DLFTI: A Deep Learning based Fast Truth Inference Mechanism for Distributed Spatiotemporal Data in Mobile Crowd Sensing

Jianheng Tang[a], Kejia Fan[a], Pengzhi Yin[b], Zhenzhe Qu[a], Anfeng Liu[a,*], Neal N. Xiong[c,*], Tian Wang[d], Mianxiong Dong[e], Shaobo Zhang[f]

[a] *School of Computer Science and Engineering, Central South University, Changsha, 410083, China*

[b] *School of Automation, Central South University, Changsha, 4100083, China*

[c] *Department of Computer Science and Mathematics, Sul Ross State University, Alpine, TX 79830, USA*

[d] *Department of Artificial Intelligence and Future Networks, Beijing Normal University & UIC, Zhuhai, Guangdong, China*

[e] *Department of Information and Electronic Engineering, Muroran Institute of Technology, Muroran, 050-8585, Japan*

[f] *School of Computer Science and Engineering of the Hunan University of Science and Technology, Xiangtan, 411201, China*

ARTICLE INFO

Keywords:
Mobile crowd sensing
Distributed spatiotemporal data
Truth inference
Deep learning
Trust computing

ABSTRAC

The paradigm of Mobile Crowd Sensing (MCS) allows for numerous applications with distributed spatiotemporal data, where great attention is drawn to the fundamental problems for truth inference. Existing works all suffer from poor accuracy and slow start, due to the lack of valid information for Ground Truth Data (GTD) and workers. These problems make the MCS system vulnerable to fraudulent attacks by malicious gangs, causing the Estimated Truth Data (ETD) to deviate significantly from GTD. In this paper, we propose a Deep Learning based Fast Truth Inference mechanism, called DLFTI, to achieve fast trust computing and accurate truth discovery in MCS. First, we introduce the Degrees-Of-Trust (DOT) to characterize the sensing ability of workers and establish worker profiles based on DOT to recognize workers' trustworthiness dynamically. Then, we abandon the unrealistic assumption of priori GTD in previous studies and instead utilize the Unmanned Aerial Vehicles (UAVs), recognized trustworthy workers and the Deep Matrix Factorization (DMF) method to construct three-level GTD and three-level ETD, which are used for fast trust computing of workers and accurate truth discovery of tasks respectively. Finally, we conduct extensive simulations on a real-world dataset to corroborate the significant performance of DLFTI.

1. Introduction

Mobile Crowd Sensing (MCS) has emerged as a promising data collection paradigm for monitoring the real world with embedded sensors in mobile devices, such as mobile iPhone, iPad and vehicle sensing device [1]–[3]. The mobile devices are used to form real-time, large-scale systems to perform data sensing and task calculations due to their powerful capabilities in large-scale and long-duration data sensing. The MCS platforms recruit workers to sense objects and upload their sensing data, and then provide data-based services to consumers (or users) [4]-[6]. MCS applications of all kinds have mushroomed in recent years, including NoiseTube [7], Ear-phone [8], SignalGuru [9] and CrowdAtlas [10]. The success of MCS applications depends on the massive data contributed by the workers to ensure service quality [11]-[13]. However, due to the presence of untrustworthy or malicious employees, the MCS system's data quality problem has become a significant issue [14]-[16]. As a result, a hot research topic called truth discovery has come out in recent years [17]- [21].

Many mechanisms have been proposed to address the issue of truth discovery [22]-[24]. The most significant type of it is based on mathematical statistics, such as the Mean method, Median method, and Weighted Mean method [25]. However, such methods have the following shortcomings [25]: First, since platforms do not have Ground Truth Data (GTD) to check the accuracy of the reported data [26], the system is vulnerable to fraudulent attacks by malicious gangs. For example, when multiple malicious workers jointly report similar fake data for the same object, the platform is prone to mistake the fake data reported by attackers as trustworthy. Second, recruiting redundant workers for one task causes much unnecessary cost.

GTD refers to the true and reliable data, which is the ground truth of the data sensing task. In MCS, GTD can serve as the benchmark to verify the quality of data reported by workers and evaluate their task completion quality. And the final goal of the platform is to fully integrate all information about the workers and the tasks to estimate the most accurate truth possible, namely Estimated Truth Data (ETD).

Since the existing truth discovery methods without GTD are unreliable in accuracy and vulnerable to attacks. Subsequent studies have proposed methods using GTD to perform truth discovery [27]. In these studies, it is assumed that platform can obtain a small percentage of GTD [28], based on which the trust computing is performed. If the data reported by the workers is consistent with GTD, their Degrees-Of-Trust (DOT) will be increased, and if not, their DOT will be decreased. Then the platform can focus on recruiting recognized trustworthy workers whose DOT is relatively high to obtain more credible data.

∗ Corresponding author.
E-mail address: tangent-heng@csu.edu.cn (J. Tang), kejiafan@csu.edu.cn (K. Fan), pengzhiyin@csu.edu.cn (P. Yin), zhenzheQu@csu.edu.cn (Z. Qu), afengliu@mail.csu.edu.cn (A. Liu), xiongnaixue@gmail.com (N. Xiong), tianwang@bnu.edu.cn (T. Wang), mx.dong@csse.muroran-it.ac.jp (M. Xiong), shaobozhang@hnust.edu.cn (S. Zhang).

