MLM-WR: A Swarm Intelligence-based Cloud-Edge-Terminal Collaboration Data Collection Scheme in The Era of AIoT

Jiaheng Lu, Zhenzhe Qu*, Anfeng Liu, Shaobo Zhang, Neal N. Xiong*

***Abstract*—Mobile Crowd Sensing (MCS) is a cloud-edge-terminal collaboration model that relies on edge terminal devices, or "workers," to sense data and build applications for cloud-hosted platforms. However, to ensure high-quality application development, recruiting truthful workers in the edge network is crucial. With the emergence of Artificial Intelligence (AI), the Internet of Things (IoT) is entering a new era, known as Artificial Intelligence of Things (AIoT). This paper proposes an AI-enabled MCS system, which includes MLM-WR, a cloud-edge-terminal collaboration data collection scheme for AIoT. MLM-WR leverages swarm intelligence to match truthful workers with sensing tasks, enabling efficient and effective data collection for AIoT applications. The matching theory is applied from two perspectives: truthful workers discovery and sensing difference discovery. To identify truthful workers, we adjust their credibility based on the deviation of their sensing data with Ground Truth Data (GTD) obtained through collaboration with the Unmanned Aerial Vehicle (UAV). In the sensing difference discovery, we obtain workers' sensing attribute reliability by calculating attribute data errors and incorporate absolute and relative sensing location preferences to determine workers' sensing quality at different locations. Additionally, MLM-WR employs the Particle Swarm Optimization (PSO) algorithm to assign workers while considering sensing attribute and location reliability and recruitment cost, thus addressing the tradeoff between recruitment cost and data quality. The effectiveness of our approach is demonstrated through extensive evaluations, where MLM-WR outperformed the state-of-the-art approaches.**

***Index Terms*—Cloud-Edge-Terminal Collaboration, Truthful Workers Discovery, Task Assignment, Artificial Intelligence of Things (AIoT), Sensing Difference discovery, Swarm Intelligence**

I. Introduction

Wireless communication and sensing advancements have driven the rise of IoT [1], a network of devices connected via the Internet. Cloud computing addresses limited local computing capacity by enabling local devices to process and transmit data to the cloud for analysis [2]. Edge computing, a response to constrained cloud bandwidth [3], collaborates with the cloud at the network edge, executing computational tasks to minimize transmission delay and bandwidth usage. MCS exemplifies cloud-edge-terminal collaboration [4], with terminal workers collecting and reporting data. This data undergoes edge-layer processing and filtering before transmission to the cloud, where applications like NoiseTube [5] and Ear-phone [6] emerge. The platform assigns sensing tasks with specific time, location, and data quality requirements [7], [8], with workers participating based on their capabilities. Ensuring high-quality data hinges on recruiting trustworthy workers [9], as malicious submissions for cost reduction and rewards can harm both the platform and users [10].

Truth data discovery involves two main approaches. The first approach employs n workers simultaneously sensing the same object and calculating the mean, median, or weighted average for truth data estimation [11], [12]. However, this method is costly and lacks assessment of individual worker credibility. The second approach evaluates worker trustworthiness before data collection and recruits truthful workers, requiring only one worker per task [13], [14]. Nevertheless, accurately identifying trustworthy workers remains a challenge. Further research is needed on truthful workers and sensing difference discovery.

(1) Truthful workers discovery. Truthful workers discovery is essential for reliable data collection. One method class involves clustering data based on known truthful workers [15], but these methods face limitations due to data's sensitivity to time, location, and individual sensing ability. Relying solely on reported data for worker classification can lead to inaccurate outcomes [16]. Some methods also assume data from a subset of truthful workers as GTD [17] for worker credibility evaluation. However, identifying initial truthful workers and maintaining GTD credibility pose challenges due to potential changes over time. This issue highlights the need for more research in this area.

(2) Sensing difference discovery. Real-world MCS sensing tasks vary spatio-temporally [18], covering diverse attributes such as sound and image [8]. However, workers' differing capabilities to gather data from various locations and attributes result in substantial variability in sensing data quality, influenced by factors like device manufacturers and personal skills [8]. Specialized devices yield accurate data for specific phenomena [19], while generic devices provide average quality. Skill levels also impact data quality, as younger workers tend to capture better images than older ones [20]. Personal preference contributes too, with some excelling in landscapes and others in indoor photography [21]. Consequently, even credible workers yield varying sensing data depending on

- Manuscript received 09 May, 2023; revised 12 August, 2023; accepted 25 August 2023. This work was supported by the National Natural Science Foundation of China 62072475. (*Corresponding author: Zhenzhe Qu, Neal N. Xiong,).
- J. Lu, Z. Qu, and A. Liu are with the School of Computer Science and Engineering, Central South University, Changsha 410083 China (e-mail: jiahenglu@csu.edu.cn, zhenzheQu@csu.edu.cn, afengliu@csu.edu.cn).
- S. Zhang is with the School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan 411201, China (e-mail: shaobozhang@hnust.edu.cn).
- N. N. Xiong is with the Department of Computer Science and Mathematics, Sul Ross State University, Alpine, TX 79830, USA (e-mail: xiongnaixue@gmail.com).

attributes and locations [8]. Improving MCS data quality entails identifying variations in workers' sensing quality across attributes and locations, aligning worker selection with task requirements. However, prior research assumes consistent credibility, disregarding reliability shifts within the same source across attributes or locations.

(3) Task assignment. In MCS, diverse task assignment schemes exist [22], often engaging multiple workers per task with response aggregation using methods like majority voting [12]. However, this proves costly and inefficient, requiring a recruitment-cost versus data-quality equilibrium. Even with known worker credibility, not all workers may be available for tasks [23]. Spatial constraints and worker preferences further narrow task options [24]. Dynamic task assignment is pivotal in addressing these intricacies. The AI evolution has birthed AIoT [25], [26], fostering prospects for intelligent task assignments. Nevertheless, this area remains largely uncharted.

In conclusion, enhancing data quality requires three essential elements: (1) recognizing and picking trustworthy workers, (2) sensing difference discovery across locations and attributes, and (3) matching workers' sensing preferences with corresponding task attributes and locations. This paper introduces MLM-WR, a cloud-edge-terminal collaborative data collection scheme for MCS, addressing these issues through swarm intelligence. The paper presents three main innovations:

(1) Introduce a rapid truthful worker discovery method that assesses workers' credibility through comparison with GTD and Sub-GTD, suitable for real-world applications.

(2) Create a technique for evaluating sensing difference quality, covering workers' performance across attributes and locations. This involves assessing attribute quality through data deviation analysis and integrating an innovative optimization framework with accuracy calculations for diverse locations, thereby establishing workers' credibility in those areas.

(3) Deploy a recruitment scheme that uses swarm intelligence to select truthful workers matching sensing data tasks in terms of attributes and locations, optimizing data quality and cost-effectiveness.

Finally, we conduct extensive evaluations to demonstrate the superiority of our approach over state-of-the-art approaches. Our results show a significant improvement in workers' recognition rate, data quality, and cost-saving, achieving improvements of 36.12%, 27.10%, and 40.19%, respectively.

The remainder of this paper is organized as follows. In Section II, we review related works. Section III presents the system model and problem statement, while Section IV describes the design of our proposed approach. We present a performance analysis of our approach in Section V, and finally, Section VI provides conclusions.

II. Related Work

Truth discovery is crucial for ensuring the quality of applications in MCS and is often studied alongside security and privacy [27]-[29]. This paper focuses on pure truth data discovery and categorizes current research into two types: those that do not require ground truth data and those that do. The former includes Mean, Median, Weighted Average, and Major Voting methods [11], [12], [17], [30]. The Mean method finds the estimated truth data by averaging the data reported by n workers, assuming that most workers are truthful [12]. The Median method takes the median of n data, potentially providing more accurate results [12]. The Weighted Average method combines the Mean and Median methods and assumes a normal distribution, making it less suitable for small n [12]. The Major Voting method includes majority principle and expert evaluation [11], with the former being suitable for discrete values and the latter being expensive and slow.

Ye et al. [15] proposed an iterative truth-finding method called the Mean and Median Check (MMC) method, which combines truth detection and the removal of false data. The method calculates the mean and median of the dataset and removes the data that deviate far from them [15]. This process is repeated until the final data is obtained [15]. However, this method does not check the credibility of workers. To improve the results, the credibility of workers can be taken into account, as workers with higher credibility report more accurate data. The cost and the number of workers required can be reduced by selecting only workers with high credibility.

Methods requiring GTD for truth data discovery are accurate but challenging and expensive to obtain [31], [32]. In contrast, unsupervised methods without GTD [11], [12], [17], [30] incur n times the cost, demanding n data points for the same object. Moreover, certain studies necessitate obtaining GTD. Du et al. classify workers based on their entity recognition accuracy using both tasks with known GTD and tasks without [13]. Tian et al. [8] employ unsupervised computation to gauge workers' relative sensing accuracy across attributes for estimated GTD.

GTD-based truth data discovery ensures accuracy, but obtaining GTD is challenging due to the difficulty and high cost of collecting it [33]. Some studies avoid addressing this crucial issue and assume the existence of GTD or credible workers, but these assumptions are problematic. Credibility is not constant, and obtaining GTD dynamically is necessary for identifying credible workers. To address this issue, we propose an effective method that involves using UAVs to sample data at sampling points [14] and comparing it with workers' data.

III. System Model and Problem Statement

The network model in this paper resembles the cloud-edge-terminal collaborative system used in Refs. [26] and [34], as shown in Fig. 1. The MLM-WR scheme comprises three layers: terminal, edge, and cloud. The terminal layer consists of end users with IoT devices applying for MCS tasks, and UAVs visit specific task locations to sense data as GTD. The edge layer acts as a bridge between terminals and the cloud, involving base stations and edge servers. Data is filtered and processed in the edge layer before transmission to cloud servers. Task assignments or application services are conveyed to end users via the edge layer from the cloud layer. The cloud layer's powerful computing capabilities and ample storage resources enable on-demand services and intelligent applications, utilizing data sensed within the MCS system. This paper's research goal is to improve the data quality at a low cost for the system, including:

(1) Recruit honest workers who can provide high-quality data.

