IMDB like 10 point-grading mechanisms levels 1-4 will correspond not useful or negative feedbacks, 5,6,7 are uncertain and 8,9,10 are useful or positive feedback categories. In case of real valued ratings, a membership function may be used that discretizes the real valued rating space into discrete rating levels.

B. Rationale of Two Selfish User Subtypes

The rationale of two sub-types of selfish users are inspired from certain real world incidents. For example, in Los Angeles California, news reports [45] surfaced that residents in a particular uptown residential area generated fake reports such that Waze would not reroute traffic through their neighborhood. Of course, fake reports are legitimately possible when such users are located in that particular area. Suppose an user goes out for work in the downtown area where she spends 10 hours of her day. She could only false report during the evening when her location is in this target uptown residential area. On the other times of the day, she has no selfish incentive to report false events. Conversely, a user who works from or stays at home throughout; for her it makes complete sense to generate a fake report on jam because these reports are auto-GPS stamped. Occassionaly, when she goes out she does not produce fake reports since she does not any incentive to produce false report at other locations. Thus for the same selfish objective in the same area, different users could have different quantities of true and false reports. Conversely, there may be users whose selfish objective is in terms of maximizing her incentives. In such case, this user will often have the urge to report some event, even if she is in a place that is relatively less eventful. Note that, this is a selfish user with more false events than true events.

C. Filtering of Rogue Raters

The report and rating are different roles that may be performed by the same physical user (interface). Over several time epochs, this assignment of the malicious roles to physical interfaces have two possible alternatives. First, the same physical (user) interface can act as a fake reporter on certain time epochs and fake rater at other time epochs, depending on their location/convenience. Second, each physical interface controlled by an adversary has a fixed role that it only act as a fake reporter or a fake rater across all time epochs. In the first case, the false rater is a role attached to the physical interface, just like a false reporter. When our QnQ framework identifies and isolates the false reporting users, they implicitly isolate the fake raters too. Hence, a separate reputation mechanism is not required under the first case. For the second case, where each malicious interface has fixed roles; once the initial false reporters in malicious team are identified by our reputation mechanism, there will be no option left for the adversary but to use some of the fake rater interfaces as fake reporters. Else, there will be no interfaces left for fake events, hence the purpose of having fake raters will be defeated. In such a case, our proposed mechanism will eventually detect all these malicious interfaces incrementally.

D. Parameter Choices under Risk Attitudes

Parameters need to be adjusted according to risk attitudes for following functions: (i) coefficients of belief mass w_b (A_b, B_b, ν), and (ii) coefficient of uncertainty mass w_u (A_u, B_u, N_{thres}, φ, w_u^{max}). (iii) coefficients of value function (θ_1,θ_2,ϕ_1,ϕ_2,λ_1,λ_2), (iv) coefficients of weighing function (δ_1,δ_2). A provider could be *risk averse* in terms of the QoI and reputation scoring by having a conservative increase of scores. It could be *risk neutral or risk seeking* in terms of the QoI and reputation scoring with more liberal increase of scores with evidence. Any instance of availability of trusted agents means that the provider may afford to lessen its risk aversion.

1) Choice of A_b, B_b and ν: The A_b is the base value of the weight given to belief mass when no rating is received. If the system is not restrictive, then a higher initial weight w_b is required, and hence a lower value of A_b is recommended. In contrast, for a conservative system, the initial weight of w_b should be very low, to ensure that it should acquire a sufficient number of ratings before attaining a substantial weight.

Fig. 3a shows the effect of B_b that controls the number of ratings N required to attain the maximum possible value of w_b once it enters the exponential phase. For example, if the concerned area is inherently crowded and higher N is expected, then B_b should be kept low such that the full weight to w_b is awarded only after a sufficient number of ratings is received. If the system is less restrictive, it can lower the value of B_b.

The parameter ν controls the value of N at which the curve first enters into the exponential growth phase. A lower value of ν is preferred if the CS system expects receipt of false ratings, or if the location historically receives lower number of ratings. Fig. 3b shows the different values of ν. To conclude, the more risk averse a provider, the smaller is the ν, the smaller is the B_b and larger is the A_b, to be chosen in the belief coefficient. This is because more evidence in terms of ratings are required for w_b to attain a higher weight which controls how positive ratings contribute to the final trust values.

