Hindawi
Mobile Information Systems
Volume 2019, Article ID 6217839, 1 page
https://doi.org/10.1155/2019/6217839

Retraction

Retracted: Spatial Crowdsourcing Quality Control Model Based on K-Anonymity Location Privacy Protection and ELM Spammer Detection

Mobile Information Systems

Received 15 May 2019; Accepted 15 May 2019; Published 27 June 2019

At the request of the authors, the article titled "Spatial Crowdsourcing Quality Control Model Based on K-Anonymity Location Privacy Protection and ELM Spammer Detection" [1] has been retracted. The authors found in a recent experiment that a serious error was caused by the server hardware failure, so the paper was written based on incorrect data and there was a big deviation in the argument.

References

[1] M. Zeng, Z. Cheng, X. Huang, and B. Zheng, "Spatial crowdsourcing quality control model based on K-anonymity location privacy protection and ELM spammer detection," *Mobile Information Systems*, vol. 2019, Article ID 2723686, 10 pages, 2019.

Hindawi
Mobile Information Systems
Volume 2019, Article ID 2723686, 10 pages
https://doi.org/10.1155/2019/2723686

Research Article

Spatial Crowdsourcing Quality Control Model Based on K-Anonymity Location Privacy Protection and ELM Spammer Detection

Mengjia Zeng,[1] **Zhaolin Cheng**,[2] **Xu Huang**,[3] **and Bo Zheng**[3]

[1]*School of Information Engineering, Huzhou University, Qiuzhen School of Huzhou Teachers College, Huzhou, Zhejiang, China*
[2]*School of Business, Huzhou University, Huzhou, Zhejiang, China*
[3]*School of Information Engineering, Huzhou University, Huzhou, Zhejiang, China*

Correspondence should be addressed to Zhaolin Cheng; 58754052@qq.com

Received 20 September 2018; Revised 4 November 2018; Accepted 26 December 2018; Published 4 February 2019

Guest Editor: Wolfram Luther

The spatial crowdsourcing task places workers at a risk of privacy leakage. If positional information is not required to submit, it will result in an increased error rate and number of spammers, which together affects the quality of spatial crowdsourcing. In this paper, a spatial crowdsourcing quality control model is proposed, called SCQCM. In the model, the spatial k-anonymity algorithm is used to protect the position privacy of the general spatial crowdsourcing workers. Next, an ELM (extreme learning machine) algorithm is used to detect spammers, while an EM (expectation maximization) algorithm is used to estimate the error rate. Finally, different parameters are selected, and the efficiency of the model is simulated. The results showed that the spatial crowdsourcing model proposed in this paper guaranteed the quality of crowdsourcing projects on the premise of protecting the privacy of workers.

1. Introduction

Computer-supported collaborative work is an exciting research field [1], wherein "crowdsourcing" is an important topic. Crowdsourcing was first proposed in June 2006 by Jeff Howe, a journalist from the United States Magazine Connection, which he defined as a type of a working model where a company or agency outsources the work of a hired employee or a full-time outsourced person to a nonfull-time group via an open network platform. Crowdsourcing tasks are usually done voluntarily by individuals or groups of people. The key of crowdsourcing is to make full use of the labor resources of the open network platform to accomplish simple or complex tasks [2]. As a successful model that makes full use of group intelligence, crowdsourcing has been widely used for tasks such as picture tagging, natural language comprehension, market prediction, and view mining. In recent years, crowdsourcing has received extensive attention in the fields of translation, logistics, transportation, and lodging and has gradually become a new research hotspot. However, the future of crowdsourcing faces many theoretical and practical challenges.

With the improvement of mobile internet technology and the computing and sensing abilities of mobile devices, a crowdsourcing form of these technologies based on user location information has become popular. Kazemi and Cyrus [3] calls this kind of crowdsourcing as "spatial crowdsourcing" (SC), whose tasks are mainly related to space and location. As a special form of crowdsourcing, SC has become a new research topic in academic circles (What TaskRabbit Offers [EB/OL] (2017-08-25) [2017-08-28]; https://support.taskrabbit.com/hc/en-us/articles/204411410-What-TaskRabbit-Offers) and industry [4]. Typical SC is achieved via a crowdsourcing platform that assigns tasks to nearby workers, who in turn move to the designated locations to complete the assigned spatial tasks. Through this kind of crowdsourcing, people can make better use of swarm intelligence to accomplish simple or complex spatial tasks. Although spatial crowdsourcing makes

full use of swarm intelligence and brings great benefits, the construction and promotion of crowdsourcing platforms are not easy. A crowdsourcing platform releases and allocates spatial tasks according to the location information submitted by the user, which will include sensitive information [5], such as the identity of the user, their home address, their health status, and living habits.

In recent years, smart mobile phones have been used as multimode sensors that collect and share various types of data, including pictures, video, location, moving speed, direction, and acceleration. Therefore, crowdsourcing platforms can obtain considerable amounts of user location data through smart mobile phones, which may lead to a leakage of sensitive information and seriously threaten the users' privacy. For example, in July 2018, the poor management of the website http://datatang.com resulted in a tremendous infringement of personal information privacy. In eight months, the http://datatang.com website used spatial crowdsourcing to transmit personal information at an average of 130 million items per day and a total cumulative transmission of compressed data of about 4000 GB, including highly private data. The problem of user information security in terms of spatial crowdsourcing has become an urgent problem in theory and practice.

Crowdsourcing information sharing is a double-edged sword. On the one hand, crowdsourcing information sharing can ensure the smooth development of work and prevent dishonest cheating workers [6] and spammers from making money. On the other hand, crowdsourcing information sharing requires location information of the workers, which not only threatens the privacy of the workers but also affects their enthusiasm for work, especially if they are worried about the leakage of their private information. How to effectively achieve a balance between privacy protection and quality control has become a difficult problem in spatial crowdsourcing, and it is a blind spot in the existing literature.

