The Quality Control in Crowdsensing based on Twice Consensuses of Blockchain

Danwei Liang
Xi'an Jiaotong University
Xi'an 710049, China
linxin1994123@163.com

Jian An*
Xi'an Jiaotong University Shenzhen Research School
High-Tech Zone, Shenzhen,
518057, P.R. China
anjian@mail.xjtu.edu.cn

Jindong Cheng
Xi'an Jiaotong University
Xi'an 710049, China
Dongzaixijiao155@mail.xjtu.edu.cn

He Yang
Xi'an Jiaotong University
Xi'an 710049, China
yangxiaohe51@163.com

Ruowei Gui
Xi'an Jiaotong University
Xi'an 710049, China

UbiComp/ISWC '18 Adjunct, October 8–12, 2018, Singapore, Singapore

ACM ISBN 978-1-4503-5966-5/18/10…$15.00
https://doi.org/10.1145/3267305.3267547

Abstract
In most crowdsensing systems, the quality of the collected data is varied and difficult to evaluate while the existing crowdsensing quality control methods are mostly based on a central platform, which is not completely trusted in reality and results in fraud and other problems. To solve these questions, a novel crowdsensing quality control model is proposed in this paper. First, the idea of blockchain is introduced into this model. The credit-based verifier selection mechanism and twice consensuses are proposed to realize the non-repudiation and non-tampering of information in crowdsensing. Then, the quality grading evaluation (QGE) is put forward, in which the method of truth discovery and the idea of fuzzy theories are combined to evaluate the quality of sensing data, and the garbled circuit is used to ensure that evaluation criteria can not be leaked. Finally, the Experiments show that our model is feasible in time and effective in quality evaluation.

Author Keywords
Crowdsensing; quality control; blockchain; consensus.

ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

Introduction

As the rapid development of information technology, intelligent terminals become popular in people's lives, which provide the impetus for expand of crowdsensing [4, 5], the novel method to collect sensing data. While crowdsensing has the ad vantages of low cost, high convenience, high speed and so on compared with the traditional fixed sensor network, there exist some problems, among which the problem of data quality is particularly significant.

The existing solutions in crowdsensing for the quality problem commonly include two kinds. One is prejudgment-type [1, 6, 11], that is, the central platform select workers who may submit higher-quality data to do the tasks after task publishing. The other kind is incentive-type [2, 9]. By reasonably setting up the incentives, using reverse auction or other related models, the central platform tries to pay the workers according to their contribution, so as to improve the quality of task completion.

However, firstly, the realization of these technologies is all based on the existence of a fully trusted central platform, which is impossible in reality. The possibility that the central platform scams users exists. Also if the reliability of the platform can not be guaranteed, the malicious attacks and tampering can damage the interests of users. Meanwhile, both kinds of methods above require reasonable evaluation of sensing data in fact while the data is difficult to evaluate in crowdsensing.

To solve the problems mentioned above, the idea of blockchain is introduced into crowdsensing and the quality grading evaluation (QGE) is proposed. Blockchain [8, 12] is a popular concept nowadays that has been widely used in bitcoin and other fields. The social characteristics of human as a participant in crowdsensing make it necessary to make a change if blockchain idea is applied to this scenario. Thus, our paper mainly focuses on how to apply the idea of blockchain to crowdsensing. And for quality control, truth discovery and fuzzy theories are used in QGE to generate reasonable evaluation criterion for sensing data and the garbled circuit is constructed to prevent cheating.

Blockchain-based Crowdsensing Quality Control Model

In the model proposed in this paper, there are three roles: verifier, task publisher and workers. The credit-based verifier selection mechanism is proposed. As shown in Figure 1, the task publisher broadcasts the task information including the basic requirements of task data such as numerical precision and the range of data and so on. After the other nodes receive this broadcast, if it wants the work, the node broadcasts its own information. Then the verifier selects the workers according to the willing nodes' information and replies them. After completing the task, the worker encrypts its sensing data using the public key of the task publisher, attaches its digital signature and the task information, generates a hash digest of the information and broadcasts. When the verifier receives the broadcast, it verifies whether the node is a worker of the task and whether it is acquired before deadline. And by the deadline, the verifier sends all the verified data to the task publisher and generates a Merkle Tree using the