This paper defines $\mathcal{P}_c$ as the proportion of truthful workers among all recruited workers to measure the accuracy of the proposed method. $\mathcal{P}_c$ is calculated by Equation (1).

$$\mathcal{P}_{\mathrm{c}} = \frac{w_{\mathrm{c}}}{w_{total}}. \quad (1)$$

If $\mathcal{P}_{\mathrm{c}}$ is close to 1, it means the proposed method can effectively identify truthful workers. Thus, the objective of the strategy is to maximize $\mathcal{P}_{\mathrm{c}}$. The proposed method's ability to identify truthful workers is evaluated using the algorithm's precision, recall, and F1-score, which are calculated respectively by Equations (2)-(4).

$$Precision = \frac{true\ positives}{true\ positives + false\ positives}. \quad (2)$$

$$Recall = \frac{true\ positives}{true\ positives + false\ negatives}. \quad (3)$$

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall}. \quad (4)$$

(2) Obtain high-quality multidimensional data

This paper discusses multidimensional attribute data, with one-dimensional attribute data being a special case. Although truthful workers will not submit malicious data, data quality can vary greatly based on different attributes and locations.

The i-th dimensional ground truth of the k-th data collection task is denoted as $Q_k^{(i)}$. $Q_{k,j}^{(i)}$ denotes the i-th dimensional data of the k-th task sensed by the j-th worker. Equation (5) represents the difference between the data sensed by worker j and the ground truth of the data in the k-th task's i-th dimension.

$$\mathfrak{I}_{k,j}^{(i)} = \frac{\left|Q_k^{(i)} - Q_{k,j}^{(i)}\right|}{Q_k^{(i)}}. \quad (5)$$

The data quality difference between the data reported by worker j and the ground truth of task k in all dimensions is denoted by Equation (6). Each location has N attributes.

$$\mathfrak{I}_{k,j} = \sum_{i=1}^{N} \mathfrak{I}_{k,j}^{(i)}. \quad (6)$$

For task k, s workers are recruited by the platform to collect data. The average data quality difference of the data collected by the platform in task k is denoted by Equation (7).

$$\widetilde{\mathfrak{I}_k} = \sum_{i=1}^{s} \mathfrak{I}_{k,i}/s. \quad (7)$$

Let there be m tasks in the platform. The total average data quality difference of the platform is represented by Equation (8).

$$\widetilde{\mathfrak{I}} = \sum_{k=1}^{m} \widetilde{\mathfrak{I}_k}/m. \quad (8)$$

The smaller the difference between the data collected by workers and the truthful data, the higher the quality of the data collected by workers recruited by the platform, which is represented by $\min(\widetilde{\mathfrak{I}})$.

(3) Optimize the tradeoff between recruitment cost and data quality in the MCS system.

This involves balancing the cost of obtaining high-quality data and minimizing the cost of recruiting workers. Redundant data may be collected at a cost to make the data close to the truthful data. The total cost of the system includes the cost of the UAV, $\mathbb{C}_{UAV}$, and the cost of recruiting workers, $\mathbb{C}_w$, which is denoted by Equation (9).

$$\mathfrak{C} = \mathbb{C}_{UAV} + \mathbb{C}_w. \quad (9)$$

Therefore, the objective is to minimize costs while identifying truthful workers, ensuring data quality, and mitigating malicious node attacks. The research problem can be expressed as follows: maximize the ratio of truthful workers selected for data collection, minimize the average data quality difference, and total cost.

$$\begin{cases} Max\ (\mathcal{P}_{\mathrm{c}}) = Max\ (w_{\mathrm{c}}/w_{total}); \\ Min(\widetilde{\mathfrak{I}}) = Min\left(\sum_{k=1}^{m} \widetilde{\mathfrak{I}_k}/m\right); \\ Min(\mathfrak{C}) = Min(\mathbb{C}_{UAV} + \mathbb{C}_w). \end{cases} \quad (10)$$

See Table 1 for a list of key parameters used in this paper.

Table 1 Parameter Description

Parameter	Meaning
$\mathcal{D}_{m,n}^{t,k}$	Worker k's observation of the n-th attribute for the location m at time t.
$U_{m,n}^{t}$	The observation of attribute n for location m offered by the UAV at time t
$Z_{k,n}^{t}$	The sensing attribute preference of worker k for the n-th attribute at time t.
$H_{k,m}^{t}$	Worker k's absolute sensing location preference for m-th location at time t.
$C_{i,t}$	The credibility of worker i at time t
E_m	The number of attributes that the platform focuses on at location m
$a_{k,n}^{t}$	Worker k's sensing attribute credibility for attribute n at time t.
$e_{k,m}^{t}$	Worker k's sensing location credibility for location m at time t.

IV. Our Proposed MLM-WR Scheme

The MLM-WR scheme aims to address the issue of unreliable workers submitting malicious data. The scheme has three components: (1) identifying workers' credibility, (2) discovering workers' preferences on sensing data attributes and locations, and (3) making optimal task assignments.

A. Credibility Evaluation Model in MLM-WR

To prevent malicious workers from causing losses to the platform, it is necessary to identify truthful workers in the MCS system. The credibility of workers is evaluated based on the data collected by UAV.

The collected data of UAV is credible and acts as GTD. $\mathcal{S}_{i,j}^{k}$ is the ratio of deviation from GTD to the data collected by worker k for attribute j of location i. The ratio mentioned is computed by utilizing Equation (11), where $U_{i,j}^{t}$ represents j-th attribute of the data gathered by the UAV at location i during time t. On the other hand, $\mathcal{D}_{i,j}^{t,k}$ denotes the value of attribute i of data collected by the worker k on location i at time t.

$$\mathcal{S}_{i,j}^{k} = \frac{\left|\mathcal{D}_{i,j}^{t,k} - U_{i,j}^{t}\right|}{U_{i,j}^{t}}. \quad (11)$$

If $\mathcal{S}_{i,j}^{k} \leq threshold\ \theta$, let $\mathcal{X}_{i,j}^{k} = 1$; else let $\mathcal{X}_{i,j}^{k} = 0$.

$A_{i,k}$, calculated by Equation (12), represents the number of correct entries collected by worker k at location i.

$$A_{i,k} = \sum_{j=1}^{n} \mathcal{X}_{i,j}^{k}. \quad (12)$$

E_i , calculated by Equation (13), is the total number of attributes of data at location i that the platform considers, and σ_r , expresses the effect of the worker-UAV interaction, as computed by Equation (14).

$$E_i = \sum_{j=1}^{n} E_{i,j}. \quad (13)$$

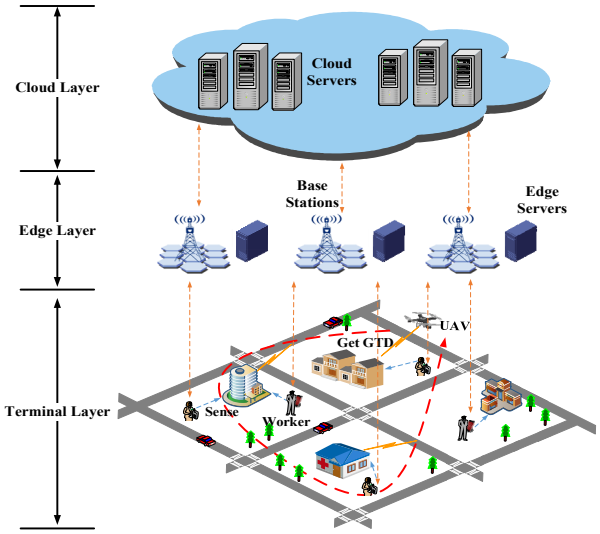


Fig. 1. The illustration of the MLM-WR scheme

$$\sigma_r = \frac{A_{i,k}}{E_i}. \tag{14}$$

(1) *Reward Mechanism*

$C_{k,t}$ is the credibility of worker k at time t, with an initial value of $C_{k,0}$. If $\sigma_r \geq \sigma_s$, worker k is considered truthful, and the Reward Mechanism (RM) is used to increase credibility using Formula (15), where ρ, a variable greater than 1, controls the updating speed of credibility. If a worker is evaluated as truthful repeatedly, the RM is applied repeatedly, leading the worker's credibility to gradually converge to 1 over time.

$$C_{k,t} = C_{k,t-1} + \frac{\sigma_r}{\rho} * \left(1 - C_{k,t-1}\right). \tag{15}$$

(2) *Punishment Mechanism*

If σ_r is less than σ_s, Punishment Mechanism (PM) is used to reduce credibility using Formula (16). If a worker is consistently deemed uncredible, repeated application of the PM reduces their credibility to 0 over time.

$$C_{k,t} = C_{k,t-1} + (\sigma_r - 1) * \left(\frac{C_{k,t-1}}{\rho}\right). \tag{16}$$

(3) *Timeout Mechanism*

To maintain data accuracy, credibility updates for inactive workers are crucial, even if they initially submitted accurate data. The Timeout Mechanism (TM) update, controlled by γ using Formulas (17) and (18), has a smaller impact on credibility compared to PM and RM due to prioritizing data quality over frequency. C_{min} is the minimum credibility (e.g., $C_{min} = 0$).

$$C_{k,t} = Max\left(C_{min}, C_{k,t-1} - \partial\right). \tag{17}$$

$$\partial = \gamma\left(1 - C_{k,t-1}\right). \tag{18}$$

After repeating the process, workers with credibility greater than ν are considered credible. Credible workers, whose data act as Sub-GTD, can update the credibility of other workers who sense data at the same place using either the Vice Reward Mechanism or Vice Punishment Mechanism. Workers not evaluated by Sub-GTD or GTD will undergo TM. $\mathcal{D}_{i,j}^{t,b}$ represents the j-th attribute value of credible worker b at location i and time t. $P_{i,j}^{k}$ represents the deviation ratio between worker k's j-th attribute value at l_i and the Sub-GTD. $A_{i,k}^{c}$ represents the number of incorrect entries worker k collected at l_i during Sub-GTD evaluation. For worker k who senses data at location i, and where the platform is concerned with l_i's attribute j, $P_{i,j}^{k}$ is calculated using Formula (19).

$$P_{i,j}^{k} = \frac{\left|\mathcal{D}_{i,j}^{t,k} - \mathcal{D}_{i,j}^{t,b}\right|}{\mathcal{D}_{i,j}^{t,b}}. \tag{19}$$

If $P_{i,j}^{k} \leq threshold\ \theta^c$, let $y_{i,j}^{k} = 1$; else let $y_{i,j}^{k} = 0$.