At present, scholars have done a lot of research on the prevention of privacy leakage. In 2002, Sweeney [7] put forward the K-anonymity privacy protection technology to solve the problem of personal and sensitive data leakage. On this basis, additional researchers further proposed a number of improved algorithms, such as the L-diversity method [8], the t-closeness method [9, 10], (α, k) anonymity algorithms [11], and the ε−cloning [12] method, which can better prevent privacy disclosure when publishing data sets. However, the above methods are often targeted as static data sets; that is, all data are published only once, and no data updates are made after publication. The location information in spatial crowdsourcing scenarios change with any change in the platform tasks, which demonstrate the dynamic characteristics of continuous publication. Hu et al. [13] studied the spatial crowdsourcing location privacy protection in a P2P communication environment and implemented spatial crowdsourcing worker location privacy protection using a peer-to-peer spatial k-anonymity algorithm [14]. Their method solved the problem that it is not considering the spatial domain attributes of each crowdsourcing worker in the differential privacy space decomposition method studied in [15]. Vu et al. [16] proposes a privacy protection mechanism based on a locally sensitive hash [17], which protected the user's identity and location information in participatory perception scenarios. The location privacy protection based on differential k-anonymity proposed by Wang et al. [18] can resist persistent and background-based attacks. A definition of spatial crowdsourcing location k-anonymity was given by An et al. [19]. All the above studies have proved that K-anonymity algorithms can solve the privacy leakage problem in spatial crowdsourcing scenarios. However, the focus of the above research was only on privacy leakage, and the quality control of crowdsourcing was not considered.

Varshney et al. [20, 21] use two different schemes based on the random noise method to prevent publishers' privacy from being attacked by multiple workers. Hiroshi et al. [22] proposed a privacy protection protocol based on decentralized computing to ensure workers' privacy under the premise of quality control. The above literature only considered the balance between privacy protection and quality control, while the issue of publisher privacy protection is considered in the current study. What we aim to find is a balance between privacy protection and the quality control of workers in SC.

To sum up, in practical application scenarios of SC, workers need to submit their own location information to the crowdsourcing service platform, which has the risk of privacy leakages. However, the existence of errors arising from the normal crowdsourcing workers and any deceptive workers/spammers has led to a quality problem in crowdsourcing services. Our aim was to protect the location privacy of crowdsourcing workers, identify and exclude spammers, and reduce the error rate to ensure crowdsourcing quality control.

This article is structured as follows. In Section 2, we give the complete definition of our proposed SC anonymity technology and privacy protection models based on spatial anonymity technology and introduce the process of spammer identification with an ELM algorithm. In Section 3, we introduce our experiments, and in Section 4, we analyze the results. Finally, in Section 5, we summarize our study.

2. Problem Solving Ideas

First, a description of the crowdsourcing quality control problem in the SC scenario is given, and a solution to the balance between location privacy protection and crowdsourcing quality control is given. Then, a complete definition of the used privacy protection model based on spatial anonymity technology and the principle of spammer recognition via ELM [23] are given.

2.1. Problem Description. Consider a typical crowdsourcing scenario: one-task publishers (requester) publish m tasks $T_i\,(1 \le i \le m)$, where n workers (worker) $W_j\,(1 \le j \le n)$ are involved in the task completion. Each task is completed by n workers, and each worker completes m tasks; however,

a single task is completed by only one worker. The matrix $V_{m\times n} = \{v_{ij}\}$ represents all the task results submitted by the workers, and $V_{m\times 1} = \{v_i\}$ represents the correct result of each task. To simplify the problem, we assume that T_i is a two-tuple problem, and W_j only needs to answer "yes" ($v_{ij} = 1$) or "no" ($v_{ij} = 0$). The conclusion of the two-tuple model is not difficult to expand and apply to other task types [22]. The quality control problem of crowdsourcing is how requesters can deduce the correct result, $V_{m\times 1}$, of all tasks according to the result, $V_{m\times n}$, submitted by the workers. In the process of crowdsourcing quality control, there are at least two types of crowdsourcing quality disturbance factors: deceptive workers, called spammers, and the worker error rate, η. In order to maximize the benefits per unit time, spammers will not seriously submit low quality task results, and even diligent and conscientious workers may submit incorrect results at a certain error rate. Therefore, to carry out crowdsourcing quality control, we need to exclude spammers and reduce the error rate.

2.2. Solutions. For the privacy protection of spatial crowdsourcing, in this paper, we give a complete definition and workflow of a privacy protection model of spatial anonymity technology based on the method presented in [23]. In order to gain greater pay, spammer-type workers will submit the most amount of information in the shortest space of time. However, the number of submissions, time changes, and other parameters of ordinary workers will show different characteristics. According to these characteristics, machine-learning algorithms can achieve the purpose of identifying spammers. An extreme learning machine (ELM) is a fast, single, hidden layer feedforward neural network training algorithm, which is faster than the traditional neural networks under the premise of ensuring good accuracy [24]. Traditional neural network learning algorithms (such as the BP (back propagation) algorithm) need to set a large number of network training parameters artificially, and they easily fall into local optima. The ELM algorithm only needs to set the number of hidden layer nodes of the network, and it does not need to adjust the input weights of the network and the biases of the hidden layer units in the process of algorithm execution; these conspire to produce a unique optimal solution. Therefore, the ELM algorithm has the advantages of fast learning speed and good generalization performance, and it is used in this paper to identify spammers.

For the problem of the worker error rate, the EM (expectation maximization) algorithm [25, 26] is used to estimate worker errors. Firstly, the correct rate (correct rate + error rate = 1) is used as the correct weight estimation for each task, and the specific implementation is to assign the same task to multiple workers who complete the task independently. Then, we take the majority of the results as the correct result and update the error rate estimation of each worker with it. Next, the error rate of multiple workers is estimated with a maximum likelihood method, and the two steps of E-step (Expectation step) and M-step (maximization step) are repeated until the result converges.

Based on the above ideas, this paper proposes a Spatial Crowdsourcing Quality Control Model (SCQCM) to solve the balance between location privacy protection and cheating-worker screening and error rate estimation.

2.3. Privacy Protection Model Based on Spatial Anonymity Technology. First, the basic concept of the spatial anonymity technology is introduced, and the workflow of our SC platform is given. Then, the k-anonymity and privacy of spatial crowdsourcing location are defined. Finally, a privacy protection model based on the spatial anonymity technology is given.

2.3.1. Basic Concepts. The following steps and terms are defined here:

(1) *The task requester* [27], in short, is called as "requester."

A requester first registers on the crowdsourcing platform, whereby it performs a series of tasks related to designing and releasing of spatial tasks, refusing or receiving the results from the workers, and collating the results. A requester is often defined as $R = \langle L_R, T_R \rangle$, where L_R represents the location information of the requester and T_R represents the task that the requester releases.

(2) *Spatial tasks* [2, 27]. A spatial task is usually a special task that has geographical location and time attributes. It is generally defined as a four tuple: $T = \langle L_T, t_{\text{begin}}, t_{\text{end}}, P_T \rangle$, where L_T represents the location of the spatial task, t_{begin} represents the release time of the spatial task, t_{end} represents the cutoff time of the spatial task, and P_T represents the reward for the completion of the task.