2) Choice of φ, N_{thres} and w_u^{max}: The Kohlrausch factor φ determines how quickly w_u discounting effect reaches minimum after N_{thres} is reached. Fig. 4a shows the effect of various choices of φ. A CS system chooses a higher value of φ if the proportion of uncertainty needs to be immediately discounted or vice versa. Effects of A_u and B_u to w_u are similar to that of A_b and B_b to w_b.

A small N_{thres} would prevent w_u to reach its maximum value, before the uncertainty discounting starts. This is true for more conservative systems and is evident from Fig. 4b. A low w_u^{max} may be required when the CS administrator comes to know about the ground truth (from other sources such as mobile trusted participants [32]), and does not want uncertainty mass to obtain higher weights. To conclude, the more risk averse a provider, the smaller is the w_u^{max} and larger is the φ parameter, and smaller N_{thres}. If the trusted agents, then we should have a smaller w^{max} and a smaller N_{thres}, and larger φ.

3) Choice of θ_1, θ_2, ϕ_1, ϕ_2, λ_1, λ_2: The parameters $\{\theta_1, \theta_2\} < 1$ gives us a risk seeking system, while $\{\theta_1, \theta_2\} > 1$ gives a risk averse system. A larger ϕ gives a risk seeking system, while a smaller ϕ is risk averse. A larger λ is a penalty factor that is high if the system is more risk averse. The problem of being risk averse in scoring is that some users may end up being demotivated due to lesser scores. For example, if the provider already has a decent user base, $\{\theta_1, \theta_2\} > 1$ is recommended.

4) Choice of δ_1, δ_2: The parameters δ_1 and δ_2 control the curvatures of weighing functions at their endpoints (finite asymptotes). If most of the events in a region are non-recurring in nature, then intuitively high prior likelihoods should not be accounted for decisions on event publishing. In such cases, a high δ_1 values is required and lower δ_2 is required. If a system has events that show recurrence or periodicity, the δ_1 value should be lower and δ_2 should be higher.

VIII. Conclusions

In this work, we addressed the issue of quality of information and reputation scoring in crowdsensing (i.e., vehicular CS application) and propose a regression-based reputation model, QnQ, which is resilient to rogue contributions and null invariance. The model assesses the QoI for a published event by incorporating the cardinality of rating feedback, proportion of positive support, and uncertainty in ratings. The QoIs of relevant events are aggregated to generate the final reputation score of a user. The resultant reputation score provides a clear segregation among honest, selfish and malicious users, and implicitly guarantees fairness within each segregated group without sacrificing either participation or quality. Further, we propose CPT-based decision scheme which takes the generated reputation score as input and supports publish/not publish decisions, and implicitly ensures operational reliability of the CS application. Extensive analytical and simulation study was carried out to establish the efficacy of the proposed approach in terms of scalability, fairness, and decision accuracy. Finally, we present the recommendations on system parameters to enable QnQ adapt under varying conditions of risk and uncertainty. In future, we will study incentive and classification trade-offs under risk averse and risk seeking systems for varying crowded locations.

Acknowledgements: The work has been supported by the following NSF grants: CNS-1818942, CNS-1545037, CNS-1545050, and DGE-1433659. A major portion of the work was completed at Missouri S & T, where Dr. Shameek Bhattacharjee was a post-doctoral fellow during 2015-2018, Dr. Nirnay Ghosh was a post-doctoral research fellow during 2016-2017, and Vijay Shah was a PhD student until August 2017.

References

[1] H. Amintoosi and S. S. Kanhere, "A Trust-based Recruitment Framework for Multi-hop Social Participatory Sensing", *IEEE DCOSS*, pp. 266-273, 2013.

[2] H. Amintoosi and S. S. Kanhere, "A Reputation Framework for Social Participatory Sensing Systems", *Springer Mobile Networks and Applications*, vol. 19, no. 1, pp. 88-100, 2014.

[3] H. Amintoosi, S. S. Kanhere, and M. Allahbakhsh, "Trust-based Privacy-aware Participant Selection in Social Participatory Sensing", *Journal of Information Security and Applications*, vol. 20, pp. 11-25, 2015.

[4] R. S. Anderssen, S. A. Husain, and R. Loy, "The Kohlrausch Function: Properties and Applications", *Anziam Journal*, vol. 45, pp. 800-816, 2004.

[5] R. P. Barnwal, N. Ghosh, S. K. Ghosh, and S. K. Das, "Enhancing Reliability of Vehicular Participatory Sensing Network: A Bayesian Approach", *IEEE SMARTCOMP*, pp. 1-8, 2016.