$A_{i,k}^{c}$ is the number of correct entries collected by worker k at location i under Sub-GTD evaluation, which is calculated by Formula (20). The interaction term σ_r^c represents the impact of the interaction between the worker and Sub-GTD, and it is computed as the ratio of $A_{i,k}^{c}$ and E_i.

$$A_{i,k}^{c} = \sum_{j=1}^{n} y_{i,j}^{k}. \tag{20}$$

$$\sigma_r^c = \frac{A_{i,k}^{c}}{E_i}. \tag{21}$$

(4) *Vice Reward Mechanism*

If $\sigma_r^c \geq threshold\ \sigma_s^c$, worker k is considered credible by Sub-GTD, and the Vice Reward Mechanism is used to update their credibility using Formula (22), where ρ^c, a variable greater than 1, controls the update speed. Updates are larger for more credible worker b, making the mechanism similar to GTD when worker b is highly credible.

$$C_{k,t} = C_{k,t-1} + \frac{\sigma_r^c}{\rho^c} * \left(1 - C_{k,t-1}\right) * C_{b,t-1}. \tag{22}$$

(5) *Vice Punishment Mechanism*

If $\sigma_r^c < \sigma_s^c$, Sub-GTD considers worker k untrustworthy and reduces worker k's credibility using Formula (23)

$$C_{k,t} = C_{k,t-1} + (\sigma_r^c - 1) * \left(\frac{C_{k,t-1}}{\rho^c}\right) * C_{b,t-1}. \tag{23}$$

B. Sensing Attribute Preference Evaluation Model in MLM-WR

In prior sections, we tackled worker credibility on a macro level. However, given the multidimensional data variations among workers, this section introduces the Sensing Attribute Preference Evaluation Model (SAPE). This model evaluates workers' Sensing Attribute Preferences (SAPs) to elevate data accuracy. $\mathcal{D}_{m,n}^{t,k}$ is the n-th attribute of data collected by worker k at time t, at location m. $U_{m,n}^{t}$ is the value of the n-th attribute of location m collected by UAV at time t, also known as Attribute-GTD (A-GTD). A-GTD helps to determine the SAP of workers. Let K be the number of workers in the network, and $Z_{k,n}^{t}$ represent worker k's sensing preference for attribute n at time t. The evaluation of SAP is described below. If attribute n at location m is relevant to the task, use Formula (24) to calculate $Y_{m,n}^{k}$. $Z_n^t = \left\{Z_{1,n}^{t}, Z_{2,n}^{t}, \ldots, Z_{K,n}^{t}\right\}$ denotes the set of sensing preferences of workers for attribute n at time t.

$$Y_{m,n}^{k} = \frac{\left|\mathcal{D}_{m,n}^{t,k} - U_{m,n}^{t}\right|}{U_{m,n}^{t}}. \tag{24}$$

If $Y_{m,n}^{k} < 1$, let $P_n = 1 - Y_{m,n}^{k}$; else, let $P_n = 0$.

(1) *Attribute Reward Mechanism*

$Z_{k,n}^{t}$ denotes worker k's sensing preference for attribute n at time t, with $Z_{k,n}^{0}$ as the initial value. When $Y_{m,n}^{k} \leq threshold\ \Omega$ ($0 < \Omega < 1$), A-GTD considers worker k credible for attribute n, and increases the preference for attribute n using Formula (25). The update rate is governed by δ ($\delta > 1$).

$$Z_{k,n}^{t} = Z_{k,n}^{t-1} + \frac{P_n}{\delta}\left(1 - Z_{k,n}^{t-1}\right). \quad (25)$$

(2) *Attribute Punishment Mechanism*

Formula (26) decreases worker k's sensing preference for attribute n if $Y_{m,n}^{k} > \Omega$, indicating not credible for attribute n.

$$Z_{k,n}^{t} = Z_{k,n}^{t-1} + (P_n - 1) * \left(\frac{Z_{k,n}^{t-1}}{\delta}\right). \quad (26)$$

(3) *Attribute Timeout Mechanism*

The Attribute Timeout Mechanism lowers inactive workers' SAP over time using Formulas (27) and (28). $Z_{k,n}^{t}$ represents worker k's SAP for attribute n at time t, Z_{min} is the minimum SAP (e.g., $Z_{min} = 0$), and φ determines the rate of decrease.

$$Z_{k,n}^{t} = Max\left(Z_{min}, Z_{k,n}^{t-1} - \varphi\right). \quad (27)$$

$$\varphi = \mu\left(1 - Z_{k,n}^{t-1}\right). \quad (28)$$

If $Z_{b,n}^{t}$ exceeds threshold §, worker b has good sensing quality for attribute n, whose data can be used as Sub-A-GTD. If multiple workers at location m (not visited by UAV at time t) have good sensing quality for n, the data from the worker with the highest preference is chosen as Sub-A-GTD. If worker b's data is selected as Sub-A-GTD at location m, worker b evaluates attribute n of other workers there and updates their sensing preference using Vice Attribute Reward or Vice Punishment Mechanism. If attribute n of worker k hasn't been evaluated by Sub-A-GTD or A-GTD, the Attribute Timeout Mechanism applies. When location m's attribute n is of interest, worker k's SAP for attribute n at location m is updated using Formula (29) to calculate $q_{m,n}^{k}$. If $q_{m,n}^{k} < 1$, let $P_n^c = 1 - q_{m,n}^{k}$; else let $P_n^c = 0$.

$$q_{m,n}^{k} = \frac{\left|\mathcal{D}_{m,n}^{t,k} - \mathcal{D}_{m,n}^{b,t}\right|}{\mathcal{D}_{m,n}^{b,t}}. \quad (29)$$

(4) *Vice Attribute Reward Mechanism*

To determine the Vice Attribute Reward or Punishment Mechanism, a threshold Ω^c $(0 < \Omega^c < 1)$ is set. If $q_{m,n}^{k} \le \Omega^c$, the Vice Attribute Reward Mechanism updates worker k's sensing attribute preference for attribute n using Formula (30), where δ^c controls the update speed.

$$Z_{k,n}^{t} = Z_{k,n}^{t-1} + \frac{P_n^c}{\delta^c} * \left(1 - Z_{k,n}^{t-1}\right) * Z_{b,n}^{t}. \quad (30)$$

(5) *Vice Attribute Punishment Mechanism*

If $q_{m,n}^{k} > \Omega^c$, worker k is considered unreliable on attribute n by Sub-A-GTD. To ensure $Z_{k,n}^{t}$ converges to 0, the minimum value of P_n^c is set to 0. The sensing preference for attribute n decreases using Formula (31).

$$Z_{k,n}^{t} = Z_{k,n}^{t-1} + (P_n^c - 1) * \left(\frac{Z_{k,n}^{t-1}}{\delta^c}\right) * Z_{b,n}^{t}. \quad (31)$$

C. Sensing Location Preference Evaluation Model in MLM-WR

The previous section assessed workers' credibility and sensing attribute preferences but didn't consider data accuracy variability based on location. The paper introduces sensing location preference and proposes the Sensing Location Preference Evaluation Model (SLPE) to evaluate workers based on their location. Absolute and relative sensing location preferences are proposed based on UAV and workers' own supervision to make the assessment more comprehensive.

(1) *Calculation of absolute sensing location preference*

For K workers in the network, Location-GTD (L-GTD), data sensed by the UAV, is compared to data sensed by workers at the same location. Absolute sensing location preference (ASLP) for worker k at location m, denoted $H_{k,m}^{t}$, is updated based on this comparison. $H_{k,m}^{0}$ is the initial value. If worker k and UAV collect data at location m during time t, and attribute n of location m is relevant, $O_{m,n}^{k}$ is calculated using Formula (32).

$$O_{m,n}^{k} = \frac{\left|\mathcal{D}_{m,n}^{t,k} - U_{m,n}^{t}\right|}{U_{m,n}^{t}}. \quad (32)$$

If $O_{m,n}^{k} \le threshold$ ℵ, let $O_{m,n}^{k} = 1$; else $O_{m,n}^{k} = 0$.