However, there are still some problems with such approaches: First, in practical MCS, GTD is subject to the strict condition that only data sensed for the same object within the same time and place range can be used as GTD [27], preventing the mass adoption of this approach. Second, this method suffers from the problem of a slow start, for it takes a long time to make the trust extension and evolution.

Hence, there are three major challenges in the truth inference mechanism. First, since there may exist malicious gangs in the worker population, the platform should obtain as many credible data benchmarks like GTD as possible, making full use of available sensing devices, data reported by workers, and the spatiotemporal correlation between the distributed data. Second, in real scenarios, recruited workers may be either trustworthy or untrustworthy, so there is an urgent need to fast distinguish trustworthy and untrustworthy workers by comparing their historical reported data with the data benchmark. Third, it is inevitable for the platform to recruit untrustworthy workers who submit fake data, so it is a huge challenge to conduct truth discovery for downstream applications from the conflicting data reported by workers.

To address the above challenges, this paper innovatively proposes a Deep Learning based Fast Truth Inference (DLFTI) mechanism for MCS. First, we send Unmanned Aerial Vehicles (UAVs) to collect data for the same tasks in parallel with workers and regard the data sensed by the UAVs as gold GTD. Since the number of UAVs is controllable, we can easily obtain sufficient GTD while controlling the system's start-up speed. Second, we create the worker profiles to dynamically record the recognition of workers and use the recognized highly trustworthy workers in it as silver GTD providers, which means the data reported by them is also regarded as trustworthy. This further increases the number of GTD and speeds up the system's response time. Third, we introduce the Deep Matrix Factorization (DMF) algorithm to obtain bronze GTD to cover all tasks, which gives all tasks credible judging criteria and avoids gang attacks by malicious workers. Finally, we achieve fast trust computing for workers to update worker profiles based on the three-level GTD, and then further design a three-level ETD for high-precision truth discovery.

Overall, the major contributions are summarized as follows:

1) We propose a Deep Learning based Fast Truth Inference mechanism, namely DLFTI. To the best of our knowledge, this is the first work that systematically addresses the problem of fast and accurate truth inference in MCS systems based on the spatiotemporal correlation between the distributed data.
2) Abandoning the unrealistic assumption of priori GTD from previous studies, we propose for the first time the multi-level GTD framework for trust computing. This approach can avoid the negative impact of gang attacks by malicious workers and ensures the security of the MCS system.
3) Based on the multi-level GTD and worker recognition, we further propose the multi-level ETD framework for truth discovery, which can ensure the accuracy of inferred truth even when several untrustworthy workers are recruited, thus improving the robustness of DLFTI.
4) Extensive simulations on the real-world dataset are conducted to demonstrate the significant performance of DLFTI. The results show that the performance presented by our DLFTI exceeds that of all the compared algorithms under all simulated scenarios.

The rest of this paper is organized as follows. Section 2 introduces related works. The system model and definitions are presented in Section 3. In Section 4, the DLFTI mechanism is proposed. Then, Section 5 provides the performance analysis. Finally, the conclusion and future work are given in Section 6.

2. Related Works

In MCS, to incentivize workers to offer high-quality data, it is necessary to give them a certain amount of rewards [29]-[33]. For example, sometimes workers need to move to a specified destination for data sensing and then report the data, which requires energy, communication bandwidth, and other resources [14], [34]-[35]. But there are some untrustworthy or malicious workers to report false data, aiming at getting the reward without any labor or attacking the system [27], [28], [36]. Therefore, calculating the qualities of workers' sensing data and distinguishing trustworthy workers from untrustworthy ones is very significant [37]-[39]. The following is a discussion of some researches related to this paper.

2.1. Methods based on mathematical statistics

The major methods of this type are Mean method, Median method, Weighted Mean method and Majority Voting method [23]. The basic process of these approaches is that the platform recruits multiple workers to sense the same object. The Mean and Median methods assume that trustworthy workers in the network are in the majority, so the inferred truth could be calculated as the mean or medium of the data reported by all workers. The Weighted Average method considers that the reported data generally obey a normal distribution, and thus the data closer to the center of the distribution reported by workers are closer to the real truth. The disadvantage of this method is that the system does not know the ground truth data (GTD) [26], so the accuracy of the results may be very low, which makes it vulnerable to joint attacks by multiple malicious workers. What's more, such methods are costly and slow in worker recognition, making them unsuitable for large-scale data collection scenarios.

2.2. Methods based on priori GTD

From the above discussion, comparative GTD is the key to worker recognition. Therefore, some researchers have proposed a class of priori GTD based methods [28], [29]. In these methods, the platform can assign workers to collect specific data with priori GTD to evaluate the workers' DOT [28]. The gap between trustworthy and untrustworthy workers will become wide enough after several cycles so that the platform can easily distinguish between them [29]. In the approach described above,

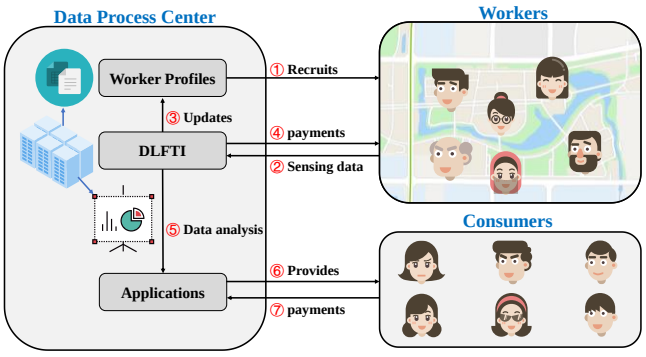


Fig. 1. Data collection model in MCS.

the number of GTD directly affects how long it will take to recognize the workers. If the number of GTD is minimal, only a small number of unknown workers can be recognized each time, making the process time-consuming. Due to the frequent joining and leaving of workers in MCS, the workers' DOT is continually shifting over time. As a result, the applications' requirements are not met by the lengthy time it takes to identify workers, which results in subpar system performance. Similarly, obtaining GTD in practical MCS systems is difficult and expensive, so the assumption of existing priori GTD is unreasonable. Based on this unrealistic assumption, these studies are difficult to apply to real MCS systems.