[6] S. Bhattacharjee and M. Chatterjee, "Trust based Channel Preference in Cognitive Radio Networks under Collaborative Selfish Attacks", *IEEE PIMRC*, pp. 1502-1507, 2014.

[7] S. Bhattacharjee, N. Ghosh, V. K. Shah, and S. K. Das, "W2Q: A Dual Weighted QoI Scoring Mechanism in Social Sensing using Community Confidence", *IEEE PerCom Workshops*, pp. 375-380, 2017.

[8] S. Bhattacharjee, N. Ghosh, V. K. Shah, and S. K. Das, "QnQ: A Reputation Model for Securing Mobile Crowdsourcing Systems from Incentive Losses", *IEEE Communications and Network Security*, October 2017.

[9] J. A. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava, "Participatory Sensing", *Center for Embedded Network Sensing*, 2006.

[10] R. Cudeck and K.J. Klebe, "Multiphase Mixed-Effects Models for Repeated Measures Data" *Psychological Methods*, vol 7, no. 1, pp. 41-63, 2002.

[11] J.R. Eiser and M.P. White, "A Psychological Approach to Understanding how Trust is Built and Lost in the Context of Risk",*Conference on Social Contexts and Responses to Risk*, 2005.

[12] Z. Feng, Y. Zhu, Q. Zhang, H. Zhu, J. Yu, J. Cao, L.M. Ni, "Towards Truthful Mechanisms for Mobile Crowdsourcing with Dynamic Smartphones" *IEEE ICDCS*, pp. 11-20, 2014.

[13] S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, "Reputation-based Framework for High Integrity Sensor Networks" *ACM Trans. on Sensor Networks*, vol. 4, no. 3, pp. 15:1-15:37, 2008.

[14] R. K. Ganti, F. Ye and H. Lei, "Mobile crowdsensing: current state and future challenges," in IEEE Communications Magazine, vol. 49, no. 11, pp. 32-39, November 2011.

[15] L. Gao, F. Hou, J. Huang "Providing Long-Term Participatory Incentive in Participatory Sensing", *IEEE INFOCOM*, pp. 2803-2811, 2015.

[16] R. D. Gupta and D. Kundu, "Generalized Logistic distributions", *Journal of Applied Statistical Science*, vol. 18, no. 1, 2010.

[17] B. Hoh, T. Yan, D. Ganesan, K. Tracton, T. Iwuchukwu, J. Lee. "TruCentive: A game-theoretic incentive platform for trustworthy mobile crowdsourcing parking services, *Proc. of ITSC*, pg. 160–166, 2012.

[18] K. L. Huang, S. S. Kanhere, and W. Hu,"Are You Contributing Trustworthy Data?:The Case for a Reputation System in Participatory Sensing", *ACM MSWiM*, pp. 14-22, 2010.

[19] K. L. Huang, S. S. Kanhere, W. Hu, "On the Need for a Reputation System in Mobile Phone based Sensing", *Elsevier Ad Hoc Networks*, vol. 12, pp. 130-149, 2014.

[20] A. Jøsang,"An Algebra for Assessing Trust in Certification Chains", *NDSS*, 1999.

[21] A. Jøsang and R. Ismail, "The Beta Reputation System", *Bled eConference*, pp. 41-55, 2002.

[22] A. Jøsang, J. Haller, "Dirichlet Reputation Systems", *IEEE Conf. on Availability, Reliability and Security*, pp. 112-119, 2017.

[23] D. Kahneman and A. Tversky, "Prospect Theory: An Analysis of Decision under Risk", *Econometrica*, pp. 263-291, 1979.

[24] I. Koutsopoulos, "Optimal Incentive-driven Design of Participatory Sensing Systems", *IEEE INFOCOM*, pp. 1402-1410, 2013.

[25] P. Massa and P. Avesani, "Trust-Aware Bootstrapping of Recommender Systems", *ECAI Workshop on Recommender Systems*, pp. 29-33, 2006.

[26] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: Rich Monitoring of Road and Traffic Conditions using Mobile Smartphones", *ACM SenSys*, pp. 323-336, 2008.

[27] H. Mousa, S. Mokhtar, O. Hasan, O. Younes, M. Hadoud, and L. Brunie "Trust Management and Reputation Systems in Mobile Participatory Sensing Applications: A survey" *Computer Networks*, vol. 90, pp. 49-73, 2015.