If the task at location m does not involve attribute n, $O_{m,n}^{k} = 0$. $F_{m,k}$ is the number of correct data entries collected by worker k at location m, evaluated by L-GTD using Formula (33). The interaction right η_m indicates the effect of worker-UAV interaction at location m.

$$F_{m,k} = \sum_{n=1}^{N} O_{m,n}^{k}. \quad (33)$$

$$\eta_m = \frac{F_{m,k}}{E_m}. \quad (34)$$

1) *Location Reward Mechanism*

A threshold η_s is used to decide whether to apply the Location Reward Mechanism or the Location Punishment Mechanism. If $\eta_m \ge \eta_s$, worker k is considered credible for location m. Formula (35) updates worker k's ASLP for location m, with τ, a variable greater than 1, controlling the update speed.

$$H_{k,m}^{t} = H_{k,m}^{t-1} + \frac{\eta_m}{\tau} * \left(1 - H_{k,m}^{t-1}\right). \quad (35)$$

2) *Location Punishment Mechanism*

Formula (36) decreases the ASLP of worker k for location m if $\eta_m < \eta_s$, indicating a lack of credibility for that location.

$$H_{k,m}^{t} = H_{k,m}^{t-1} + (\eta_m - 1) * \left(\frac{H_{k,m}^{t-1}}{\tau}\right). \quad (36)$$

3) *Location Timeout Mechanism*

If worker k's data for location m is not evaluable at time t, the ASLP for location m will gradually decline by Formulas (37) and (38), with the parameter φ controlling the decline rate. H_{min} is the minimum ASLP (e.g. $H_{min} = 0$).

$$H_{k,m}^{t} = Max\left(H_{min}, H_{k,m}^{t-1} - \Psi\right). \quad (37)$$

$$\Psi = \varphi\left(1 - H_{k,m}^{t-1}\right). \quad (38)$$

If a worker's ASLP for location m exceeds threshold Γ, the worker is deemed to have a high ASLP for that location. If multiple workers have high ASLP for location m, not sensed by UAV at time t, the data of the worker with the highest ASLP, denoted as worker b, acts as Sub-L-GTD to assess other workers at location m using the Vice Location Reward or Punishment Mechanism. The Location Timeout Mechanism is applied to worker k's ASLP for location m if not evaluated by either Sub-L-GTD or L-GTD. The ratio between ordinary worker k's entry and that of Sub-L-GTD is denoted as $R_{m,n}^{k}$.

$$R_{m,n}^{k} = \frac{\left|\mathcal{D}_{m,n}^{t,k} - \mathcal{D}_{m,n}^{b,t}\right|}{\mathcal{D}_{m,n}^{b,t}}. \quad (39)$$

If $R_{m,n}^{k} \le threshold\ \aleph^c$, let $R_{m,n}^{k} = 1$; else $R_{m,n}^{k} = 0$; if attribute n is not relevant to the task at location m, then $R_{m,n}^{k} = 0$. $F_{m,k}^{c}$ counts the number of correct entries collected by worker k in location m, as evaluated by Sub-L-GTD.

$$F_{m,k}^{c} = \sum_{n=1}^{N} R_{m,n}^{k}. \tag{40}$$

$$\eta_m^c = \frac{F_{m,k}^{c}}{E_m}. \tag{41}$$

4) *Vice Location Reward Mechanism*

If $\eta_m^c \geq threshold\ \eta_s^c$, Formula (42) is used to increase worker k's ASLP for location m, and τ^c, a variable greater than 1, controls the update speed.

$$H_{k,m}^{t} = H_{k,m}^{t-1} + \frac{\eta_m^c}{\tau^c} * \left(1 - H_{k,m}^{t-1}\right) * H_{b,m}^{t}. \tag{42}$$

5) *Vice Location Punishment Mechanism*

Formula (43) is used to decrease worker k's ASLP for location m if $\eta_m^c < \eta_s^c$, indicating not credible for location m.

$$H_{k,m}^{t} = H_{k,m}^{t-1} + (\eta_m^c - 1) * \left(\frac{H_{k,m}^{t-1}}{\tau^c}\right) * H_{b,m}^{t}. \tag{43}$$

(2) *Calculation of relative sensing location preference*

Some workers cannot be evaluated by L-GTD or Sub-L-GTD. As a solution, we propose the relative sensing location preference (RSLP), which includes the Absolute Sensing Location Preference Sequence (ASLPS) and the Autonomy Supervised Sensing Location Preference Sequence (ASSLPS).

1) *The setup of ASLPS*

ASLPS is a sequence derived from ASLP. At round t and location m, H_m^t represents the set of ASLP of all K workers. H_m' is a copy of H_m^t with workers whose sensing quality for location m has not been evaluated by L-GTD or Sub-L-GTD removed. The remaining k_t elements in H_m' are sorted in descending order to obtain the ASLPS for location m, denoted by $L_m^t = \{L_{m,1}^t, \dots, L_{m,k_t}^t\}$. If worker k appears in the j-th place in L_m^t, then $L_{m,j}^t = k$, indicating that worker k has the j-th largest ASLP for location m among all workers.

2) *Calculation of RSLQW*

Relative Sensing Location Quality Weight (RSLQW) indicates the relative quality of data sensed by workers at the same location. ASSLPS represents the partial order relationship of sensing quality among workers at location m based on their RSLQW. To obtain ASSLPS, it is necessary to establish RSLQW first using an optimization framework. $\mathcal{T}_{m,n}^{(*)}$ represents the relative truthful value of location m's attribute n, established during the RSLQW calculation process, using Formula (44) to determine initial entries of relative truthful data.

$$\mathcal{T}_{m,n}^{(*)} = \frac{\sum_{i=1}^{K_m} \mathcal{D}_{m,n}^{t,k}}{K_m}. \tag{44}$$

The higher the RSLQW, whose sensing data should be to the truthful data. The objective function (45) aims to minimize the weighted sum of deviations between the data sensed by workers and relative truthful data. The RSLQW of worker k for location m at time t is represented by $b_{k,m}^t$, and the set of RSLQW of location m is $B_m^t = \{b_{1,m}^t, b_{2,m}^t \dots, b_{K_m,m}^t\}$. It is assumed that K_m workers collect data at location m at time t, and each location has N attributes.

$$\min \sum_{k=1}^{K_m} b_{k,m}^t * \sum_{n=1}^{N} \left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|, \quad S.T. f_c\left(b_{k,m}^t\right) = 1. \tag{45}$$

The RSLQW calculation assumes an exponential distribution, and Equation (46) is obtained.

$$f_c\left(b_{k,m}^t\right) = \sum_{k=1}^{K_m} e^{\left(-b_{k,m}^t\right)} = 1. \tag{46}$$

The RSLQW is calculated iteratively using two steps:

Step 1: Update RSLQW of location m.

Theorem 3. Assuming that the true value is static, the optimal solution for RSLQW of workers is given by:

$$b_{k,m}^t = \log\left(\frac{\sum_{k=1}^{K_m}\sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|}{\sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|}\right).$$

Proof. Supposing that $v_{k,m}^t = e^{\left(-b_{k,m}^t\right)}$, The objective function (45) can be translated into the objective function (47):

$$\min \sum_{k=1}^{K_m} -\log\left(v_{k,m}^t\right) \sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|, \quad S.T. \sum_{k=1}^{K_m} v_{k,m}^t = 1. \tag{47}$$

The objective function (47) is convex and can be solved using the Lagrange multiplier method since it is a linear combination of negative logarithm functions, and the constraint is linear to $v_{k,m}^t$. Then we get equation (48).

$$L\left(\left\{v_{k.m}{}_{k=1,m=1}^{t\ K_m,M}, \lambda\right\}\right) = \sum_{k=1}^{K} -\log\left(v_{k,m}^t\right) * \sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right| + \lambda\left(\sum_{k=1}^{K_m} v_{k,m}^t - 1\right). \tag{48}$$

Set the partial derivative of the Lagrangian with respect to all $v_{k,m}^t$ to 0, where λ is a Lagrange multiplier.

$$\begin{cases} -\dfrac{1}{v_{1,m}^t}\displaystyle\sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,1}\right)\right| + \lambda = 0, \\ \dots \\ -\dfrac{1}{v_{K_m,m}^t}\displaystyle\sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,K_m}\right)\right| + \lambda = 0. \end{cases}$$

After summing the equations above, we can have the Equation (49):

$$\lambda \sum_{k=1}^{K_m} v_{k,m}^t = \sum_{k=1}^{K_m}\sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|. \tag{49}$$

Because of $\sum_{k=1}^{K_m} v_{k,m}^t = 1$, Equation (49) becomes:

$$\lambda = \sum_{k=1}^{K_m}\sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|. \tag{50}$$

Supposing k is static, Equation (49) becomes:

$$\lambda v_{k,m}^t = \sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|. \tag{51}$$

Combine Equations (50) and (51). We can get the following:

$$v_{k,m}^t = \frac{\sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|}{\sum_{k=1}^{K_m}\sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|}.$$

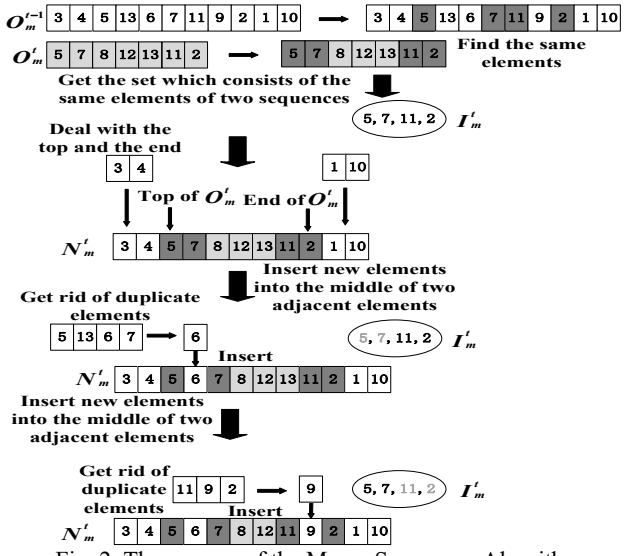


Fig. 2. The process of the Merge Sequences Algorithm

Because $v_{k,m}^t = e^{(-b_{k,m})}$, the optimal solution is:

$$b_{k,m}^t = \log\left(\frac{\sum_{k=1}^{K_m}\sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|}{\sum_{n=1}^{N}\left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|}\right). \quad (52)$$

Step 2: Update relative truthful data for location m based on the newly calculated RSLQW set B_m^t using the objective function (53), where n and m are constants.

$$\mathcal{T}_{m,n}^{(*)} \leftarrow \arg\min \sum_{k=1}^{K_m} b_{k,m}^t * \left|\left(\mathcal{T}_{m,n}^{(*)} - \mathcal{D}_{m,n}^{t,k}\right)\right|. \quad (53)$$

Using the weighted average method (Equation (54)) can minimize objective function (53), producing updated relative truthful data for location m denoted as $\mathcal{T}_{m,n}^{(*),x}$ at the x-th iteration. Formula (55) calculates the difference in relative truthful data between consecutive iterations, stopping when the difference is small ($d_m > threshold$ Д), or the number of iterations exceeds threshold ψ.

$$\mathcal{T}_{m,n}^{(*)} = \frac{\sum_{k=1}^{K_m} b_{k,m}^t * \mathcal{D}_{m,n}^{t,k}}{\sum_{k=1}^{K_m} b_{k,m}^t}. \quad (54)$$

If $\frac{\left|\mathcal{T}_{m,n}^{(*),x} - \mathcal{T}_{m,n}^{(*),x-1}\right|}{\mathcal{T}_{m,n}^{(*),x-1}} < threshold$ з, $\hbar_{m,n} = 1$; else $\hbar_{m,n} = 0$.

$$d_m = \frac{\sum_{n=1}^{N} \hbar_{m,n} * E_{m,n}}{E_m}. \quad (55)$$

3) *Set up of ASSLPS*

Assuming there are K_m^t workers at location m at time t, with known RSLQW B_m^t and set of worker numbers $X^t = \left\{x_1^t, x_2^t, \ldots, x_{K_m^t}^t\right\}$, ASSLPS $R_m^t = \left\{r_{m,1}^t, \ldots, r_{m,K_m^t}^t\right\}$ is obtained by sorting B_m^t in descending order and getting the corresponding worker numbers. If worker k has the second-largest weight in B_m^t, then $r_{m,2}^t = k$. The Merge Sequences Algorithm is used to combine ASSLPS from the previous round with the current ASSLPS to get N_m^t. The algorithm retains the elements in $O_m^t = \left\{y_{m,1}^t, \ldots, y_{m,x_t}^t\right\}$ and adds as many elements from $O_m^{t-1} = \left\{y_{m,1}^{t-1}, \ldots, y_{m,x_{t-1}}^{t-1}\right\}$ as possible, with O_m^t having a higher priority. The algorithm is detailed in Algorithm 1.