2.3. Methods based on GTD acquired by UAVs

In all the described studies, it is assumed that priori GTD is available. But in practice, obtaining GTD is not easy, and the system has to pay for it. Guo et al [29] proposed an effective method to collect some GTD through UAVs targeted for truth discovery. This type of method solves the problem of how to obtain GTD [29]. However, it can only gain a small amount of GTD [41]. As long as the speed of recognized trustworthy workers exceeds the speed of trustworthy worker exits, an increasing number of trustworthy workers will be recognized in the network. And after a period of time, we will have a sufficient number of trustworthy workers to collect trustworthy data, which makes the high-precision truth discovery possible. Nevertheless, this requires a stable network and a long time, both of which limit the performance of such methods. However, because the system can actively acquire GTD, it can still correctly identify workers when attacked by gangs of malicious workers, thus avoiding system crashes caused by gangs of malicious workers.

3. System Model and Definitions

3.1. System model

The MCS network used in this paper is similar to most of the crowdsourcing networks studied, as shown in Fig. 1. Specifically, it has the following components: (1) Data Process Center (DPC). DPC firstly issues sensing tasks to workers, then collects the data reported by workers, and finally analyzes the data to conduct various applications. (2) Workers. They refer to a large number of people with sensing devices in the Internet of Things. They are numerous and mobile, so they can move very economically to a specified location after receiving tasks to sense the data and submit the sensed data to the DPC. The DPC rewards the workers who submit the data as an incentive. (3) Consumers. The DPC synthesizes the data into applications and makes them available to consumers, thereby gaining revenue from the consumers to cover the cost of collecting and processing data.

In this paper, we focus on the interaction between DPC and workers. That is, once the DPC allocates tasks to unknown workers, how to evaluate the workers' trustworthiness and infer the truth data for downstream analysis, based on the data submitted by workers without a priori GTD. The tasks and unknown workers are defined as follows:

Definition 1 (Task, Round, Cycle). For the needs of downstream analysis, the DPC issues m distributed spatiotemporal data collection tasks, denoted by $W = \{w_1, w_2 \dots, w_m\}$. The data collection is divided into multiple rounds, denoted by $t \in \{1, 2, \dots\}$. And each τ round is noted as a cycle, denoted by $C = \{c_1, c_2, \dots\}$, where $c_k = \{(k-1)\tau + 1, (k-1)\tau + 2, \dots, k\tau\}$.

Definition 2 (GTD). GTD refers to the true and reliable data, which is the ground truth of the data sensing task. In MCS, GTD can serve as the benchmark to verify the quality of data reported by workers and evaluate their task completion quality. Meanwhile GTD can also be used to evaluate the accuracy of MCS system data. In this paper, we denote the GTD of all tasks in each round t as V_t^G, where $V_t^G = \{v_{1,t}^G, v_{2,t}^G, \dots, v_{m,t}^G\}$.

Definition 3 (Unknown worker, Sensing Quality). There are n unknown workers, denoted by $S = \{s_1, s_2 \dots, s_n\}$. The DPC will recruit μ workers for every task per round, and R_j^t represents the recruiting set for w_j in the t-th round, so that $|R_j^t| = \mu$ is satisfied. The data reported by worker s_i for task w_j the t-th round is denoted as $v_{j,t}^i$. Since the tasks assigned to each worker are not fixed, $\varphi_{j,t}^i \in \{0, 1\}$ is used to indicate whether worker s_i is assigned to finish task w_j in the t-th round. Due to the precision of the sensing device, $v_{j,t}^i$ is usually unequal to $v_{j,t}^G$. So, the sensing quality of the worker needs to be defined to characterize the worker's data quality of the task. The sensing quality of worker s_i completing task w_j in the t-th round is denoted as $q_{i,j}^t \in [0,1]$, where the higher value indicates better sensing quality.

Considering the real situation of MCS, there are both trustworthy workers and untrustworthy workers. In this paper, we use a trust-based score to classify workers into different classes, denoted as Degree Of Trust (DOT). Specifically, we calculate workers' DOT by their sensing quality every cycle, i.e., when a worker shows a high sensing quality in one cycle, then he can gain a high DOT in that cycle. Otherwise, the data he reports is considered false or malicious and thus he gets a relatively low DOT in that cycle. This way, after a worker has performed multiple cycles of tasks, we can calculate his composite DOT based on his historical DOT. The DOT and multiple types of workers are defined as follows:

Definition 4 (DOT). Since a worker may perform multiple tasks in a given cycle, we use $Q_k = \{q_{1,k}, q_{2,k}, \dots, q_{n,k}\}$ to denote the DOT for all workers in the k-th cycle, taking into account the performance of all the tasks that the workers perform in the k-th cycle. Furthermore, after the k-th cycle, each worker has a composite DOT, denoted by $\hat{Q}_k = \{\hat{q}_{1,k}, \hat{q}_{2,k}, \dots, \hat{q}_{n,k}\}$, which integrates all the DOT of history.