(3) *Spatial crowdsourcing worker*, in short, is called as "worker" [2, 27]. The workers are the mobile device users who perform the spatial task(s). They can select a spatial task, accept the task, submit positional information, and submit the result by registering on the crowdsourcing platform. A worker is usually defined as a three-tuple: $W = \langle L_W, R_W, T_{\max} \rangle$, where L_W represents the current location information of the worker, R_W indicates the spatial domain that the worker can accept, and $T_{\max}$ represents the maximum number of tasks that the worker can accept in the spatial domain, R_W.

(4) *Spatial crowdsourcing*. The complete SC includes task requesters, SC tasks, SC platforms, and SC workers. Spatial crowdsourcing usually refers to the process where a requester designs a SC task and publishes it to the SC platform. In turn, the SC platform realizes the task assignment, and the workers accept and complete the spatial task at a designated place. The basic spatial crowdsourcing model is shown in Figure 1.

2.3.2. Work Flow. As the core of SC, the SC platform establishes a cooperative relationship between the requester and the worker based on the spatial task, which is

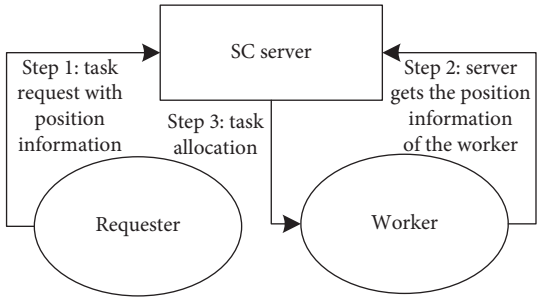

Figure 1: Basic model of spatial crowdsourcing (SC).

responsible for comprehensively processing the task and/or individual position information submitted by the requester and the worker. Figure 2 shows the SC workflow. In general, the SC platform first collects the task information from the requester and the location information from the worker. The information is preprocessed by the data processing module who then submits a request to the task allocation module, which then completes the task allocation. Finally, the spatial tasks are completed by the worker, and the results are submitted to the quality control module.

According to the allocation of spatial tasks, SC can be divided into two operating modes: WSTs (Worker Selected Tasks; worker selection modes) and SATs (Server Assigned Tasks; server assignment task modes). First, let us consider the WST mode workflow: crowdsourcing workers take the initiative to find tasks released on the platform according to their own spatial location information and choose the appropriate spatial tasks to perform. Next, in the SAT mode workflow, workers first submit their spatial location information to the platform. The data process module matches the position information of the worker with the task, and if it is matched, it allocates a task to the worker. Then, the workers decide whether to accept the assigned task. As the task selection in the WST mode is done by the crowdsourcing workers themselves, they do not need to upload their location information: hence, this mode is not considered in this paper. Instead, in this article, we only analyze SAT patterns.

2.3.3. Spatial Crowdsourcing Location K-Anonymity. In SC, the location attribute of a worker is a quasi-identifier. In an anonymous spatial area, the location of any worker cannot be distinguished from the location of at least $k-1$ workers. Among them, the quasi-identifier is the minimum attribute set [28], which combines other external information and identifies the target's location with a high probability. As shown in Figure 3, the real location of a spatial crowdsourcing worker is L, and then the location point L is extended to a hidden area R to replace the exact location information of the worker. In this anonymous spatial area, every worker is hidden in at least $k-1$ workers, which mean any attacker can only judge the number of workers in the hidden area, but they cannot determine their exact locations. This approach gives a certain degree of privacy protection to the workers.

2.3.4. θ Privacy of Location L. $P(L_t)$ represents the probability that a user is at position L at time t, $L_t^{\leftarrow}$ represents the location data that the attacker has collected before t, and θ is the maximum attack effect expended by an attacker:

$$P\left(L_t \mid L_t^{\leftarrow}\right) - P\left(L_t\right) \leq \theta. \quad (1)$$

2.3.5. Privacy Protection Model Based on Spatial Anonymity Technology. Our privacy-preserving model based on spatial anonymity is illustrated in Figure 4. The crowdsourcing platform is a trusted third party. First, the location privacy policy of a worker (activity 3 in Figure 4) is formulated according to the task released by the requester. Then, the platform blurs (i.e., "fuzzifies") the submitted position using k-anonymity (activity 5) and transfers the protected location information to the requester (activity 6). Figure 5 shows a spatial crowdsourcing task map. The m task location $L_i\,(1 \leq i \leq m)$ is distributed in different locations (L_i is the correct execution location of task T_i). Using a Voronoi graph as the initial point set of the task point, the map is divided into m regions, $R_i\,(1 \leq i \leq m)$, which satisfy the condition that, for any point, L, within the region, R_i, L_i, is the nearest task point, that is,

$$\left|L - L_i\right| \leq \left|L - L_l\right|, \quad \forall L\, \varepsilon\, R_i,\ l = 1, 2, \ldots, m. \quad (2)$$

Suppose a worker completes the P_i point task and submits the results before leaving A_i. Using the information entropy to measure the degree of crowdsourcing system privacy protection, the greater the information entropy is, the greater the uncertainty of the position of the worker is, and the higher their degree of protection is. The position information entropy W_j at t time is as follows:

$$I_t\left(W_j\right) = -\sum_{i=1}^{m} L\left\{W_j \text{ lies in } \mathrm{R}_i\right\} \log L\left\{W_j \text{ lies in } \mathrm{R}_i\right\}. \quad (3)$$

2.4. Using the ELM to Discriminate Spammers. The ELM learning process consists of two steps: first, (1) random feature mapping. Here, the ELM generates input weights randomly and initializes implicit layer unit biases and maps input vectors to the feature space using nonlinear mapping functions; and (2) a solution of linear parameters, where the ELM model is used to solve the output.

For a data set with N number of examples, (x_i, t_i) satisfies $x_i = [x_{i1}, x_{i2}, \ldots, x_{iN}]^T \in R^N$ and $t_i = [t_{i1}, t_{i2}, \ldots, t_{iM}]^T \in R^M$. There are L number of hidden layer nodes, and the activation function is $g(x)$. The single hidden layer neural network then can be described as

$$\sum_{i=1}^{L} \beta_i g\left(w_i \cdot X_j + b_i\right) = o_j, \quad j = 1, \ldots, N, \quad (4)$$

where $w_i = [w_{i1}, w_{i2}, \ldots, w_{iN}]^T$ is the input weight vector, $\beta = [\beta_1, \beta_2, \ldots, \beta_L]^T$ is the output weight vector between the hidden layer nodes and output nodes, $b = [b_1, b_2, \ldots, b_L]^T$ is the bias vector of the hidden layer, b_i is the bias of the hidden layer unit i, and $w_i \cdot X_j$ is the inner product of w_i and X_j.