Operation of The Special Condition:

If adjacent elements $y_{m,k}^t$ and $y_{m,k+1}^t$ in O_m^t also exist in O_m^{t-1}, and segment $\left\{y_{m,k}^t, y_{m,j}^{t-1}, \ldots, y_{m,j+u}^{t-1}, y_{m,k+1}^t\right\}$ is in O_m^{t-1},

Algorithm 1 Merge Sequences Algorithm

Input: *Sequence* O_m^t *and* O_m^{t-1}, O_m^t *has a higher priority*
Output: *The result* N_m^t

If $O_m^t == NULL$ **and** $O_m^{t-1} != NULL$
  $N_m^t = O_m^{t-1}$
Else if $O_m^t != NULL$ **and** $O_m^{t-1} == NULL$
  $N_m^t = O_m^t$
Else if $O_m^t != NULL$ **and** $O_m^{t-1} != NULL$
  $I_m^t = O_m^t \cap O_m^{t-1}$
  If $I_m^t == \left\{y_{m,1}^t\right\}$ //Add elements before $y_{m,1}^t$ in O_m^{t-1}
    $N_m^t = \left\{y_{m,1}^{t-1}, \ldots, y_{m,1}^t, y_{m,2}^t, \ldots, y_{m,x_t}^t\right\}$
  Else if $I_m^t == \left\{y_{m,x_t}^t\right\}$ // Add items after y_{m,x_t}^t in O_m^{t-1}
    $N_m^t = \left\{y_{m,1}^t, \ldots, y_{m,x_t}^t, \ldots, y_{m,x_{t-1}}^{t-1}\right\}$
  Else if $len(I_m^t) > 1$
    If $y_{m,1}^t$ *in* I_m^t **and** y_{m,x_t}^t *in* I_m^t
      $N_m^t = \left\{y_{m,1}^{t-1}, \ldots, y_{m,1}^t, \ldots, y_{m,x_t}^t, \ldots, y_{m,x_{t-1}}^{t-1}\right\}$
    Else if $y_{m,1}^t$ *not in* I_m^t **and** y_{m,x_t}^t *in* I_m^t
      $N_m^t = \left\{y_{m,1}^t, \ldots, y_{m,x_t}^t, \ldots, y_{m,x_{t-1}}^{t-1}\right\}$
    Else if $y_{m,1}^t$ *in* I_m^t **and** y_{m,x_t}^t *not in* I_m^t
      $N_m^t = \left\{y_{m,1}^{t-1}, \ldots, y_{m,1}^t, \ldots, y_{m,x_t}^t\right\}$
    End If
    For $i \leftarrow 1$ *to* $len(O_m^t) - 1$ **do**
      If $O_m^t(i)$ **and** $O_m^t(i+1)$ **in** I_m^t
        do **Operation of The Special Condition**
      End If
    End For
End If
Return N_m^t

then the segment is inserted into N_m^t between $y_{m,k}^t$ and $y_{m,k+1}^t$ after removing elements present in both N_m^t and the segment. Algorithm 1 updates the ASSLPS R_m^t using inputs R_m^t and R_m^{t-1}, with priority given to R_m^t. Fig. 2 shows the process.

Algorithm 2 Integrated Merge Algorithm

Input: *ASSLPS sets* R^t, R^{t-1}, *ASLPS set* L^t. L_m^t *has a higher priority than* R_m^t, *set of all workers' number* W.
Output: *The set of SLPPOR* $G^t = \{G_1^t, G_2^t, \ldots, G_M^t\}$

For $m \leftarrow 1$ *to* M **do**
  If $t > 1$
    Apply Merge Sequences Algorithm to merge R_m^t *and* R_m^{t-1}, *with higher priority given to* R_m^t, *then obtain* N_m^t *and assign it to* R_m^t
  If $L_m^t != NULL$ **and** $R_m^t == NULL$
    $G_m^t = L_m^t$
  Else if $L_m^t == NULL$ **and** $R_m^t != NULL$
    $G_m^t = R_m^t$
  Else if $L_m^t == NULL$ **and** $R_m^t == NULL$
    $G_m^t == NULL$
  Else if $L_m^t != NULL$ **and** $R_m^t != NULL$
    Apply Merge Sequences Algorithm to merge L_m^t *and* R_m^t, *with higher priority given to* L_m^t, *then get* N_m^t *and assign it to* G_m^t
  End If
  Get the set W'
  Shuffle W' *to get* w_m^t, *then link* G_m^t *and* w_m^t
End For
Return G^t

4) *The integration of ASSLPS and ASLPS*

After updating R_m^t, merge it with ASLPS L_m^t using the Merge Sequences Algorithm to obtain G_m^t, which represents the Sensing Location Preference Partial Order Relationship (SLPPOR) of all workers at location m and time t. The shuffled set of workers' numbers whose sensing ability at location m has

not been evaluated, W', is added to the end of G_m^t, and R^t, L^t, and G^t represent the sets of ASSLPS, ASLPS, and SLPPOR, for M locations respectively. Algorithm 2 outlines the process.

D. The Task Assignment Model in MLM-WR

In preceding sections, we obtain sets G^t (SLPPOR), $Z = \{Z_1^t, \dots, Z_N^t\}$ (SAP), and $C^t = \{C_{1,t}, \dots, C_{K,t}\}$ (worker credibility) for the network. Given limited resources, we need to select the most efficient recruitment and assignment of sensing tasks from K workers. The task sequence is $p = \{p_1, p_2 \dots, p_M\}$, where the set of the M locations is represented by the set ℓ. Workers select multiple locations, each with a cost c, but can only work at one location at a time, with each location having N attributes.

(1) *Filter malicious workers*

Exclude workers with credibility below threshold Θ, indicating malicious workers. Let K'' be the number of remaining workers after this filtering process.

(2) *Arrange the locations where the UAV senses data*

Assuming Y workers are identified as credible among K'' workers and the others are ordinary workers. Let U be the number of locations to be sensed by the UAV, and ℓ' be the set of locations where Y workers are willing to go. The set of locations where no truthful workers are willing to go is denoted by ℓ''. The set of locations where the UAV senses data is ℓ^U. The set of locations where truthful workers are arranged is ℓ^C. If ℓ'' has more than U elements, randomly select U elements from ℓ'' as UAV sensing locations; otherwise, all ℓ'' locations are assigned to the UAV and $U - len(\ell'')$ locations are randomly selected from ℓ' to form ℓ^U. The set of locations where truthful workers are arranged is $\ell^C = \ell' - \ell^U$.

(3) *Arrange truthful workers*

To maximize efficiency, the Y truthful workers need to be allocated properly due to limited resources. First, the set of sensing attribute preferences $Z_n^t = \{Z_{1,n}^t, \dots, Z_{K,n}^t\}$ is sorted in descending order to obtain the sequence Z_n'. The sequence consists of workers' numbers, with earlier numbers indicating higher sensing attribute preferences for attribute n. The position of worker k's number in Z_n' is denoted by $s_{k,n}^t$, with smaller $s_{k,n}^t$ value indicating better sensing attribute quality for attribute n. The value range of $s_{k,n}^t$ is $[1, K]$, and $s_n^t = \{s_{1,n}^t, \dots, s_{K,n}^t\}$. The sensing attribute credibility of worker k for attribute n, denoted as $a_{k,n}^t$, is established using Formula (56). Once the set ℓ^C, consisting of J elements, has been identified, the task sequence for truthful workers is determined as $p_c = \{p_{\mathcal{L}_1}, p_{\mathcal{L}_2} \dots, p_{\mathcal{L}_J}\}$. This sequence specifies which tasks truthful workers need to perform. $\mathcal{L}_i$ refers to the location corresponding to the i-th task in the sequence, where $\mathcal{L}_i = m$ indicates that task $p_{\mathcal{L}_i}$ requires workers to go to location m to execute the task p_m.

$$a_{k,n}^t = K - S_{k,n}^t + 1. \tag{56}$$

The rank of worker k in the set G_m^t is represented by $g_{k,m}^t$, and the smaller the rank, the higher the sensing location credibility of worker k for location m. The sensing location credibility of worker k for location m at moment t is $e_{k,m}^t$, which can be calculated using Formula (57). The credibility sets for sensing attributes and sensing locations are denoted as $a_n^t = \{a_{1,n}^t, \dots, a_{K,n}^t\}$ and $e_m^t = \{e_{1,m}^t, \dots, e_{K,m}^t\}$, respectively. The set of credibility for N attributes is $a^t = \{a_1^t, \dots, a_N^t\}$, and the set of credibility for M locations is $e^t = \{e_1^t, \dots, e_M^t\}$.