For ease of reference, we have classified and summarized the major notations and abbreviations used in this paper in Table 1 and Table 2.

Table 1. Description of major notations.

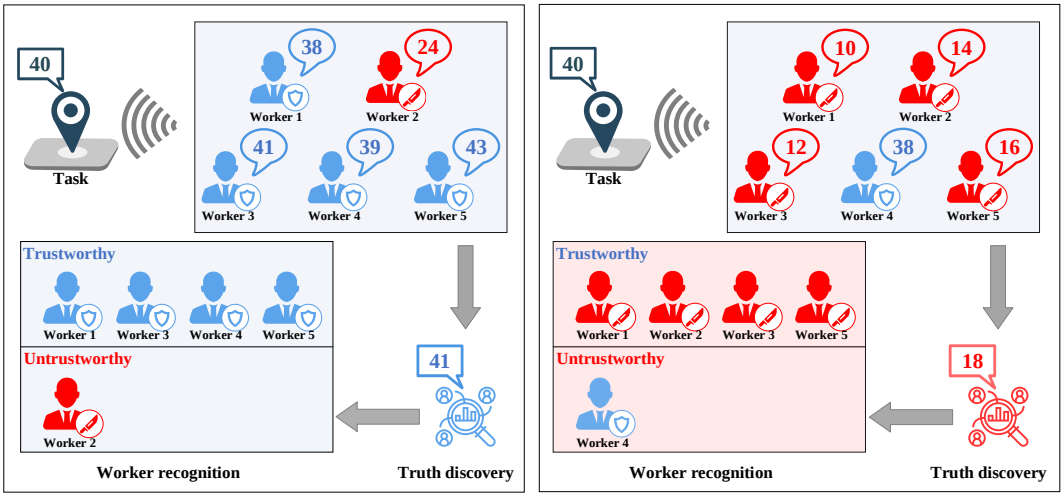
Notations	Description
m	Number of tasks
n	Number of workers
τ	Number of rounds per cycle
μ	Number of recruited workers for every task
$v_{j,t}^{i}$	Reported data by s_i for w_j in the t-th round
$q_{i,j}^{t}$	Sensing quality of s_i for w_j in the t-th round
$q_{i,k}$	DOT of s_i in the k-th cycle
$\hat{q}_{i,k}$	Composite DOT of s_i in the k-th cycle
$v_{j,t}^{G}$	Real GTD of w_j in the t-th round
$v_{j,t}^{UAV}$	UAVs' report Data for task w_j in the t-th round
$v_{j,t}^{Gold}$	Gold GTD for task w_j in the t-th round
$v_{j,t}^{Silver}$	Silver GTD for task w_j in the t-th round
$v_{j,t}^{Bronze}$	Bronze GTD for task w_j in the t-th round
$u_{j,t}^{E}$	ETD of w_j in the t-th round
$u_{j,t}^{Gold}$	Gold ETD for task w_j in the t-th round
$u_{j,t}^{Silver}$	Silver ETD for task w_j in the t-th round
$u_{j,t}^{Bronze}$	Bronze ETD for task w_j in the t-th round

Table 2. Description of the abbreviations.

Abbreviations	Description
MCS	Mobile Crowd Sensing
DPC	Data Process Center
UAV	Unmanned Aerial Vehicle
GTD	Ground Truth Data
Gold GTD	Data provided by the UAV
Silver GTD	Data provided by the highly trustworthy workers
ETD	Estimated Truth Data
DOT	Degrees-Of-Trust
DMF	Deep Matrix Factorization

3.2. Problem statement

In the MCS system, the DPC needs to recruit many workers to complete some data collection tasks assigned within a specific budget. Workers inevitably incur costs when completing tasks, such as traffic costs, network costs, and time costs. Malicious workers may therefore benefit more by directly submitting false data instead of honestly performing the sensing task, to receive the same reward without the cost. To ensure the quality of collected data, the platform often recruits multiple



(a) A conventional case (b) A special case of malicious group attack

Fig. 2. The sensing system in DLFTI.

workers for the same task and aggregates their uploaded data to infer the truth and identify malicious workers. However, some malicious workers will join together to attack the DPC. When malicious workers are in the majority, the platform will assume that the data submitted by the malicious workers is correct and the data submitted by other workers is false. At this point, the MCS system breaks down, and all the inference results are untrustworthy.

In the conventional case, as shown in Fig. 2 (a), the GTD of a sensing task is 40, and the platform recruits five workers to complete this sensing task. The sensing results reported to the platform by these five workers are respectively 38, 24, 41, 39, and 43. Since the majority of recruited workers are trustworthy (i.e., four out of five), the data reported by them will be around the GTD. Therefore, it is easy for the platform to directly conduct truth inference by multiple statistical methods. For example, if we use the mean method, we can get an acceptable result (i.e., 37), which is closer to the GTD (i.e., 40). Then trustworthy workers (38, 41, 39, 43) will receive higher DOT because their data is close to the final inferred truth. Meanwhile, the untrustworthy worker (24) will be recognized because of the large distance between his data and the inferred truth, and thus the platform will reduce his trustworthiness. Thus, the whole system will gradually converge and eventually have an accurate judgment on the DOT of each worker.