The structure of the ELM algorithm is shown in Figure 6.

The matrix expression of equation (4) is

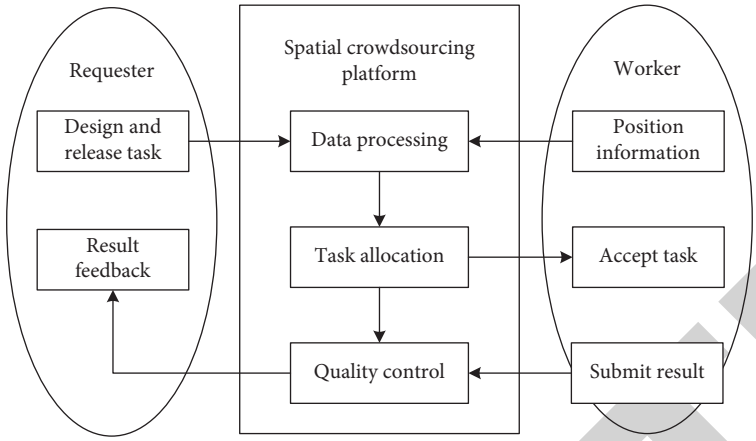

Figure 2: SC workflow.

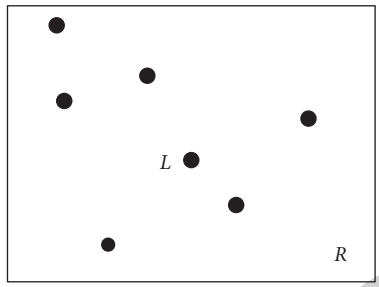

Figure 3: Spatial K-anonymity scheme.

$$H\beta = T,$$

$$H = \begin{bmatrix} g(w_1 \cdot X_1 + b_1) & \cdots & g(w_L \cdot X_1 + b_L) \\ \vdots & \ddots & \vdots \\ g(w_1 \cdot X_N + b_1) & \cdots & g(w_L \cdot X_N + b_L) \end{bmatrix},$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L \times m}, \tag{5}$$

$$T = \begin{bmatrix} T_1^T \\ \vdots \\ T_N^T \end{bmatrix}_{N \times m},$$

where H is the output of the hidden layer nodes, β is the output weight, and T is the expected output.

The training objective of ELM is to minimize the output error, that is,

$$\sum_{j=1}^{N} \left\| o_j - t_j \right\| = 0. \tag{6}$$

t_j is a sample of T. For β_i, w_i, and b_i,

$$\sum_{i=1}^{L} \beta_i g\left(w_i \cdot X_j + b_i\right) = t_j, \quad j = 1, \ldots, N. \tag{7}$$

Solving $\widehat{\beta}_i$, $\widehat{w}_i$, and $\widehat{b}_i$, we find

$$\left\| H\left(\widehat{w}_i, \widehat{b}_i\right)\widehat{\beta}_i - T \right\| = \min_{w,b,\beta} \left\| H\left(w_i, b_i\right)\beta_i - T \right\|, \quad i = 1, 2, \ldots, L, \tag{8}$$

which is equivalent to the optimal loss function:

$$E = \sum_{j=1}^{N} \left(\sum_{i=1}^{L} \beta_i g\left(w_i \cdot X_j + b_i\right) - t_j \right)^2. \tag{9}$$

In the ELM algorithm, the input weight w_i and the hidden layer bias b_i are selected randomly during training. When the activation function is infinitely differentiable and the number of hidden layer nodes is large enough, the ELM can approximate any continuous function. According to the values of W_i and b_i, only the determined output matrix H is calculated. The training single hidden layer neural network is converted to a least squares solution that is solved as a linear system, $H\beta = T$, whose solution is

$$\widehat{\beta} = H^{+}T = \left(H^T H\right)^{-1} H^T H. \tag{10}$$

In equation (10), H^+ is the generalized inverse matrix of the output matrix H.

The ELM learning algorithm is mainly implemented by the following steps:

(1) Determine the number of hidden layer units and then randomly generate the input weight w_i and hidden layer offset b_i

(2) Select an infinitely differentiable function as the activation function of the hidden layer element and then the output matrix H is obtained

(3) Calculate the output weight β_i according to the output matrix H

(4) The output t_j is obtained according to equation (7)

In addition, ELM is widely used in cluster [29], feature selection [30], and other fields.

3. Experiments

We collected approximately 100,000 data points provided by a crowdsourcing platform, each of which include the task

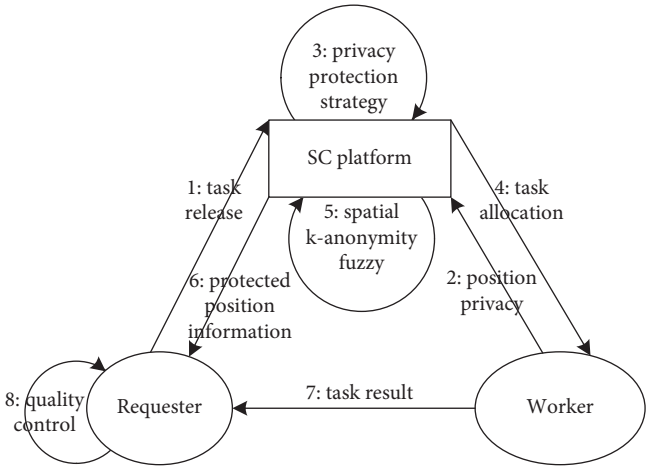

Figure 4: Model of SC based on spatial anonymity privacy protection.

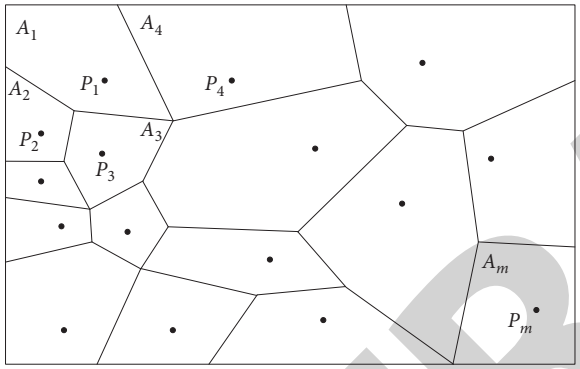

Figure 5: Task map.