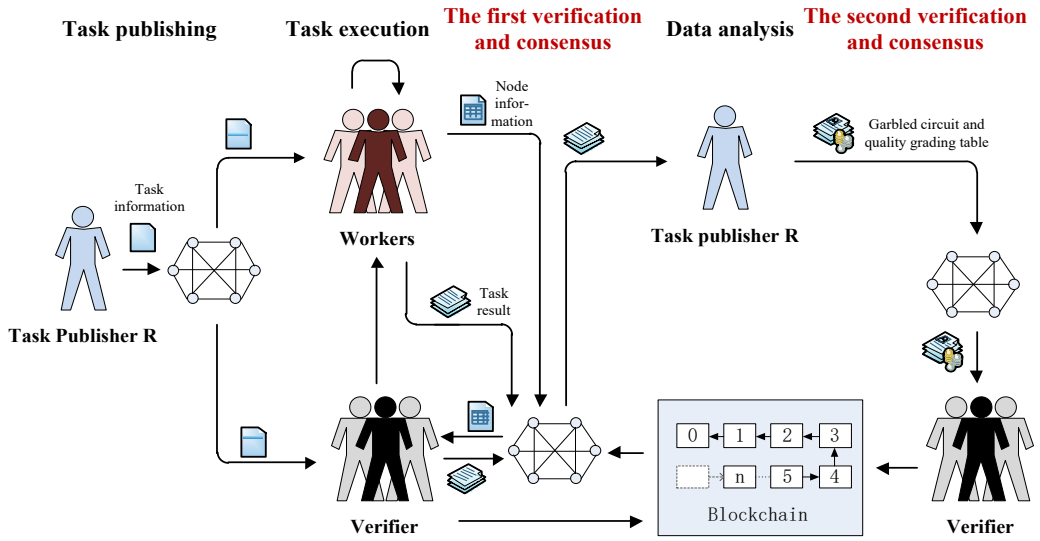


Figure 1: The blockchain-based crowdsensing quality control model

hash digests of these data. The Merkle Tree is broadcast and the first consensus is conducted. The task publisher generates a garbled circuit and a quality grading table and broadcasts them. When the verifier receives them, it inputs the hash digest in the Merkle Tree into the garbled circuit to get the quality level of each data, and then compares the quality level with the quality grading table to determine the reward of each worker. The verifier generates a Merkle Tree using the information about the worker identity and their rewards, and combines it with the Merkle Tree generated in the first consensus to form a new Merkle Tree and generate a new block which is joined into the blockchain.

Details in the Model

The Credit-based Verifier Selection Mechanism

First, a group of nodes with high credit is selected. The node with the highest credit becomes the verifier within the time threshold to generate block. If it is times out, or the node completes the generation, another node in the group becomes the new verifier to generate a block. After generating n blocks, the credit of the nodes of the whole network is recalculated and a new group of candidate verifier nodes is selected to generate blocks. The calculation of credit includes two aspects: the quality of historical task completion QoH and the verification success rate VsR. Considering the time decay in QoH, the exponential decay and time window

IF	And	Then
satisfactory of basic requirements	correctness	grading
H	H	A
H	M	B
H	L	C
H	Er	E
M	H	B
M	M	C
M	L	D
M	Er	E
L	H	C
L	M	D
L	L	E
L	Er	E

Table 1: Fuzzy rules for determining the quality level of the result

are used, which is as follows.

$$QoH = baseQ + \sum_{i=1}^{n}\left(e^{-\lambda_i} * \frac{1}{m_i}\sum_{j=1}^{m_i}\left(Q_{i,j} - baseQ\right)\right) \quad (1)$$

where, $baseQ$ denotes the basic value of QoH, n denotes the number of time windows, m_i and $Q_{i,j}$ denote the number of completed tasks and the jth task completion quality during the ith time window respectively, and λ_i denotes the lower bound of the ith time window. The average quality increment of each time window is calculated, and then multiplied by the amount of time decay. Finally the attenuated increment of each time window is added to the $baseQ$ to obtain QoH.

Using the same idea in QoH, the calculation of VsR is as follows.

$$VsR = baseV + \sum_{i=1}^{n}\left(e^{-\lambda_i} * \frac{Num_i^{cor}}{Num_i^{total}}\right) \quad (2)$$

where VsR denotes the verification correct rate, $baseV$ denotes the basic value of VsR, Num_{cor}^{i} and Num_{total}^{i} denote the frequency of the correct verification and the total number of verifications by the node in the ith time window respectively.