Now considering a case where the platform suffers a malicious group attack, as shown in Fig. 2 (b). The GTD for this task is still 40, but most workers (i.e., four out of five) recruited by the platform are colluding with each other in a malicious attack group. As a result, the perceived results reported to the platform are 10, 14, 12, 38 and 16. At this point, since untrustworthy workers are in the majority of the worker group, the previous assumption for statistical analysis is no longer satisfied. If we directly use statistical analysis, such as the mean method to conduct truth inference, the final result (i.e., 18) will have a huge gap with GTD. This makes the inferred truth obtained this time unavailable, which will drastically reduce the performance of the application. More seriously, it will directly lead to serious errors in the results of worker recognition. Specifically, at this point, we will give higher DOT to malicious workers (10, 14, 12, 16) and will give lower DOT to trustworthy workers (38), thus treating malicious workers as trustworthy and good workers as untrustworthy. This will undoubtedly lead to a chain reaction, making the data subsequently obtained by the platform false, which in turn makes the whole MCS system crash.

Based on the above analysis, we hope to develop a robust and secure model to ensure that the DPC can obtain the most truthful data and the most accurate estimated truthful data, achieving a higher quality application, while being able to avoid gang attacks by malicious workers. Specifically, we can judge the strengths and weaknesses of the solution to the problem by the following 2 indicators:

(1) **Worker Recognition.** The ability to accurately distinguish the workers' DOT without prior GTD. In general, workers with high DOT tend to submit real data close to the ground truth, while workers with low DOT are more likely to submit malicious data. The use of data reported by malicious workers can be detrimental to the quality of our downstream applications, so we prefer to assign tasks to workers with high DOT to obtain high quality data.

(2) **Truth Discovery.** The ability to estimate the ground truth based on the uneven data submitted by workers. Owing to the limitations of the instrumentation used by workers to collect data, the subjectivity of the workers' observations and the variability of the task itself, there are large or small errors between the data submitted by the workers and the ground truth. We are expected to obtain inferred truth as close to the real truth as possible from the error-prone data submitted by workers.

The specific evaluation methods of the above two indicators are shown below.

(1) Worker Recognition Metrics

Obviously, recruiting trustworthy workers will improve the accuracy of inferred truth, while recruiting untrustworthy workers may lead to a large error in inferred truth. So, we want to distinguish trustworthy workers from untrustworthy ones

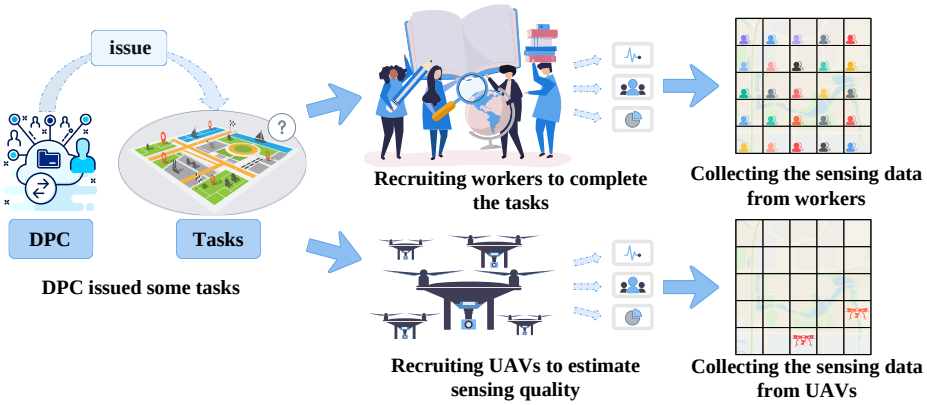


Fig. 3. The sensing system in DLFTI.

as accurately and fast as possible, thus giving a guideline for worker selection.

In this paper, we use the Receiver Operating Characteristic (ROC) curve to characterize the effect of worker recognition, which has a nice property: the ROC curve can remain constant when the distribution of positive and negative samples in the worker set changes. In the real situation, the ratio of trustworthy workers to untrustworthy workers is not fixed, so using the ROC curve as an indicator can eliminate the effect of class imbalance.

In order to present the comparison results of the ROC curve more intuitively, we also use the value of Area Under Curve (AUC) to characterize the effect of worker recognition, which is defined as the area under the ROC curve. A larger value of AUC means that the method's worker recognition is better. In particular, when the AUC of one algorithm equals to 0.5, it means that the algorithm is no different from completely random classification.

(2) Truth Discovery Metrics

Before defining truth discovery metrics, truth discovery results should be defined first. And Estimated Truth Data (ETD) aggregated by reported data is used as the result of truth discovery, which is defined as follows.

Definition 5 (ETD). The estimated data is the ground truth estimation by DPC after receiving the data reported by all workers. In MCS, we need to perform worker identification and truth discovery, where we use GTD as the basis for worker identification. After worker identification, we infer an estimate for the task based on the identified worker information, noted as ETD, which serves as our truth discovery result. We use $U_t^E = \{u_{1,t}^E, u_{2,t}^E, \dots, u_{m,t}^E\}$ denote the ETD for all m tasks in the t-th round, and we expect that the closer U_t^E is to V_t^G, the better.