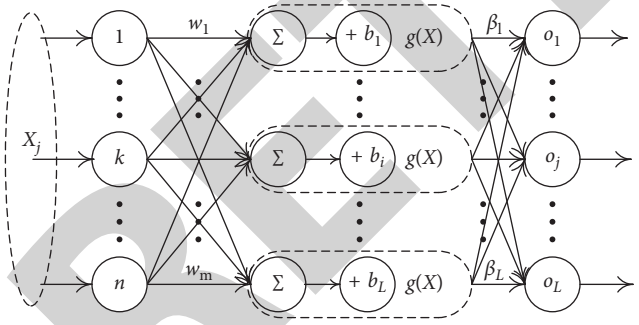

Figure 6: ELM algorithm structure.

number, task name, task position, release time, payment amount, dispatch time, worker's name, worker's position, position of periodicity reported by the worker, time of submission, and so on. For spammer-type workers, because of their desire to quickly end a task and earn a reward, their position changes in the entire order cycle are different from those of ordinary workers. In this paper, these features are used as input to the ELM, which were trained with neural networks to achieve the purpose of identifying spammers.

In this method, a neural network model activation function $g(x)$ selected the "sigmoid" function. The specific steps of the neural network training are as follows: (1) the data of the workers are grouped according to an hour, 24 hours a day as the observation unit, and the user's day behavior is divided into 24 groups. (2) We calculated the reappearance frequency of each group's behavior. Then, (3) according to the distribution of the reappearance frequency in the time series and the duration of the worker's task of establishing the characteristic description of the workers, we formed a temporal behavior matrix, which can be described as

$$\text{Log} = \begin{bmatrix} (t_1, f(a_{11})) & \cdots & (t_N, f(a_{1N})) \\ \vdots & \ddots & \vdots \\ (t_1, f(a_{K1})) & \cdots & (t_N, f(a_{KN})) \end{bmatrix}, \tag{11}$$

where t_i is the duration of a worker's task, $f(a_{ij})$ is the reappearance frequency of worker behavior i in time j, $N = 24$, and K is the number of behavior categories. The reappearance frequency is defined as the ratio of the total number of actions in a certain period to the total number of acts in a day as $f(\cdot) = c_a/c_{\text{all}}$.

Next, we selected all elements about $f(a_{ij})$ from the rows of matrix log and mapped them into a N dimension vector: $TT^{\text{LINE}} = [(t_1, f(a_{X_1 1})), \ldots, (t_i, f(a_{X_i i})), \ldots, (t_N, f(a_{X_N N}))]$, where $f(a_{ij})$ is the maximum of $\{f(a_{1i}), \ldots, f(a_{ki})\}$ and $X_i \in \{1, \ldots, K\}$. This method detects the highest recurrence frequency of specific behavior.

In the ELM algorithm, we used the N dimension vector TT^{LINE} or the $K \times N$ dimension vector TT^{ALL} as the input vector, which was recorded as TT. The question of spammer detection in then inverted into a two-classification problem:

$$\sum_{i=1}^{L} \beta_i g\left(w_i \cdot L_j + b_i\right) = o_j, \quad j = 1, \ldots, N. \tag{12}$$

Among them,

$$g_i(x) = G(a_i, b_i, x), \quad a_i \in R^d, b_i \in R, o_j = [o_{i1}, o_{i2}]^T, \tag{13}$$

where the vector set $\{(W_i, TT_j)\}$ is the training data and $W_j = [W_{j1}, W_{j2}]^T$. $W_{ji} \in \{0, 1\}$ indicates the ordinary worker and spammer $TT_j = TT_j^{\text{LINE}}$ or $TT_j = TT_j^{\text{ALL}}$.

The ELM process of detecting spammers is as follows: (1) analyze the data, group the worker behavior sequence in time, and calculate the appearance frequency of each group. Then, (2) serialize the worker behavior and task length of time and position into the worker information matrix. Next, (3) determine the parameters of the ELM model and use the worker information matrix to train the single hidden layer feedback neural network. Finally, (4) distinguish general workers and spammers.

3.1. The Impact of a Small Number of Error Results on the Overall Results. For a two-element spatial crowdsourcing task, as n workers submit their results, their average rate error is η, It is stipulated as $\eta < 0.5$ here. The requesters use the majority voting method [31] to estimate the correct results, where the correct result is 1, estimated at 0, or the

correct result is 0, estimated at 1. The posteriori probability of error estimation is as follows:

$$P_e = \sum_{i=(n+1)/2}^{n} C_n^i \eta^i (1-\eta)^{n-i}. \quad (14)$$

The posteriori probability of error estimation P_e exponentially declines with the number of workers n. So, when there are more workers to complete a task, P_e tends to 0, where P_e indicates the error probability of a worker submitting a result directly to the requester without using the crowdsourcing platform. Adding Δn wrong results into the n task results ($\Delta n \ll n$), the posterior probability of $n + \Delta n$ task results are estimated to be

$$P_e' = \sum_{i=(n-\Delta n+1)/2}^{n} C_n^i \eta^i (1-\eta)^{n-i}. \quad (15)$$

If we subtract equation (14) from equation (15), we find

$$\Delta P_e = P_e' - P_e = \sum_{i=(n-\Delta n+1)/2}^{(n-1)/2} C_n^i \eta^i (1-\eta)^{n-i} \le \left(\frac{\Delta n}{2} - 1\right) C_n^{(n-1)/2} \eta^{(n-\Delta n+1)/2} (1-\eta)^{n+1/2}. \quad (16)$$

As n increases, the right-hand side of equation (16) tends to zero. The above analysis shows that if a small number of error results are mixed with a high-quality result set, this does not significantly interfere with the final judgment, and it does not significantly affect the accuracy of the estimated result.

3.2. Error Rate and Correct Result Estimation. In this paper, the error rate of workers is taken as a potential variable to estimate the correct result of crowdsourcing tasks via maximum likelihood estimation. The $n-$dimensional vector $\eta_{n\times 1} = \{\eta_j\}$ is the error rate of all workers. A worker W_j completes a task according to a certain error rate η_j, $\eta_j = (\eta_j^1, \eta_j^2)$, where η_j^1 and η_j^2 are independent of each other, which indicates the error rate of W_j when the correct results are "1" and "0," respectively, as follows:

$$\eta_j^1 = P(V_{ij} = 0 \mid \tilde{v}_i = 1), \quad \eta_j^2 = P(V_{ij} = 1 \mid \tilde{v}_i = 0). \quad (17)$$

The expectation maximization algorithm is used to estimate the error rate of the EM algorithm. The specific steps are as follows:

(1) *E-step*: we define the m dimension vector μ, $\mu_i (1 \le i \le m)$, which indicates the posteriori probability of the task T_i' correct result is 1, that is,

$$\mu_i = P(\tilde{v}_i = 1 \mid V_{m\times n}, \eta). \quad (18)$$

Using the correct rate as the weight to the initial value of μ,

$$\mu_i^{(t)} = \frac{pa_i}{pa_i + (1-p)b_i}. \quad (19)$$

Among them, t indicates the tth iteration, p is the expectation probability of task's correct result (which is 1): $a_i = \prod_{j=1}^{n} (1-\eta_j^1)^{V_{ij}} (\eta_j^1)^{1-V_{ij}}$ and $b_i = \prod_{j=1}^{n} (1-\eta_j^2)^{1-V_{ij}} (\eta_j^2)^{V_{ij}}$.