[28] R. Neopolitan, "Learning Bayesian Networks", *Prentice Hall*, 2003.

[29] D. Peng, F. Wu and G. Chen, "Data Quality Guided Incentive Mechanism Design for Crowdsensing," *IEEE Trans. on Mobile Computing*, vol. 17(2), pp. 307-319, Feb. 2018.

[30] N. Ram, K. Grimm, "Handbook on Child Psychology, Developement Science and Methods: Growth Curve Modeling and Longitudinal Factor Analysis", *Wiley*, pp. 758-785, 2015.

[31] S. Reddy, D. Estrin, and M. Srivastava, "Recruitment Framework for Participatory Sensing Data Collections". *International Conference on Pervasive Computing (Springer)*, pp. 138-155, 2010.

[32] F. Restuccia and S. K. Das, "FIDES: A Trust-based Framework for Secure User Incentivization in Participatory Sensing", *IEEE WoWMoM*, pp. 1-10, 2014.

[33] F. Richards, "A Flexible Growth Function for Empirical Use", *Journal of experimental Botany*, vol. 10, no. 2, pp. 290-301, 1959.

[34] M. B. Sinai, N. Partush, S. Yadid, and E. Yahav, "Exploiting Social Navigation", *arXiv preprint arXiv:1410.0151*, 2014.

[35] C. Tham and T. Luo, "Quality of Contributed Service and Market Equilibrium for Participatory Sensing" *IEEE Trans. on Mobile Computing*, Vol. 14(4), April 2015.

[36] A. Tversky and D. Kahneman, "Advances in Prospect Theory: Cumulative Representation of Uncertainty", *Journal of Risk and Uncertainty*, vol. 5, no. 4, pp. 297-323, 1992.

[37] G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, and B. Y. Zhao, "Defending Against Sybil Devices in Crowdsourced Mapping Services", *ACM MobiSys*, pp. 179-191, 2016.

[38] X. O. Wang, W. Cheng, P. Mohapatra, and T. Abdelzaher, "Enabling Reputation and Trust in Privacy-Preserving Mobile Sensing", *IEEE Trans. on Mobile Computing*, vol. 13, no. 12, pp. 2777-2790, 2014.

[39] Q. Xiang, J. Zhang, I. Nevat, and P. Zhang, "A Trust-based Mixture of Gaussian Processes for Robust Participatory Sensing", *ACM AAMAS*, pp. 1760-1762, 2017.

[40] L. Xiao, Y. Li, G. Han, H. Dai and H. V. Poor, "A Secure Mobile Crowdsensing Game With Deep Reinforcement Learning," in IEEE Trans. on Information Forensics and Security, vol. 13, no. 1, pp. 35-47, Jan. 2018.

[41] C. Yi, S. Huang and J. Cai, "An Incentive Mechanism Integrating Joint Power, Channel and Link Management for Social-Aware D2D Content Sharing and Proactive Caching," *IEEE Trans. on Mobile Computing*, vol. 17(4), pp. 789-802, 1 April 2018.

[42] B. Yu and M. P. Singh. "An Evidential Model of Distributed Reputation Management", *ACM AAMAS*, pp. 294-301, 2002.

[43] R. Yu, R. Liu, X. Wang, and J. Cao, "Improving Data Quality with an Accumulated Reputation Model in Participatory Sensing Systems", *Sensors Journal*, vol. 14, no. 3, pp. 5573-5594, 2014.

[44] D. Zhao, X. Li, and H. Ma, "Budget-Feasible Online Incentive Mechanisms for Crowdsourcing Tasks Truthfully" *IEEE/ACM Transactions on Networking*, Vol. 99, 2015.

[45] [Online] (2014, Nov.) Irate Homeowners Are Spoofing Waze, Available: https://jalopnik.com/irate-homeowners-are-spoofing-waze-to-reroute-la-traff-1660192011.

[46] [Online] (2015, Feb.) Sending fake data to Waze, Available: https://www.yahoo.com/news/miami-cops-sending-fake-data-waze-stop-people-210552132.html

Shameek Bhattacharjee is an Assistant Professor at the Department of Computer Science at Western Michigan University, USA. He received his Ph.D. and M.S. from the University of Central Florida, Orlando in 2015 and 2011 respectively and his B.Tech from West Bengal University of Technology, India, 2009. Between 2015-2018, he worked as a post-doctoral researcher at the Missouri S & T at Rolla, MO, USA, where he is also affiliated as an Adjunct Faculty. His current research interests include information security in cyber-physical systems, wireless and social networks, particularly in topics such as anomaly detection, trust models, secure crowd-sensing, and dependable decision theory. He is a receipient of Provost Fellowship and IEEE PIMRC Best Paper Award.