Root Mean Square Error (RMSE) is used as a metric to evaluate the accuracy of ETD in each cycle, which is defined as

$$RMSE = \sqrt{\frac{\sum_{t=1}^{\tau}\sum_{j=1}^{m}\left(v_{j,t}^G - u_{j,t}^E\right)^2}{m\tau}}. \tag{1}$$

In the whole cycle, the smaller RMSE indicates the higher accuracy of our three-level ETD framework. Meanwhile, Finish Rate (FR) is used as a measure of how well the tasks are completed, which is a coarse-grained metric in the real world. In our study, we assume that task collection can tolerate a certain degree of error, i.e., as long as the reported data is within a certain error range, then it is considered to be correctly completed. Whether task w_j is finished correctly in t-th round is denoted by $\varphi_{j,t}^{Finish}$ which is defined as below.

$$\varphi_{j,t}^{Finish} = \begin{cases} 1, u_{j,t}^E \in \left[v_{j,t}^G - \varepsilon_m, v_{j,t}^G + \varepsilon_m\right], \\ 0, else. \end{cases} \tag{2}$$

Here ε_m denotes the task's error tolerance range, so FR can be calculated by Eq. (3).

$$FR = \frac{\sum_{t=1}^{\tau}\sum_{j=1}^{m}\varphi_{j,t}^{Finish}}{m\tau}. \tag{3}$$

4. The design and analysis of DLFTI scheme

In this section, we propose the fast truth inference mechanism DLFTI, shown as Fig. 3 and Algorithm 1. We first introduce the basic idea of DLFTI, and then present the multi-level GTD system, followed by the worker recognition system. Finally, we describe how to perform high-precision truth discovery using multi-level ETD based on DLFTI.

4.1. Preliminary and Basic Idea

Existing truth discovery methods based on statistics all assume that trustworthy workers in the MCS are in the majority, so they used methods like Mean to approximate the GTD by all reported data. Because the system does not actually know the GTD, the accuracy of the results may be very low and vulnerable to joint attacks by multiple malicious workers. Imagine an extreme situation where a task is assigned to a group of malicious workers, and the malicious workers jointly fabricate data to

cheat for rewards. In such a situation, the DPC will suffer from joint fraud attacks, and the final truth inference results will be seriously distorted. Furthermore, none of the above methods has established a perfect mechanism for data completion. Suppose part of tasks cannot be completed with valid data due to unexpected conditions. In that case, none of the existing methods can make truth inference for the missing data, resulting in reducing the effectiveness of the overall truth inference.

Therefore, to avoid relying exclusively on worker-reported data, a basic idea is to use data provided by UAVs and highly trustworthy workers as an objective and credible benchmark to judge the quality of all worker-reported data, shown as Fig. 3. The UAVs and highly trustworthy workers are defined as follows.

Definition 6 (UAV). With the emergence of driverless technology and aviation systems, using UAVs for sensing data has become a feasible and accurate way of MCS. UAVs can provide efficient, flexible, and low-cost data sensing services for applications. Specifically, after receiving a data sensing task, the platform can dispatch UAVs to the designated area and establish a communication and sensing system. Through the professional data measurement equipment carried by the UAVs, they can perceive and collect the data of the task site and submit it to the server. As UAVs can be equipped with professional data measurement equipment, and have high maneuverability and flexibility, the data collected by UAVs generally has extreme accuracy. Therefore, data sensed by UAVs can be considered as the posteriori baseline for specific tasks. Specifically, UAVs are dispatched to collect data synchronously with workers for the same tasks, and dynamically modify the workers' DOT by comparing the data reported by them and UAVs. $\varphi_{j,t}^{UAV}$ is used to indicate whether there is a UAV dispatched for task w_j in the t-th round, and $v_{j,t}^{UAV}$ denote the data collected. We assume that the data collected by the UAVs is very close to the GTD, that is $\left|v_{j,t}^{G} - v_{j,t}^{UAV}\right| < \varepsilon$, where ε is a tiny positive number.

Definition 7 (Highly Trustworthy Worker). We define a threshold θ_1, and if the composite DOT of a trustworthy worker s_i satisfies $\hat{q}_{i,k} \geq \theta_1$ in the k-th cycle, he would be considered a highly trustworthy worker. As highly trustworthy workers are a special case of trustworthy workers at a higher composite DOT, highly trustworthy workers are considered more trustworthy than general trustworthy workers. A set S_k^{HT} is maintained to contain all the highly trustworthy workers in the k-th cycle.

Because of the high quality of the data reported by the UAVs and highly trustworthy workers, we consider the data obtained from these two components as the gold and silver GTD respectively, which are defined as follows:

Definition 8 (Gold GTD, Silver GTD). The data collected by the distributed UAVs for task w_j in the t-th round is denoted as $v_{j,t}^{Gold}$. Note that not all tasks have corresponding gold GTD, so we let $\varphi_{j,t}^{Gold} \in \{0,1\}$ indicate whether there is gold GTD for task w_j in the t-th round. Similarly, the data collected by highly trustworthy workers for task w_j in the t-th round is denoted as $v_{j,t}^{Silver}$. Likewise, for the silver GTD,$\varphi_{j,t}^{Silver} \in \{0,1\}$ indicates whether there is silver GTD for task w_j in the t-th round.

Combining the above analysis, the basic idea includes the following steps:

1) All workers have initial DOT, and in the process of recruiting workers for data collection, a small number of UAVs will also be dispatched to finish the same tasks as the workers do.
2) The data collected by the UAVs are used as the gold GTD to verify the data submitted by the workers and to update the workers' DOT. After a short period of time, we can initially distinguish the trustworthy workers from the untrustworthy ones, and further get a group of workers with extremely high DOT, called highly trustworthy workers.
3) Then, the data collected by the highly trustworthy workers is used as the silver GTD to verify the data submitted by other workers, thus speeding up the worker recognition.