(2) *M-step*: according the expectation value of μ in the E-step, we can estimate the value of p as

$$p = \frac{1}{m} \sum_{i=1}^{m} \mu_i. \quad (20)$$

The maximum likelihood estimation is then calculated, and the estimation of the error rate variable is obtained:

$$\eta_j^1 = 1 - \frac{\sum_{i=1}^{m} \mu_i V_{ij}}{\sum_{i=1}^{m} \mu_i}, \quad \eta_j^2 = 1 - \frac{\sum_{i=1}^{m} (1-\mu_i)(1-V_{ij})}{\sum_{i=1}^{m} (1-\mu_i)}. \quad (21)$$

Next, the Q function is

$$Q(p, \eta_j) = \sum_{i=1}^{m} [\mu_i \log\ pa_i + (1-\mu_i)\log(1-p)b_i]. \quad (22)$$

To judge the model is convergent, we use ε, which is the convergence threshold. ε is an artificial very small value, such as 10^{-6}:

$$\frac{\left|Q\left(p^{(t+1)}, \eta_j^{(t+1)}\right) - Q\left(p^{(t)}, \eta_j^{(t)}\right)\right|}{\left|Q\left(p^{(t)}, \eta_j^{(t)}\right)\right|} < \varepsilon. \quad (23)$$

In every iteration, we calculate Q function, if it makes the inequality (23) true; we think the model is convergent and then return the estimation result μ and end the calculation. Otherwise, we return to the E-step and begin the next iteration.

4. Analysis of the Experimental Results

A series of data sets were generated by changing the task parameters, such as the task number, m, worker number, n, worker error rate, η, spammer ratio, r, and other experimental parameters. The data generation steps included the following: (1) generate the correct result vector $\tilde{v}_{m\times 1}$ of all tasks, where each correct result $\tilde{v}_i$ obeys the Bernoulli distribution of $p = 0.5$, and p is the probability that the correct result of the task is "1." Next, (2) generate the task results of all workers, $W_j (1 \le j \le n)$. If W_j is a spammer, the result $v_{ij} (1 \le i \le m)$ obeys the Bernoulli distribution $B(1, 0.5)$; otherwise, for the task of $\tilde{v}_i = 1$ and $\tilde{v}_i = 0$, v_{ij} obey the Bernoulli distribution $B(1, 1-\eta_j^1)$ and $B(1, 1-\eta_j^2)$ respectively about error rate (η_j^1, η_j^2). Finally, (3) generate the location of each worker. If W_j is a spammer, we select an

area randomly from the m region of Figure 5 as the result submission position. Otherwise, we submit the location area R_i that corresponds to task T_i.

4.1. The Influence of the Fuzzy Coefficient K on Information Entropy and Accuracy. The quality control level of crowdsourcing systems was measured with an accuracy index, where the so-called accuracy rate is the consistency rate between the correct results estimated via statistical methods and the real results. We assumed $\tilde{v}_i = 1$ when the posterior probability of the correct result of task T_i is $\mu_i > 0.5$; otherwise, it is $\tilde{v}_i = 0$. First, the positional information is "fuzzily processed" with the fuzzy coefficient, k. According to equation (3), the average information entropy of the fuzzy coefficient, k, shown in Table 1 can be obtained at different locations. When $k = 1$, it represents a model that does not carry out k-anonymous handling in the spatial crowdsourcing location. A change of k indicates a change of the fuzzy degree. It is not difficult to find that the model proposed in this paper can produce an obvious protection effect if the position information is slightly fuzzy ($k = 6$), and the uncertainty degree of the worker's position is close to half of the case without submitting the position information ($k = m$). The data in Table 1 also show that although the three task publishers obtain different amounts of location information, they produce the same quality control results. The results show that location information may not be helpful for quality control in some real situations. The effect of the error rate and spammers on the results is not considered at this time. The error rate and spammer ratio parameters are considered in the subsequent discussion.

4.2. The Influence of Task Scale, Number of Workers, Error Rate, and Spammer Ratio on the Accuracy Rate. In accordance with the test data set for different parameters, we compared the relationship between the accuracy and other parameters in Figures 7–10 with fuzzy coefficients of $k = 1, 6$, and m. There is a lower error rate $\eta = (0.2, 0.2)$ and spammer ratio $r = 0.2$ in Figures 7 and 8. Regardless of the change in the number of tasks and workers, the quality of the three models is always close. The difference in the two figures is that changing the number of tasks does not affect the model quality. With an increase in the number of workers, the quality of the model has increased. Figures 9 and 10 show that, when η and r are low, the accuracies of the three models are still close. However, with an increase in η and r, the quality control level of the $k = m$ model begins to be significantly worse than the other two models. Moreover, the quality control level of the case when $k = 6$ is always close when $k = 1$. That is, when the error rate and spammer ratio are high, the quality control results are completely different from those without considering spammer and error rate. The experimental results of Figures 7–10 prove that the spatial crowdsourcing privacy protection model with a fuzzy coefficient of $k = 6$ effectively protects the workers' location privacy under the premise of effectively controlling the quality of the crowdsourcing.

Table 1: The influence of the fuzzy coefficient k on the information entropy and accuracy rate.

	$k = 1$	$k = 6$	$k = m$
Average information entropy	0	5.12	10.89
Accuracy rate	0.941	0.942	0.942

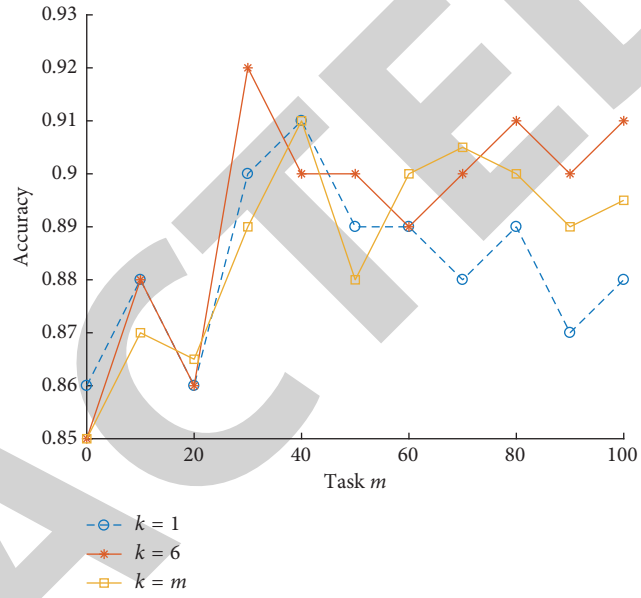

Figure 7: Task number versus accuracy.