Nirnay Ghosh is a research fellow at the iTrust Research Center for Cyber Security, in the Singapore University of Technology and Design (SUTD). Prior to this, he worked as a postdoctoral fellow at the Missouri S & T (MST), Rolla, USA. Dr. Ghosh completed his masters (MS-by research) and PhD from the Department of Computer Science and Engineering, IIT Kharagpur, in 2010 and 2016, respectively. His research interest and experience include graph theory applications to security, secure cloud computing, participatory sensing, and internet-of-things. He is receipient of TCS Research Fellowship and IEEE ADCOM Best Paper Award.

Vijay K. Shah is a Ph.D. student at the Department of Computer Science, University of Kentucky, USA. From Jan. 2015-Aug. 2017, he was a PhD student at Missouri S & T. His Ph.D. advisors are Dr. Simone Silvestri and Dr. Sajal K. Das. He has authored/co-authored over a dozen scientific papers in top-tier networking journals and conferences, including ACM TOSN, IEEE TMC, ACM BuildSys, IEEE INFOCOM, and IEEE CNS. His research interests primarily include Wireless Networked Systems, Spectrum Sharing, 5G Networks, Smart Cities, Internet of Things.

Sajal K. Das is a professor of Computer Science and the Daniel St. Clair Endowed Chair at the Missouri University of Science and Technology, where he was the Chair of Computer Science Dept. during 2013-2017. His research interests include cyber-physical security and trustworthiness, wireless sensor networks, mobile and pervasive computing, crowd-sensing, cyber-physical systems and IoTs, smart environments (e.g., smart city, smart grid and smart health care), cloud computing, biological and social networks, applied graph theory and game theory. He has published over 700 research articles in high quality journals and refereed conference proceedings. Dr. Das holds 5 US patents and coauthored 4 books – Smart Environments: Technology, Protocols, and Applications (John Wiley, 2005), Handbook on Securing Cyber-Physical Critical Infrastructure: Foundations and Challenges (Morgan Kauffman, 2012), Mobile Agents in Distributed Computing and Networking (Wiley, 2012), and Principles of Cyber-Physical Systems: An Interdisciplinary Approach (Cambridge University Press, 2018). His h-index is 83 with more than 28,000 citations according to Google Scholar. He is a recipient of 10 Best Paper Awards at prestigious conferences like ACM MobiCom and IEEE PerCom, and numerous awards for teaching, mentoring and research including the IEEE Computer Society's Technical Achievement Award for pioneering contributions to sensor networks and mobile computing, and University of Missouri System President's Award for Sustained Career Excellence. He serves as the founding Editor-in-Chief of Elsevier's Pervasive and Mobile Computing Journal, and as Associate Editor of several journals including the IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Mobile Computing, and ACM Transactions on Sensor Networks. Dr. Das is an IEEE Fellow.

Appendix for:
QnQ: A Quality and Quantity Unified Approach for Secure and Trustworthy Crowdsensing

1 Appendix A

Expected Bayesian belief (E^J) is given as: $E^J = b + a.u$, where $b = \frac{r+1}{r+s+t+3}$; $d = \frac{s+1}{r+s+t+3}$; $u = \frac{t+1}{r+s+t+3}$; and r, s, and t denote the number of positive, negative, and uncertain ratings. The value $a = 0.5$ is the relative atomicity which is equal to the reciprocal of the cardinality of inference state space $\{true, false\}$.