In this way, through the joint accelerated verification of gold and silver GTD, we can quickly and dynamically update the workers' DOT. Meanwhile, we can recruit recognized trustworthy workers to collect data, thus improving the accuracy of truth discovery.

However, the above method still suffers from the problem of the slow start. In large-scale MCS, the scope of data collection is huge that the volume of data is usually in the millions, and the number of potential workers is in the tens of millions, while the number of UAVs is only in the hundreds. Therefore, it takes a long time to get the workers' DOT; thus, the number of recognized trustworthy workers remains a relatively low level for a long time. To finish all the tasks, the DPC will have to recruit many unknown workers, which makes the system vulnerable to attacks. In a MCS network where workers change dynamically, if the system cannot recognize workers as fast as the system changes, the system will remain with many unknown workers for a long time, making the system's performance very poor. Another drawback is that dispatching UAVs to collect data for comparison over a long period of time makes the system costly.

The performance of the above method is not good enough because the amount of data collected by UAVs is limited, so it is impossible to produce enough gold GTD to cover all the tasks. And at the beginning of the system, there is no recognized highly trustworthy worker to provide silver GTD, so there is little comparative GTD to evaluate the sensing quality of workers, resulting in the slow start. To better illustrate these problems, we conducted theoretical analysis to propose a better mechanism that can address these problems, which is described in the Subsec. 4.2.

4.2. Theoretical analysis

In this paper, we evaluate workers' qualities by computing their DOT, as high-quality workers tend to provide high-quality data. Data quality is closely related to the quality of the truth inference results. Specifically, the higher the requirement for worker DOT, the higher the quality of the obtained data, and the smaller the error in the truth inference results. $\mathbb{G}(\Theta)$ is

defined as the error obtained in truth inference when the worker DOT threshold is set to Θ, which is calculated by

$$\mathbb{G}(\Theta) = \frac{e_0 e_1}{(e_0 - e_1)\Theta^3 + e_1}. \tag{4}$$

Here, e_0 and e_1 are two constants in the interval $(0, 1)$ and satisfy $e_0 \gg e_1$. Specifically, e_0 represents the error obtained in truth inference when $\Theta = 0$, which means there is no requirement for the workers' DOT, and it can be regarded as the error using the data provided by all workers. Meanwhile, e_1 represents the error obtained when $\Theta = 1$, which can be considered as the error obtained by using the data from the UAVs since the data from the UAVs is completely trustworthy and accurate, and thus $e_1 \approx 0$.

In the interval $[0, 1]$, $\frac{d\mathbb{G}(\Theta)}{d\Theta}$ is negative and thus the function $\mathbb{G}(\Theta)$ is monotonically decreasing. It is evident that $\mathbb{G}(\Theta)$ represents the relationship between the worker DOT selection threshold Θ and the error $\mathbb{G}(\Theta)$ in truth inference.

To present more clearly the accuracy and effectiveness of data collection and inference in the MCS system, we define the precision $\mathcal{Q}$ to represent the precision of a single task's estimated truth, which is defined below.

$$\mathcal{Q}(\Theta) = 1 - \mathbb{G}(\Theta) + \mathbb{e}_{\mathcal{Q}}, \tag{5}$$

where $\mathbb{e}_{\mathcal{Q}}$ represents the random perturbation of precision $\mathcal{Q}$ under different circumstances, with $\mathbb{E}(\mathbb{e}_{\mathcal{Q}}) = 0$.

4.2.1 Directly introducing UAVs may not result in higher benefits

Suppose there are m tasks in the MCS system, and the total cost of recruiting workers is $\mathcal{C}_W$. Meanwhile, we assume the cost of a worker to complete a task is $\mathfrak{C}_2$. Since μ workers are assigned to each task, the total costs of assigning workers to complete all m tasks in a round are calculated by

$$\mathcal{C}_W = \mu \mathfrak{C}_2 m. \tag{6}$$

We assume that the number of tasks completed by UAVs is denoted as ϖ. And we define the distance traveled by UAVs as $\mathcal{d}$. The distance can be calculated by

$$\mathcal{d} = \mathfrak{A}\varpi + \mathbb{e}_{\mathcal{d}}, \tag{7}$$

where $\mathfrak{A}$ is a coefficient representing the average distance that UAVs need to travel to complete a task. Similarly, $\mathbb{e}_{\mathcal{d}}$ is a random disturbance that occurs during the UAVs task completion process and satisfies $\mathbb{E}(\mathbb{e}_{\mathcal{d}}) = 0$.

The price of UAVs is usually related to the distance they need to travel. Specifically, the further UAVs fly, the higher the costs, so we set the cost per unit distance of UAVs' movement as $\mathfrak{C}_1$, and the total cost of dispatching UAVs as $\mathcal{C}_{UAV}$. The cost of dispatching UAVs, $\mathcal{C}_{UAV}$, can be calculated as follows.

$$\begin{aligned} \mathcal{C}_{UAV} &= \mathfrak{C}_1 \mathcal{d} + \mathfrak{B} \\ &= \mathfrak{C}_1(\mathfrak{A}\varpi + \mathbb{e}) + \mathfrak{B}, \end{aligned} \tag{8}$$

where $\mathfrak{B}$ is a constant factor that can be explained by various factors, such as fixed costs when UAVs start up.