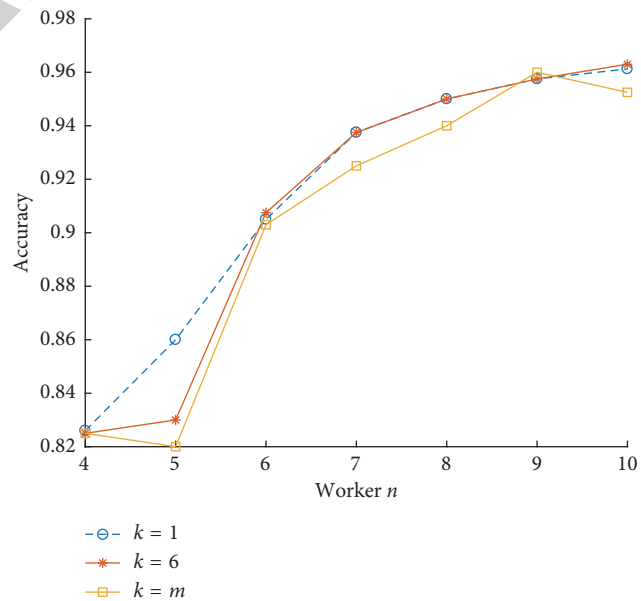

Figure 8: Worker number versus accuracy.

5. Conclusions

The spatial crowdsourcing task results in the leakage risk of the workers' locations privacy. If location information is not required to ensure privacy, this will have the side effect of an increase in the error rate and an increase in the number of

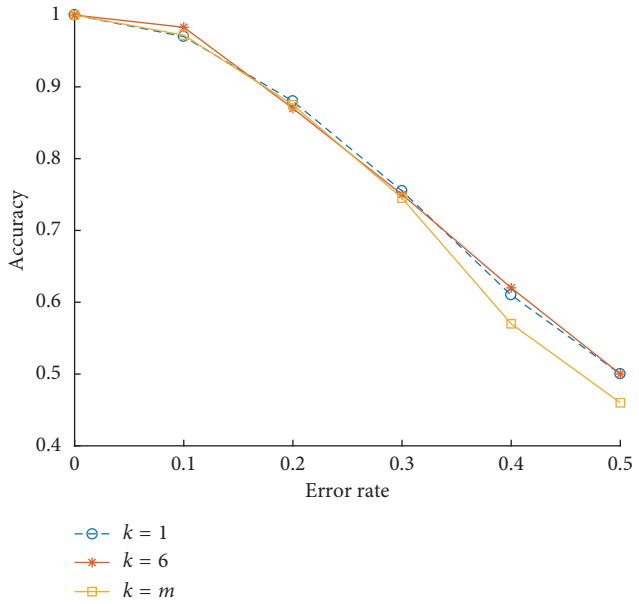

Figure 9: Error rate versus accuracy.

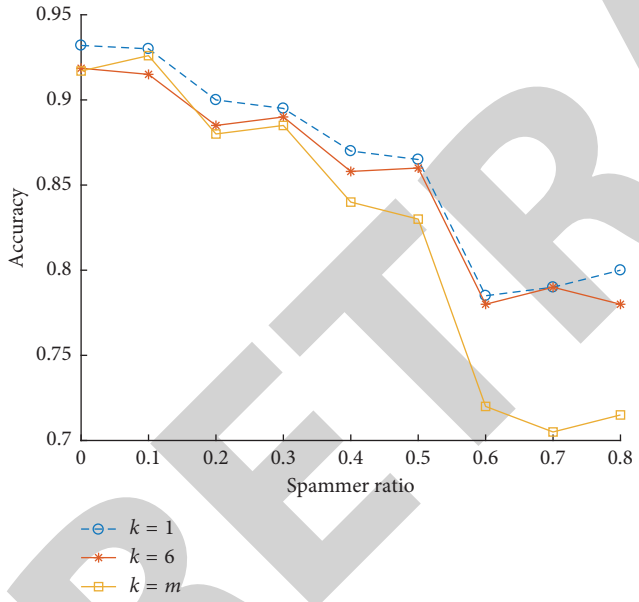

Figure 10: Spammer ratio versus accuracy.

spammers, both of which would affect the quality of the crowdsourcing. In this paper, a SC model is proposed. A spatial k-anonymity algorithm is used to protect the location privacy of the public spatial crowdsourcing workers. Next, an ELM algorithm is used to detect spammers, and an EM algorithm is used to estimate the error rate. Finally, different parameters were selected, and the efficiency of the model was simulated. The results show that the SC model proposed in this paper can guarantee the quality of the crowdsourcing project on the premise of protecting the privacy of the workers.

Aiming at achieving a balance between location privacy protection and crowdsourcing quality control, we proposed a SC quality control model based on spatial k-anonymity and the ELM algorithm for location privacy protection and deception worker screening. The main contributions of this paper are as follows:

(1) On the basis of Wang et al. [18], we provided a definition of SC anonymity technology, a workflow of spatial crowdsourcing platform based on spatial anonymity technology, a definition of spatial crowdsourcing location k-anonymity, and formulae for privacy protection.

(2) We used the ELM algorithm to realize the automatic identification of spammers and used the EM algorithm to estimate the error rate.

(3) By considering different test data sets, the proposed model was verified. The simulated results show that the proposed SC model can protect the workers' location privacy on the premise of ensuring the quality of crowdsourcing projects.

Next, we will further study how to apply the model to actual crowdsourcing platform systems, and we will further explore whether the privacy protection and quality control requirements of different types of crowdsourcing tasks have relevant characteristics, and whether we can establish a model to study them. If the said model can be constructed using an adaptive algorithm, perhaps in the case where the k value used for different crowdsourcing tasks no longer has the same fixed value, we may be able to calculate the k value according to the type of task, so as to achieve the best privacy protection and quality control effect.