$$e_{k,m}^t = K - g_{k,m}^t + 1. \tag{57}$$

T_m^t represents the task execution degree of the task p_m at time t, which is calculated using Formula (58). This formula takes into account the worker's sensing location credibility for location m and the sensing attribute credibility for the attributes that task p_m focuses on and reflects that the execution degree of the task p_m is higher if both of these are high.

$$T_m^t = \sum_{n=1}^{N} e_{k,m}^t * a_{k,n}^t * E_{mn}. \tag{58}$$

The algorithm for obtaining the optimal solution uses PSO, which simulates a group of animals moving together and sharing information. PSO uses particles as solution candidates, and their positions represent their current solutions [35]. The PSO has a population size of P, where each particle's position is represented by $PSO[i]$, a J-dimensional vector. The fitness function (Equation (59)) evaluates particle quality based on task sensing. The objective is to maximize the total task execution (T_{total}), with particle i's fitness ($f[i]$) directly tied to T_{total}. Historical optimal solutions for each particle are denoted as $P_b[i]$, with corresponding fitness as $P_{b\_f}[i]$; the overall optimal solution is represented by P_g. The index of the current global optimal particle is G_b, and the highest historical optimal fitness among particles is f_g.

$$T_{total} = \sum_{j=1}^{J} T_{\mathcal{L}_j}^t. \tag{59}$$

$PSO[i][j]$ has a unique numbering system for each task, such that the same number in different $PSO[i][j]$ corresponds to a different worker. $PSO[i][j]$ encodes the c workers assigned to the task $p_{\mathcal{L}_j}$, and ranges from $\{1,2,3 \dots \mathbb{x}_j\}$. For example, $PSO[i][j] = 3$ means the 3rd worker is assigned to the task $p_{\mathcal{L}_j}$ according to $p_{\mathcal{L}_j}$'s numbering method. $PSO[i]$ initializes with the first task in the sequence. A worker who is willing to perform $p_{\mathcal{L}_j}$ and has not been assigned is randomly selected and assigned to perform the task. If no worker is available for the task $p_{\mathcal{L}_j}$ among those who are willing and unassigned, then $PSO[i][j] = -1$, indicating no worker will be assigned. We can obtain the following optimization equation for the i-th particle.

$$\begin{aligned} V[i] = \omega * V[i] + \varrho_1 * r_1(P_b[i] - PSO[i]) \\ + \varrho_2 * r_2\big(P_g - PSO[i]\big). \end{aligned} \tag{60}$$

$$PSO[i] = PSO[i] + V[i]. \tag{61}$$

Velocity vector elements are floored after updating with Formulas (60) and (61) to eliminate decimal fractions. If a number is less than -1, it is converted to -1. If $PSO[i][j]$ is greater than $\mathbb{x}_j$, it will be converted to $\mathbb{x}_j$. If a worker is assigned to multiple locations, the task with the highest execution degree is kept for that worker, and other locations will not be assigned workers. The number of scheduled workers of particle i is denoted by $h[i]$, while $h_b[i]$ represents the largest number of historically arranged workers of particle i, and h_g represents the largest number of historically arranged workers of all particles. The best particle is determined by comparing the number of scheduled workers first, then by the fitness function. In other words, if $h[i] > h[j]$, particle i is considered better. If $h[i] = h[j], f[i] > f[j]$, then particle i is considered better.

Algorithm 3 Task Assignment of Truthful Workers

Input: $Sensing\ attribute\ /\ location\ credibility\ sets\ a^t, e^t$
Output: $Historical\ global\ optimum\ solution\ P_g$

For $i \leftarrow 1\ to\ P$ **do** //Initialization
  $Initialize\ PSO[i]$
End for
For $each\ particle\ i\ in\ the\ population$ **do**
  $Initialize\ V[i]\ randomly$
  $Evaluate\ f[i]\ and\ h[i]$
  $Set\ P_{b\_f}[i] = f[i], h_b[i] = h[i], P_b[i] = PSO[i]$
End for
  $Set\ G_b\ to\ the\ current\ global\ optimum's\ index$
  $Set\ P_g = PSO[G_b], f_g = f[G_b], h_g = h[G_b]$
Repeat //Update
  For $each\ particle\ i\ in\ the\ population$ **do**
    $Update\ PSO[i]\ and\ evaluates\ its\ fitness\ f[i]$
  End For
  $Set\ G_b\ to\ the\ current\ global\ optimum's\ index.$
  $Update\ P_g, h_g$
  End If
  For $each\ particle\ i\ in\ the\ population$ **do**
    $Update\ P_b[i], P_{b\_f}[i], h_b[i]$
  End for
Until $a\ specified\ number\ of\ generations;$
Return P_g

The process is shown in Algorithm 3.

(4) *Arrangement of ordinary workers.*

Ordinary workers are workers who are not classified as either truthful or malicious. At a given time t, the set of available ordinary workers who are willing to perform the task p_m is denoted as $W_{p_m}^t$.

1) When no truthful workers apply:

If no truthful worker or UAV is available to perform the task p_m, a threshold α is set. If the number of workers in $W_{p_m}^t$ is greater than α, then α workers are randomly selected from the set to perform p_m. If the number of workers in $W_{p_m}^t$ is less than or equal to α, then all workers in the set are dispatched to perform the task.

2) When truthful workers apply:

If a truthful worker or UAV is available to perform the task p_m, a threshold β is set to determine the number of ordinary workers to be dispatched. If the number of available workers is less than β, then all workers in $W_{p_m}^t$ are dispatched. If the number of available workers is greater than β, then β workers with the highest credibility are selected from $W_{p_m}^t$ to perform p_m. There are M locations in the network. The value of β is adjusted based on σ, the number of workers that can be dispatched. If $\sigma < \theta_1 * \mathrm{M}$, let $\beta = \beta_1$; else if $\sigma < \theta_2 * M$, let $\beta = \beta_2$; else if $\sigma < \theta_3 * M$, let $\beta = \beta_3$; else if $\sigma < \theta_4 * M$, let $\beta = \beta_4$; otherwise, β is set to 0. $\theta_1, \theta_2, \theta_3$ and θ_4 are greater than 1.

V. Experimental Results

A. Experiment Setup

The MLM-WR scheme uses Python to implement the simulation experiments with The Diabetes Dataset [35], which contains 768 objects and 6912 observations, each with nine attributes, making them suitable for simulating data from different locations.

Four experimental scenarios are set up with varying numbers of workers, selected objects, and UAV visitation patterns. In scenario 1, there were 100 workers and 10 locations, with 2 locations visited by the UAV each time. Scenario 2 had 150 workers and 15 locations, with 3 locations visited by the UAV each time. Scenario 3 had 200 workers and 20 locations, with 2 locations visited by the UAV each time. Lastly, scenario 4 had 2000 workers and 200 locations, with 20 locations visited by the UAV each time. Each scenario generates 20 rounds of data to simulate real-life location data that fluctuate by 20% from baseline data. Each location has 5 attributes, with the platform focusing on specific attributes for different locations. The results of each scenario are the means of 20 experiment results.

The workers' attribute deviation is represented by ц, and the location deviation is represented by ч. If the truthful value of an entry is T^r, the generated worker's entry value is $T^r * (1 + \text{ц}) * (1 + \text{ч})$, where ц and ч have a 10% chance of being negative. The range of ц and ч values varies depending on the type of worker: the range for malicious workers is $|[0.15,0.2]|$, the range for ordinary workers is $|[0.1,0.15]|$, and the range for truthful workers is $|[0.05,0.1]|$. The proportion of workers is 20% malicious, 20% ordinary, and 60% truthful, and the parameters for each worker type are randomly generated. In the Credibility Evaluation Model, $\sigma_s = \sigma_s^c = E_m/2$. $\rho = 4$, $\nu = 0.7$, $\Theta = 0.3$, $\gamma = 0.003$, $\theta = 0.25$, $\theta^c = 0.25$; in the SAPE model, $\delta = 4$, $\Omega = 0.25$, $\delta^c = 4$, $\Omega^c = 0.25$, $\mu = 0.003$, $\S = 0.7$; in the SLPE model, $\eta_s = \eta_s^c = E_m/2$, $\tau = 4$, $\tau^c = 4$, $\aleph^c = 0.25$, $\aleph = 0.25$, $\Gamma = 0.6$, $\varphi = 0.003$, $з = 0.01$, $Д = 3$; in Task Assignment Model, $\varrho_1 = 3$, $\varrho_2 = 3$, $\omega = 0.9$. In the first round, each location is assigned 5 workers chosen at random. After the first round, the following parameters are set: $\theta_1 = 1.8$, $\beta_1 = 4$, $\theta_2 = 3.0$, $\beta_2 = 3$, $\theta_3 = 4.2$, $\beta_3 = 2$, $\theta_4 = 5.0$, $\beta_4 = 1$.

We selected two reference schemes for comparison. The first scheme is TAFR and TFGR [8], which have state-of-the-art performance in estimating workers' reliability. The second scheme is the Random method mentioned in reference [37].

B. Experiment Result

In Fig. 3, credibility changes over time with three update mechanisms. When $\sigma_r > \sigma_s$, credibility increases and converges to 1 as update time increases, with a larger σ_r leading to a more significant increase. When $\sigma_r < \sigma_s$, the PM decreases credibility, which eventually converges to 0. Smaller σ_r in this case results in a faster decrease of credibility.

Fig. 4 depicts credible worker ratio among employed workers in a network of 100 workers. TFAR+TFGR does not filter workers after credibility evaluation, and the Random method struggles to select credible workers, resulting in no increase in the ratio of truthful workers over rounds. However, MLM-WR's GTD and Sub-GTD detect truthful workers, resulting in an increasing ratio of truthful workers among employed workers.