2 Appendix B

D-S based reputation model uses ternary feedback-based evidence space. For a particular agent, each witness assigns probability values to the three elements (viz. trustful, distrustful, uncertain) in the state space, thus forming the belief, disbelief, and uncertainty masses, respectively. The probability masses obtained from different sources are combined by an orthogonal sum ($\bigoplus$) operator and then the difference between overall belief and disbelief masses gives the agent's reputation score. The orthogonal sum operator is defined as follows:

Let Bel_1 and Bel_2 be the belief functions of two sources over a frame of discernment Θ, with the belief mass assignments m_1 and m_2 respectively. If the focal elements be given as $x_1, x_2, \ldots, x_k$ and $y_1, y_2, \ldots, y_l$ respectively, then the combined belief mass function $m_\Theta^{1,2} = m_\Theta^1 \bigoplus m_\Theta^2 : 2^{|\Theta|} \mapsto [0,1]$ is defined by:

$$m_{1,2}(\Phi) = 0$$

$$m_{1,2}(x) = m_1(x) \bigoplus m_2(x) = \frac{\sum_{i,j,x_i \cap y_j = x} m_1(x_i).m_2(y_j)}{1 - \sum_{i,j,x_i \cap y_j = \phi} m_1(x_i).m_2(y_j)}$$

As the underlying principle of both QnQ and D-S reputation model is based on classical Bayesian, we considered the latter for fair comparison. Mathematically, Dempster-shafer theory proposes an orthogonal sum operator $\oplus$ which combines evidences from multiple sources to generate cumulative belief, disbelief and uncertainty functions.

For example, if Bel_1 Bel_2 $\cdots$ are the belief functions obtained from different sources, the cumulative belief is given as: $Bel(i) = Bel_1 \bigoplus Bel_2 \bigoplus \cdots$ and the cumulative disbelief is: $DisBel(i) = DisBel_1 \bigoplus DisBel_2 \bigoplus \cdots$

Then, the reputation of agent i will be given by

$$Rep(A_i) = Bel(A_i) - DisBel(A_i) \quad (1)$$

3 Appendix C

The following Fig. 1, shows that our better performance during decision making is preserved over time compared to the EUT performance.

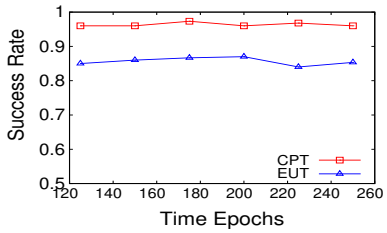


Figure 1: Success Rate vs Time Epochs

Figure 2: Decision Model Comparison over Time Epochs

4 Appendix D

The generalized logistic curve is parameterized by a variable N, is given as:

$$w(N) = L + \frac{U - L}{(1 + A.e^{-BN})^{1/\nu}} \quad (2)$$

where, L is the lower asymptote, U is the upper asymptote, B is the growth rate, ν is the parameter that controls the value of N at which the curve enters the exponential growth phase, and A is the initial value of the coefficient at $w(0)$.

5 Appendix E

In this section, we compare the QoI score generated by QnQ with Jøsang's expected truthfulness (E^J), which is equivalent to τ_k in our approach (because the scale of both metrics have to be between 0 and 1 for fair comparison).

To better explain, the five important aspects of QoI and reputation scoring in QnQ, we consider two different scenarios: (i)*sparsely-crowded* location, and (ii) *densely-crowded* location. Let the total number of ratings received be 50 and 300 for the sparse and the dense locations respectively. The event published by the CS administrator based on the reports can be either true or false. Thus, there could

be four possible scenarios: (a) false event/sparse location, (b) false event/dense location, (c) true event/sparse location, and (d) true event/dense location. The densities keep changing all the time. Under the four cases, let the adversary have a budget to manage a fixed number of raters (say 20), in every location, who can pose the following four types of threats: (i) false ratings to true events (bad mouthing); (ii) true ratings to false events (ballot stuffing); (iii) deliberate undecided ratings to boost up expected truthfulness of false events (obfuscation stuffing); and (iv) combine false and undecided ratings (mixed attack). Let the parameters of our system take the following values: $A_b = A_u = 20$, $\nu = 0.25$, $\varphi = 0.2$, $N_{thres} = 60$ and $w_u^{max} = 0.5$. Only the parameters B_b and B_u are adjusted in the runtime of QoI scoring. For cases (a) and (c), which correspond to a sparse location that has a low number of feedbacks, the growth rate parameter is adjusted $B_b = B_u = 0.08$. For cases (b) and (d), which correspond to a dense location, $B_b = B_u = 0.04$. Note that the only difference is in the growth parameter B which mostly depends on the expectation that more ratings will be available over time.