In summary, the credit Cre is calculated as follows:

$$Cre = QoH * \alpha + VsR * (1-\alpha) \quad (3)$$

where α denotes the proportion of QoH in credit.

Quality Grading Evaluation (QGE)

The quality grading evaluation (QGE) is proposed for data evaluation in this paper. First, the task publisher divides the data into three sets, H (high), M (medium) and L (low), according to the basic requirements in the task information using the idea of fuzzy mathematics. Then, the idea of truth discovery is used to evaluate the correctness of the sensing data. The similarity degree among the data is calculated. Data with similarity degree within a given requirement by task publisher is treated as the same set, and the data in the same set is considered to be the same. The turnout rate VR for each set is calculated as follows:

$$VR(v) = \frac{Vote(v)}{Vote} \quad (4)$$

Where $Vote(v)$ denotes the number of data in the set v, $Vote$ denotes the total amount of sensing data.

The correct probability calculation of data based on the conditional probability Bayesian formula is as follows:

$$\begin{aligned} P(\psi(o) \mid v \text{ true}) &= \prod_{S \in S_o(v)} Cre_S \cdot \prod_{S \in S_o - S_o(v)} \frac{1 - Cre_S}{n} \\ P(\psi(o)) &= \sum_{v \in V(o)} P(\psi(o) \mid v \text{ true}) \cdot P(v \text{ true}) \\ P(v) = P(v \text{ true} \mid \psi(o)) &= \frac{\prod_{S \in S_o(v)} \frac{nCre_S}{1 - Cre_S}}{\sum_{v_0 \in V(o)} \prod_{S \in S_o(v_0)} \frac{nCre_S}{1 - Cre_S}} \end{aligned} \quad (5)$$

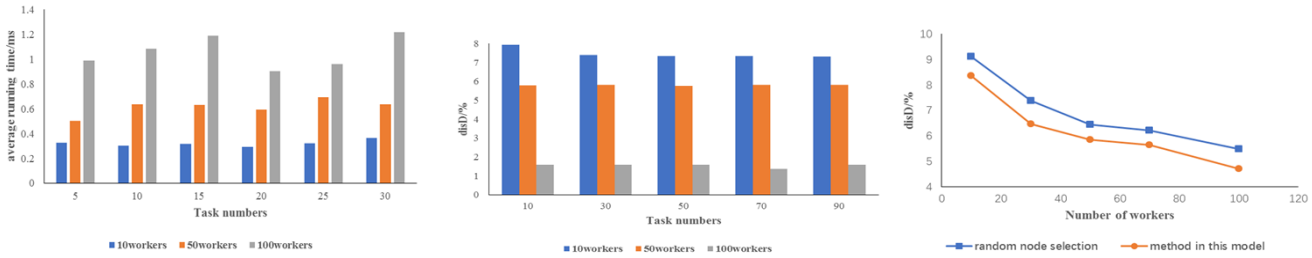
Where $\psi(o)$ denotes the data value space for the requested object o in the task, n denotes the total number of data values, S_o denotes the worker set that provides the data value, and $S_o(v)$ denotes the worker set that provides the data value v to o, Cre_S is the credit of worker S. According to the formulas, the correct possibility of each data is calculated using the credit of workers in this data set.

Thus, the correctness degree is calculated as follows.

$$C(v) = \frac{P(v) + VR(v)}{2} \quad (6)$$

Where $C(v)$ denotes the correctness degree of v.

Then, the data is divided into three sets again according to $C(v)$ using the idea of fuzzy mathematics. The list



(a) The running time of block generation (b) Different number of tasks and workers (c)Adding malicious nodes

Figure 2: The experiment results

of fuzzy rules generated is as Table 1. The reward for each level is as follows.

$$Reward_i = \lambda_i * \frac{Bud}{n},\ i = A, B, C, D\ or\ E \tag{7}$$

where Bud denotes the budget provided in the task information, n denotes the number of workers, and $\lambda_i \in [0,1]$ & $\lambda_E \le \lambda_D \le \lambda_C \le \lambda_B \le \lambda_A$. λ_E is often set to be zero.