In Fig. 5, truthful worker recognition exponentially increases with rounds, fueled by consistent GTD evaluations and the growing pool of credible workers as Sub-GTD contributors. The curve flattens due to decreased worker hiring. This rapid credible worker detection ensures swift task assignments. In Fig. 6, the F1-score change with increasing rounds is shown. A higher ratio of UAV-detected locations to all evaluated locations leads to a faster increase in evaluated worker proportion, resulting in a quicker F1-score increase. With 200

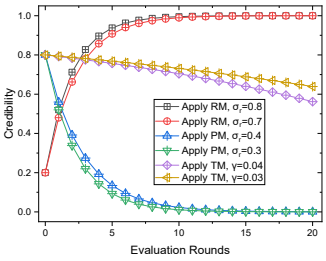


Fig. 3. Credibility change per round with $\rho = 2$

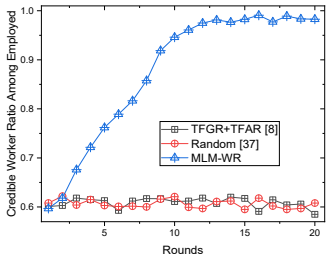


Fig. 4. Credible worker ratio among employed (100 workers)

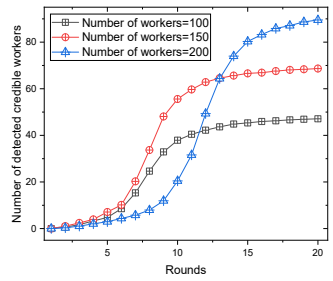


Fig. 5. Detected truthful workers

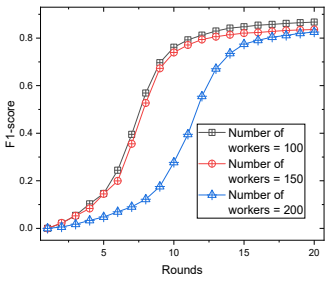
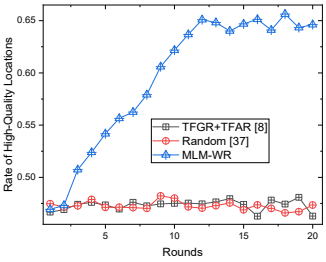
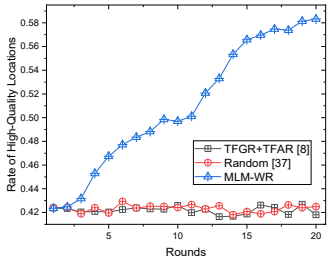


Fig. 6. F1-score



(a) 100 workers



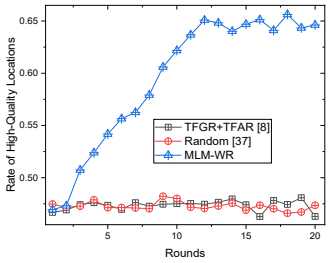
(b) 200 workers

Fig. 7. This figure displays the rate of high-quality attributes for different numbers of workers in the network. (a) represents 100 workers, and (b) represents 200 workers.

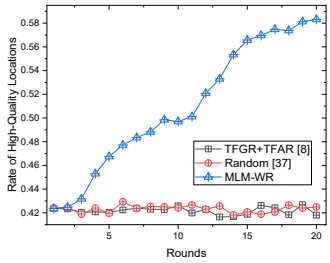
workers, the lowest ratio of locations detected by UAV results in a slower detection of truthful workers and a slower F1-score increase compared to the other cases.

Figs. 7 display the rate of high-quality attributes, which are sensing attributes of workers with attribute deviation ц in $|[0.05, 0.75]|$. The rate is calculated by dividing the number of high-quality attributes among employed workers by the product of the number of employed workers and the number of attribute categories. In MLM-WR, the rate of high-quality attributes increases as the number of rounds increases, and truthful workers with better sensing attribute quality are selected. However, the TFAR+TFGR method does not incorporate historical sensing attribute evaluations, leading to a lack of significant increase in the rate of high-quality attributes.

High-quality locations are sensing locations of workers whose location deviation ч ranges from $|[0.05,0.75]|$. The rate of high-quality locations is determined as the ratio of high-quality locations to the number of dispatched workers multiplied by the number of locations. The MLM-WR method employs a combination of absolute and relative sensing location preference. The calculation of absolute sensing location preference is similar to that of sensing attribute preference, but the Sensing Location Preference Evaluation Model is more comprehensive due to the existence of relative sensing location preference. The curve for high-quality locations is smoother than that for high-quality attributes, reflecting this difference.

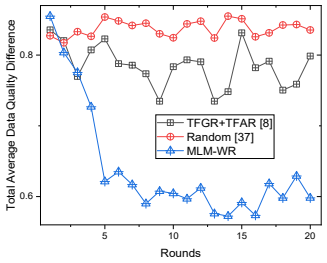


(a) 100 workers

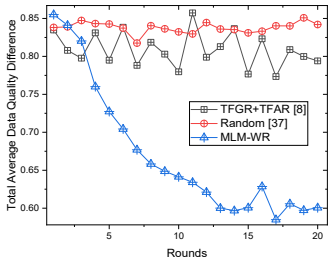


(b) 200 workers

Fig. 8. This figure displays the rate of high-quality locations for different numbers of workers in the network. (a) represents 100 workers, and (b) represents 200 workers.

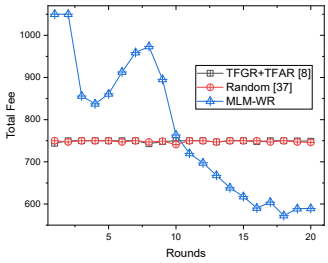


(a) Number of workers:100

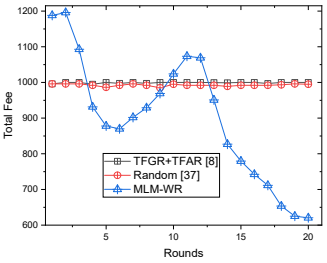


(b) Number of workers: 200

Fig. 9. This figure displays the total average data quality difference for different numbers of workers in the network. (a) is for 100 workers, and (b) is for 200 workers.



(a) Number of workers:150

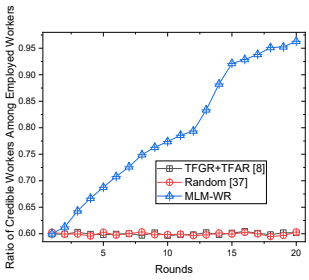


(b) Number of workers: 200

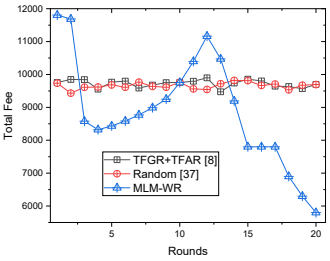
Fig. 10. This figure shows the total fee for 150 and 200 workers in the network, as depicted in (a) and (b), respectively.

Fig. 8 illustrates the change in the rate of high-quality locations in the network, showing smoother curves than those in Fig. 7.

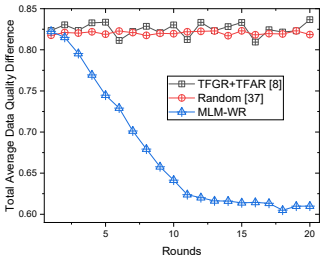
Fig. 9 illustrates the change in total average data quality difference for 100 and 200 workers within the network. MLM-WR significantly outperforms the other two methods in reducing this difference due to its enhanced worker quality and optimized assignment based on sensing attributes and locations. Its dynamic credibility evaluation mechanism surpasses TFGR+TFAR in accurately detecting credible workers. By leveraging UAV-provided GTD, MLM-WR assesses workers' sensing attribute and location reliability more precisely than TFGR+TFAR, which relies on estimated GTD. MLM-WR effectively addresses limitations of the Random method, which is susceptible to malicious data and underutilization of worker sensing abilities. In the 200-worker scenario, the decline in the UAV-detected location ratio leads to a slower reduction in the total average data quality difference compared to the 100-worker scenario. TFGR+TFAR outperforms the Random method in reducing the difference due to its attribute and location trustworthiness identification. However, TFGR+TFAR struggles with accurate credibility assessment as locations and workers increase.



(a) Ratio of Truthful Workers Among Employed Workers



(b) Total Fee



(c) Total Average Data Quality Difference

Fig. 11. The figure displays three experiments analyzing different aspects of a 2000-worker network: (a) ratio of credible employed workers, (b) total fee, and (c) total average data quality difference.

In Fig. 10, the total fee is displayed, with a cost of 100 for a UAV's sensing task and 10 for a worker's task. MLM-WR decreases the number of workers per location as credible ones are identified, leading to cost savings. When truthful workers are initially scarce, additional workers are dispatched, raising the cost. Over time, MLM-WR's cost significantly decreases, despite its higher initial cost compared to the Random and TFAR+TFGR methods.

MLM-WR was tested on a network of 2,000 workers and compared with Random and TFAR+TFGR methods. In the 20th round, MLM-WR saved 40.15% and 40.19% compared to Random and TFAR+TFGR, respectively, in terms of total fees. It also outperformed these methods by 25.48% and 27.10%, respectively, in terms of total average data quality difference. MLM-WR improved by 36.12% compared to TFAR+TFGR in recognizing truthful workers. These results demonstrate the practicality of MLM-WR in real-world scenarios. Fig. 11 provides a visual representation of the results.

VI. Conclusion

This paper proposes MLM-WR as a cost-effective approach to improve the data quality in AI-enabled MCS, which is a type of AIoT system, through cloud-edge-terminal collaboration. The Credibility Evaluation Model updates worker credibility by calculating the error of their sensing data. Then, we calculate workers' sensing attribute preference using error calculations involving Attribute-GTD and Sub-Attribute-GTD. For sensing location preference, we combine absolute and relative sensing location preferences, achieved by aligning their location data with Location-GTD and Sub-Location-GTD, and utilizing a novel optimization framework. Furthermore, we build an algorithm based on Swarm Intelligence to find the best task assignment with the consideration of total fee and data quality, which combines sensing location and attribute credibility. Experimental results demonstrate that the MLM-WR approach outperforms previous evaluation methods in accurately estimating sensing attribute and location credibility and provides a state-of-the-art task assignment approach.