For cases (a) and (b), where the event is false, we do not consider the bad mouthing attack as it is not relevant, as such attacks are meant to manipulate a true events into false by deliberate fake (negative) ratings. We represent the rating distributions for each threat barring bad mouthing by a tuple $\langle r, s, t\rangle$. The expected truthfulness scores generated by Jøsang's model and QnQ are depicted in Table 1. It is clearly evident from Table 1, for sparse location,

Table 1: Case-a: False Event at a Sparse Location

Threat	$\langle r, s, t\rangle$	**b/d/u**	**Jøsang**	**QnQ**
No threat	$\langle 5, 42, 3\rangle$	0.11/0.81/0.075	0.128	0.036
Ballot stuffing	$\langle 25, 22, 3\rangle$	0.49/0.43/0.07	0.525	0.147
Obfuscation	$\langle 5, 22, 23\rangle$	0.11/0.43/0.45	0.335	0.093
Mixed	$\langle 15, 22, 13\rangle$	0.3/0.43/0.26	0.43	0.12

the truthfulness score assigned by QnQ to false events are much less compared to that given by the Jøsang's model. Moreover, our model can readily detect the obfuscation attack, assigning it lowest value.

For case (b), the truthfulness comparison is presented in Table 2. If the location is densely populated, the QoI score assigned by Jøsang's model is relatively less compared to sparse locations. However, it is still on the higher side compared to the scores generated by QnQ, which computed the coefficients as $w_b = 0.99$ and $w_u = 0.05$. Like (A), here also QnQ is particularly able to be severe on obfuscation attack.

For cases (c) and (d), as the event is true, ballot stuffing attack is not practical since that is meant to manipulate a false event to true by deliberate fake (positive) ratings. Tables 3 and 4 give the comparison of truthfulness values computed by the two models in these two cases. For

Table 2: Case-b: False Event at a Dense Location

Threat	$\langle r, s, t\rangle$	**b/d/u**	**Jøsang**	**QnQ**
No threat	$\langle 25, 260, 15\rangle$	0.085/0.86/0.052	0.111	0.087
Ballot stuffing	$\langle 45, 240, 15\rangle$	0.15/0.79/0.052	0.176	0.15
Obfuscation	$\langle 25, 240, 35\rangle$	0.085/0.79/0.12	0.205	0.09
Mixed	$\langle 35, 240, 25\rangle$	0.12/0.79/0.085	0.163	0.12

Table 3: Case-c: True event at a Sparse Location

Threat	$\langle r, s, t\rangle$	**b/d/u**	**Jøsang**	**QnQ**
No threat	$\langle 30, 15, 5\rangle$	0.58/0.3/0.11	0.635	0.177
Bad mouthing	$\langle 30, 35, 5\rangle$	0.42/0.49/0.08	0.46	0.128
Obfuscation	$\langle 30, 15, 25\rangle$	0.42/0.22/0.36	0.6	0.156
Mixed	$\langle 30, 25, 15\rangle$	0.42/0.36/0.22	0.53	0.164

case (c), it is clearly evident that Jøsang's model assigns high QoI even . However, QnQ refrains from assigning higher truthfulness value even to true events unless it receives a substantial number of ratings. The truthfulness value given by QnQ is lowest under bad mouthing attack, which shows that our model is less robust if rogue raters give deliberate false ratings to true events. The values of the coefficients computed here are $w_b = 0.28$ and $w_u = 0.14(\text{normal}), 0.2(\text{other threats})$. Unlike the other

Table 4: Case-d: True event at a dense location

Threat	$\langle r, s, t\rangle$	**b/d/u**	**Jøsang**	**QnQ**
No threat	$\langle 180, 90, 30\rangle$	0.597/0.3/0.102	0.648	0.596
Bad mouthing	$\langle 180, 110, 30\rangle$	0.56/0.34/0.09	0.605	0.558
Obfuscation	$\langle 180, 90, 50\rangle$	0.56/0.28/0.16	0.64	0.562
Mixed	$\langle 180, 100, 40\rangle$	0.56/0.31/0.13	0.625	0.56

three cases, the truthfulness scores assigned in (d) by both models are on the higher side and are at par with each other. This is because, the event is true and substantial ratings have been received, which has lead to the generation of high scores.

Summarizing the results depicted in Tables 1, 2, 3, and 4, we draw three important observations: (i) QnQ is resilient to *ballot stuffing* attack by preventing false events to be portrayed as a true ones for both low and high number of ratings; (ii) unlike Jøsang's model, QnQ is completely null-invariant and thwarts the threat of *obfuscation*; and (iii) the proposed model is robust against *bad mouthing* attacks, if substantial number of ratings are available.