A quality grading table is generated to show the relationship between the quality level and reward and a garbled circuit is constructed according to the relationship between the hash digest and the quality level. Yao's garbled circuit [10] is a technique proposed by Andrew Chi-Chih Yao in 1980s. The garbled circuit is constructed for quality assessment, so that the criteria for perceived data are not Leak out, avoiding cheating.

Performance evaluation

In order to verify the performance of the model we set up an experimental environment based on Ethereum, and used the malic acid content attribute in the UCI standard data set of wine [3] as the sensing data set to simulate the entire process of crowdsensing.

The Running Time of Block Generation

As shown in the figure 2 (a), with the increase in the number of users, the average time for generating blocks was increase, but all remain at the millisecond level. The running time of block generation is mainly affected by the number of workers in the task rather than the number of tasks.

The Data Deviation

In the experiments, data deviation $disD$ is a measure of the degree of deviation between the data submitted by the user and the correct data value in the simulation, which is calculated as follows.

$$disD = \frac{|data - answer|}{answer} \tag{8}$$

where $data$ denotes the task data submitted by the workers and $answer$ denotes the correct data in the experiments. As can be seen from the figure 2 (b), as the number of workers in each task increases, the distance appears a clear downward trend. In addition, we randomly added 50 malicious nodes in the experiment and submitted error data. As figure 2 (c) shows, after the quality control in this paper, although the $disD$ is still slightly higher than the condition before

the malicious node joins, the magnitude of change is not large, and com-pareing with the method of randomly node selection, it has a significantly better effect.

Conclusions

To solve the problems resulting by the central platform, the model based on blockchain is proposed. It achieves the centerless, irrevocable and non-repudiation, avoiding the harm of the attacks and frauds through twice consensuses mechanism. Meanwhile, this model realizes quality control through the quality grading evaluation (QGE) conducted by the task publisher during the analysis of result data which using the idea of truth discovery and fuzzy theories. It enables task publisher to obtain high-quality task data and workers to receive reasonable reward according to contribution.

Acknowledgments

This work was supported by the NSFC (61472316, 61502380), Science and Technology Program of Shenzhen (JCYJ20170816100939373).

References

1. An J., Gui X., Wang Z., Yang J., & He X. (2015). A crowdsourcing assignment model based on mobile crowd sensing in the internet of things. *IEEE Internet of Things Journal*, 2(5), 358-369.
2. van Berkel N., Goncalves J., Hosio S., and Kostakos V. (2017). Gamification of mobile experience sampling improves data quality and quantity. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3), 107.
3. Dua, D. and Karra Taniskidou, E. (2017). UCI Machine Learning Repository. from http://archive.ics.uci.edu/ml
4. Ganti, R. K., Ye F., and Lei H. (2011). Mobile crowdsensing: current state and future challenges. *IEEE Communications Magazine*, 49(11), 32-39.
5. Guo B., Wang Z., Yu Z., Wang Y., Neil Y. Yen, Runhe H., and Zhou X. (2015). Mobile crowd sensing and computing: the review of an emerging human-powered sensing paradigm. *Acm Computing Surveys*, 48(1), 7.
6. Goncalves J., Hosio S., Van Berkel N., Ahmed F., and Kostakos, V. (2017). CrowdPickUp: Crowdsourcing Task Pickup in the Wild. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3), 51.
7. Lee, C. C. (1990). Fuzzy logic in control systems: fuzzy logic controller. i. *IEEE Trans Smc*, 20(2), 404-418.
8. Pass R., Seeman L., and Shelat A. (2017). Analysis of the Blockchain Protocol in Asynchronous Networks. *International Conference on the Theory and Applications of Cryptographic Techniques* (pp.643-673). Springer, Cham.
9. Song B., Shah-Mansouri H., and Wong, V. W. S. (2017). Quality of sensing aware budget feasible mechanism for mobile crowdsensing. *IEEE Transactions on Wireless Communications*, PP(99), 1-1.
10. Yao, A. C. (1982). Protocols for secure computation. *Focs*, 160-164.
11. Zhang D., Xiong H., Wang L., and Chen G. (2014, September). CrowdRecruiter: selecting participants for piggyback crowdsensing under probabilistic coverage constraint. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (pp. 703-714). ACM.
12. Yuan Y., and Wang, F. Y. (2016). Blockchain: the state of the art and future trends. *Acta Automatica Sinica.*