Future work includes developing an incentive mechanism based on auction models to prevent malicious competition among workers. The incentive mechanism will consider the reliability of workers' sensing location and attribute to determine their payment, improving the feasibility of the MCS system and avoiding low participation rates.

References

[1] D. Wu, H. Shi, H. Wang *et al*., "A Feature-Based Learning System for Internet of Things Applications," in *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1928-1937, April 2019.

[2] Y. Liu, M. Peng, G. Shou, Y. Chen and S. Chen, "Toward Edge Intelligence: Multiaccess Edge Computing for 5G and Internet of Things," in *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6722-6747, 2020.

[3] Z. Cao, P. Zhou, R. Li, S. Huang and D. Wu, "Multiagent Deep Reinforcement Learning for Joint Multichannel Access and Task Offloading of Mobile-Edge Computing in Industry 4.0," in IEEE Internet of Things Journal, vol. 7, no. 7, pp. 6201-6213, July 2020.

[4] D. Wu, S. Si, S. Wu and R. Wang, "Dynamic Trust Relationships Aware Data Privacy Protection in Mobile Crowd-Sensing," in *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2958-2970, 2018.

[5] N. Maisonneuve, M. Stevens, M. E. Niessen *et al*., "NoiseTube: Measuring and mapping noise pollution with mobile phones," *Information Technologies in Environmental Engineering*, pp. 215–228, 2009.

[6] R. K. Rana, C. T. Chou, S. S. Kanhere *et al*., "Ear-Phone," *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, 2010.

[7] D. Wu, Z. Yang, B. Yang, R. Wang and P. Zhang, "From Centralized Management to Edge Collaboration: A Privacy-Preserving Task Assignment Framework for Mobile Crowdsensing," in *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4579-4589, 2021.

[8] H. Tian, W. Sheng, H. Shen, and C. Wang, "Truth finding by reliability estimation on inconsistent entities for heterogeneous data sets," *Knowledge-Based Systems*, vol. 187, p. 104828, 2020.

[9] J. Huang, L. Kong, L. Cheng *et al*., "BlockSense: Towards Trustworthy Mobile Crowdsensing via Proof-of-Data Blockchain," in *IEEE Transactions on Mobile Computing*, 2022.

[10] H. Jin, L. Su, and K. Nahrstedt, "Theseus: Incentivizing truth discovery in mobile crowd sensing systems" *Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2017.

[11] C. Meng, W. Jiang, Y. Li, J. Gao, L. Su, H. Ding, and Y. Cheng, "Truth discovery on crowd sensing of correlated entities," *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, 2015.

[12] Y. Zheng, G. Li, Y. Li *et al*., "Truth inference in crowdsourcing," *Proceedings of the VLDB Endowment*, vol. 10, no. 5, pp. 541–552, 2017.

[13] Y. Du, YE. Sun, and H. Huang *et al*., "Bayesian Co-Clustering Truth Discovery for Mobile Crowd Sensing Systems," in *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1045-1057, Feb. 2020.

[14] J. Guo, A. Liu, K. Ota *et al*., "ITCN: An Intelligent Trust Collaboration Network System in IoT," in *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 1, pp. 203-218, 1 Jan.-Feb. 2022.

[15] S. Ye, J. Wang, H. Fan, and Z. Zhang, "Probabilistic model for truth discovery with mean and median check framework," *Knowledge-Based Systems*, vol. 233, p. 107482, 2021.

[16] J. Picaut, A. Boumchich, E. Bocher *et al*., "A smartphone-based crowd-sourced database for environmental noise assessment," *International Journal of Environmental Research and Public Health*, vol. 18, no. 15, p. 7777, 2021.

[17] T. Liang, L. Chen, M. Huang *et al*., "RLTD: A reinforcement learning-based Truth Data Discovery Scheme for decision support systems under Sustainable Environments," *Applied Soft Computing*, vol. 143, p. 110369, 2023.

[18] F. Lyu, J. Ren, N. Cheng *et al*., "LeaD: Large-Scale Edge Cache Deployment Based on Spatio-Temporal WiFi Traffic Statistics," in *IEEE Transactions on Mobile Computing*, vol. 20, no. 8, pp. 2607-2623, 2021.

[19] T. Luo, J. Huang, S. S. Kanhere *et al*., "Improving IoT Data Quality in Mobile Crowd Sensing: A Cross Validation Approach," in *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5651-5664, June 2019.
[20] X. Tao and A. S. Hafid, "ChainSensing: A Novel Mobile Crowdsensing Framework With Blockchain," in *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2999-3010, 2022.
[21] J. Wang, L. Wang, Y. Wang, D. Zhang, and L. Kong, "Task Allocation in Mobile Crowd Sensing: State-of-the-Art and Future Opportunities," in *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3747-3757, Oct. 2018.
[22] W. Gong, B. Zhang and C. Li, "Task Assignment in Mobile Crowdsensing: Present and Future Directions," in *IEEE Network*, vol. 32, no. 4, pp. 100-107, July/August 2018.
[23] G. Manogaran, A. Alshaikh, R. Alsabet, and D. B. Rawat, "Replication based crowd sensing for optimal service response in 5G Communications using information-centric wireless sensor networks," *2021 IEEE 22nd International Conference on Information Reuse and Integration for Data Science (IRI)*, 2021.
[24] A. Hamrouni, H. Ghazzai, T. Alelyani *et al*., "Low-Complexity Recruitment for Collaborative Mobile Crowdsourcing Using Graph Neural Networks," in *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 813-829, 2022.
[25] J. Zhang and D. Tao, "Empowering Things With Intelligence: A Survey of the Progress, Challenges, and Opportunities in Artificial Intelligence of Things," in *IEEE Internet of Things Journal*, vol. 8, no. 10, pp. 7789-7817, 2021.
[26] S. Duan, D. Wang, J. Ren *et al*., "Distributed Artificial Intelligence Empowered by End-Edge-Cloud Computing: A Survey," in *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 591-624, Firstquarter 2023.
[27] T. Li, A. Liu, N. N. Xiong *et al*., "A trustworthiness-based vehicular recruitment scheme for information collections in Distributed Networked Systems," *Information Sciences*, vol. 545, pp. 65–81, 2021.
[28] C. Zhao, S. Yang, and J. A. McCann, "On the data quality in privacy-preserving mobile crowdsensing systems with untruthful reporting," *IEEE Transactions on Mobile Computing*, vol. 20, no. 2, pp. 647–661, 2021.
[29] P. K. Singh, R. Singh, S. K. Nandi, K. Z. Ghafoor, D. B. Rawat *et al*., "Blockchain-Based Adaptive Trust Management in Internet of Vehicles Using Smart Contract," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3616-3630, June 2021.
[30] P. Inkeaw, P. Udomwong, and J. Chaijaruwanich, "Density based semi-automatic labeling on multi-feature representations for ground truth generation: Application to handwritten character recognition," *Knowledge-Based Systems*, vol. 220, p. 106953, 2021.
[31] Y. Zhao, X. Gong, F. Lin and X. Chen, "Data Poisoning Attacks and Defenses in Dynamic Crowdsourcing With Online Data Quality Learning," in *IEEE Transactions on Mobile Computing*, vol. 22, no. 5, pp. 2569-2581, 1 May 2023.
[32] S. Liu, Z. Zheng, F. Wu, S. Tang, and G. Chen, "Context-aware data quality estimation in Mobile Crowdsensing," *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, 2017.
[33] J. Gao, S. Fu, Y. Luo, and T. Xie, "Location Privacy-Preserving Truth Discovery in mobile crowd sensing," *2020 29th International Conference on Computer Communications and Networks (ICCCN)*, 2020.
[34] Z. Ma, H. Li, W. Fang *et al*., "A Cloud-Edge-Terminal Collaborative System for Temperature Measurement in COVID-19 Prevention," *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2021, pp. 1-6.
[35] Z. Yu, Z. Si, X. Li, D. Wang, and H. Song, "A Novel Hybrid Particle Swarm Optimization Algorithm for Path Planning of UAVs," in *IEEE Internet of Things Journal*, vol. 9, no. 22, pp. 22547-22558, 2022.
[36] National Institute of Diabetes and Digestive and Kidney Diseases, 2020, "Diabetes Dataset",https://www.kaggle.com/datasets/mathchi/diabetes-data-set, Kaggle.
[37] C.-J. Ho and J. Vaughan, "Online task assignment in Crowdsourcing Markets," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 26, no. 1, pp. 45–51, 2021.

Jiaheng Lu is with the School of Computer Science and Engineering, Central South University, China. His research interests include wireless sensor networks. E-mail: jiahenglu@csu.edu.cn.

Zhenzhe Qu received his master's degree in the School of Software, Central South University, China, in 2019. He is currently pursuing his Ph.D. degree at the School of Computer Science and Engineering, Central South University, China. His research interests include edge computing. E-mail: zhenzheQu@csu.edu.cn.

Anfeng Liu received M.Sc. and Ph.D. degrees from Central South University, China, in 2002 and 2005, respectively, both in computer science. He is currently a professor at the School of Computer Science and Engineering, Central South University, China. His major research interests include the Internet of Things, Information Security, and Crowdsourcing. E-mail: afengliu@mail.csu.edu.cn.

Shaobo Zhang received his B.Sc. and M.Sc. degrees in computer science from Hunan University of Science and Technology, Xiangtan, China, in 2003 and 2009, respectively, and his Ph.D. degree in computer science from Central South University, Changsha, China, in 2017. He is currently a professor at Hunan University of Science and Technology, China, focusing on research in privacy, security, and cloud computing. E-mail: shaobozhang@hnust.edu.cn.

Neal N. Xiong (S'05–M'08–SM'12) is currently an Associate Professor (4-year credits) at the Department of Computer Science and Mathematics, Sul Ross State University, Alpine, TX 79830, USA. He received both his Ph.D. degrees from Wuhan University (2007) and Japan Advanced Institute of Science and Technology (2008), respectively. Before attending Sul Ross State University, he worked at Georgia State University and Northeastern State University. His research interests include Cloud Computing, Security and Dependability, Parallel and Distributed Computing. E-mail: xiongnaixue@gmail.com