Data Availability

The data used in this study are owned by a third party.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partly supported by the Zhejiang Natural Science Foundation (LY18G020008 and LQ18F020002), the Zhejiang Soft Science Foundation (2019C35006), the National Natural Science Foundation of China (61202290), and the Huzhou University's Scientific Research Foundation in 2018 (2018XJKJ63).

References

[1] W. Luther, H. Ogata, and J. Pino, "Computer supported collaborative work," *Journal of Universal Computer Science*, vol. 22, no. 10, pp. 1274–1276, 2016.

[2] J. Howe, "The rise of crowdsourcing," *Wired Magazine*, vol. 14, no. 6, pp. 1–4, 2006.

[3] L. Kazemi and S. Cyrus, "GeoCrowd: enabling query answering with spatial crowdsourcing," in *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, pp. 189–198, Redondo Beach, CA, USA, November 2012.

[4] Y. Zhao and Q. Han, "Spatial crowdsourcing: current state and future directions," *IEEE Communications Magazine*, vol. 54, no. 7, pp. 102–107, 2016.

[5] Z. Wu, H. Sun, Z. Guan et al., "Survey on location privacy preservation of continuous spatial queries," *Application Research of Computers*, vol. 32, no. 2, pp. 321–325, 2015.

[6] D. Karger, S. Oh, and D. Shah, "Budget-optimal task allocation for reliable crowdsourcing systems," *Operations Research*, vol. 62, no. 1, pp. 1–24, 2014.

[7] L. Sweeney, "k-anonymity: a model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557–570, 2002.

[8] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "L-diversity: privacy beyond k-anonymity," in *Proceedings of the 22nd IEEE International Conference on Data Engineering*, pp. 24–36, IEEE Press, Atlanta, GA, USA, April 2006.

[9] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: privacy beyond k-anonymity and L-diversity," in *Proceedings of IEEE 23rd International Conference on Data Engineering (ICDE 2007)*, pp. 106–115, Istanbul, Turkey, April 2007.

[10] J. Cao, P. Karras, P. Kalnis, and K.-L. Tan, "SABRE: a sensitive attribute bucketization and R E-distribution framework for t-closeness," *VLDB Journal*, vol. 20, no. 1, pp. 59–81, 2011.

[11] C. Wong, J. Li, A. Fu, and K. Wang, "(α, k) anonymity: an enhanced k-anonymity model for privacy preserving data publishing," in *Proceedings of KDD*, pp. 754–759, Philadelphia, PA, USA, August 2006.

[12] M. Baig, J. Li, J. Liu, and H. Wang, "Cloning for privacy protection in multiple independent data publications," in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management-CIKM'11*, pp. 885–894, Glasgow, UK, October 2011.

[13] J. Hu, L. Huang, L. Li, M. Qi, and W. Yang, "Protecting location privacy in spatial crowdsourcing," in *Web Technologies and Applications*, pp. 113–124, Springer International Publishing, Cham, Switzerland, 2015.

[14] C. Chow, M. Mokbel, and X. Liu, "Spatial cloaking for anonymous location based services in mobile peer to peer environments," *GeoInformatica*, vol. 15, no. 2, pp. 351–380, 2011.

[15] H. To, G. Ghinita, and C. Shahabi, "Framework for protecting worker location privacy in spatial crowdsourcing," *Proceedings of the VLDB Endowment*, vol. 7, no. 10, pp. 919–930, 2014.

[16] K. Vu, R. Zheng, and J. Gao, "Efficient algorithms for k anonymous location privacy in participatory sensing," in *Proceedings of IEEE INFOCOM*, pp. 2399–2407, IEEE Press, Orlando, FL, USA, March 2012.

[17] M. Datar, N. Immorlica, D. Indyk, and V. S. Mirrokni, "Locality sensitive hashing scheme based on p-stable distributions," in *Proceedings of the 20th Annual Symposium on Computational Geometry*, pp. 253–262, ACM Press, Brooklyn, NY, USA, June 2004.

[18] Y. Wang, Z. Cai, Z. Chi, X. Tong, and L. Li, "A differentially k-anonymity-based location privacy-preserving for mobile crowdsourcing systems," *Procedia Computer Science*, vol. 129, pp. 28–34, 2018.

[19] Y. An, K. Qin, and G. Luo, "Survey on location privacy preservation technology in spatial crowdsourcing," *Application Research of Computers*, vol. 35, no. 8, pp. 2241–2244, 2017.

[20] L. Varshney, "Privacy and reliability in crowdsourcing service delivery," in *Proceedings of Service Research and Innovation Institute Global Conference*, pp. 55–60, San Jose, CA, USA, July 2012.

[21] L. Varshney, A. Vempaty, and P. Varshney, "Assuring privacy and reliability in crowdsourcing with coding," in *Proceedings of the 2014 Information Theory and its Applications Workshop (ITA)*, pp. 1–6, San Diego, CA, USA, February 2014.

[22] K. Hiroshi, A. Hiromi, and K. Hisashi, "Preserving worker privacy in crowdsourcing," *Data Mining and Knowledge Discovery*, vol. 28, no. 5-6, pp. 1314–1335, 2014.

[23] G. Huang, Q. Zhu, and C. Siew, "Extreme learning machine: theory and application," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.

[24] G. Huang, S. Song, J. Gupta, and C. Wu, "Semi-supervised and unsupervised extreme learning machines," *IEEE Transactions on Cybernetics*, vol. 44, no. 12, pp. 2405–2417, 2014.

[25] A. M. Dawid and A. P. Skene, "Maximum likelihood estimation of observer error-rates using the EM algorithm," *Applied Statistics*, vol. 28, no. 1, pp. 20–28, 1979.

[26] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[27] J. Feng, G. Li, and J. Feng, "A survey on crowdsourcing," *Chinese Journal of Computers*, vol. 38, no. 9, pp. 1713–1726, 2015.

[28] Y. Wu, Q. Tang, W. Ni, and Z.-H. Sun, "Algorithm for k-anonymity based on rounded partition function," *Journal of Software*, vol. 23, no. 8, pp. 2138–2148, 2012.

[29] Q. He, X. Jin, C. Du, F. Zhuang, and Z. Shi, "Clustering in extreme learning machine feature space," *Neurocomputing*, vol. 128, pp. 88–95, 2014.

[30] G. Huang, "An Insight into extreme learning machines random neurons, random features and kernels," *Cognitive Computation*, vol. 6, no. 3, pp. 376–390, 2014.

[31] D. Yue, G. Yu, D. Shen et al., "Crowdsourcing quality evaluation strategies based on voting consistency," *Journal of Northeastern University (Natural Science)*, vol. 35, no. 8, pp. 1097–1101, 2014.