So, the expected total cost of dispatching UAVs is calculated by

$$\begin{aligned} \mathbb{E}(\mathcal{C}_{UAV}) &= \mathbb{E}\big(\mathfrak{C}_1(\mathfrak{A}\varpi + \mathbb{e})\big) + \mathbb{E}(\mathfrak{B}) \\ &= \mathbb{E}(\mathfrak{C}_1\mathfrak{A}\varpi) + \mathbb{E}(\mathfrak{C}_1\mathbb{e}) + \mathbb{E}(\mathfrak{B}) \\ &= \mathfrak{C}_1\mathfrak{A}\varpi + \mathfrak{B}. \end{aligned} \tag{9}$$

After obtaining the cost of UAVs and workers, we can calculate the total cost by Eq. (10), which is denoted as $\mathcal{C}$.

$$\mathcal{C} = \mathfrak{T}\mathcal{C}_{UAV} + \mathcal{C}_W. \tag{10}$$

When ϖ equals 0, which means that no UAV is dispatched to work, we consider that UAVs do not generate any cost, and $\mathfrak{T}$ equals 0. Otherwise, $\mathfrak{T}$ equals 1. Meanwhile, the expected total cost $\mathbb{E}(\mathcal{C})$ can be calculated by

$$\begin{aligned} \mathbb{E}(\mathcal{C}) &= \mathfrak{T}\mathbb{E}(\mathcal{C}_{UAV}) + \mathbb{E}(\mathcal{C}_W) \\ &= \mathfrak{T}\mathfrak{C}_1\mathfrak{A}\varpi + \mathfrak{T}\mathfrak{B} + \mu\mathfrak{C}_2 m. \end{aligned} \tag{11}$$

The expected accuracy of the data provided by UAVs $\mathbb{E}_{UAV}(\mathcal{Q})$ and the expected accuracy of inferred truth based on all workers' data $\mathbb{E}_W(\mathcal{Q})$ can be obtained as follows.

$$\mathbb{E}_{UAV}(\mathcal{Q}) = \mathbb{E}\big(\mathcal{Q}(1)\big) = 1 - e_1, \tag{12}$$

$$\mathbb{E}_W(\mathcal{Q}) = \mathbb{E}\big(\mathcal{Q}(0)\big) = 1 - e_0. \tag{13}$$

To evaluate the effectiveness of different methods more accurately in obtaining benefits per unit cost, we define the Accuracy-Cost-Ratio $\mathcal{R}$ as the evaluation index, which is calculated according to Eq. (14).

$$\begin{aligned} \mathcal{R} &= \frac{\varpi\mathbb{E}_{UAV}(\mathcal{Q}) + (m - \varpi)\mathbb{E}_W(\mathcal{Q})}{\mathbb{E}(\mathcal{C})} \\ &= \frac{\varpi\mathbb{E}_{UAV}(\mathcal{Q}) + (m - \varpi)\mathbb{E}_W(\mathcal{Q})}{\mathfrak{T}\mathfrak{C}_1\mathfrak{A}\varpi + \mathfrak{T}\mathfrak{B} + \mu\mathfrak{C}_2 m}. \end{aligned} \tag{14}$$

The Accuracy-Cost-Ratio $\mathcal{R}_{|\mathfrak{T}=0}$ without using any UAV is shown as follows.

$$\mathcal{R}_{|\mathfrak{T}=0} = \frac{m\mathbb{E}_W(\mathcal{Q})}{\mu\mathfrak{C}_2 m} = \frac{\mathbb{E}_W(\mathcal{Q})}{\mu\mathfrak{C}_2}. \tag{15}$$

Assumption 1: *When* $\mathfrak{T} = 0$*, it holds that* $\frac{\mathbb{E}_W(\mathcal{Q})}{\mathbb{E}_{UAV}(\mathcal{Q})} > \frac{\mu\mathfrak{C}_2\varpi}{\mathfrak{T}\mathfrak{C}_1\mathfrak{A}\varpi+\mathfrak{T}\mathfrak{B}+\mu\mathfrak{C}_2\varpi}$.

Justification: Here, $\mathfrak{T}\mathfrak{C}_1\mathfrak{A}\varpi + \mathfrak{T}\mathfrak{B}$ represents the cost of using UAVs to complete ϖ tasks, while $\mu\mathfrak{C}_2\varpi$ represents the cost of recruiting workers to complete ϖ tasks. The cost of dispatching UAVs is several times higher than the cost of recruiting workers, and the quality of inferred truth obtained through all worker data is usually not lower than 0.5.

Theorem 1: *Under Assumption 1, introducing UAVs cannot lead to higher Accuracy-Cost-Ratio.*

Proof: We assume that introducing UAVs can lead to higher returns. Therefore, we have:

$$\frac{\varpi\mathbb{E}_{UAV}(\mathcal{Q}) + (m-\varpi)\mathbb{E}_W(\mathcal{Q})}{\mathfrak{T}\mathfrak{C}_1\mathfrak{A}\varpi + \mathfrak{T}\mathfrak{B} + \mu\mathfrak{C}_2 m} > \frac{\mathbb{E}_W(\mathcal{Q})}{\mu\mathfrak{C}_2}. \tag{16}$$

Simplifying the above inequality yields Eq. (17).

$$\frac{\mathbb{E}_W(\mathcal{Q})}{\mathbb{E}_{UAV}(\mathcal{Q})} < \frac{\mu\mathfrak{C}_2\varpi}{\mathfrak{T}\mathfrak{C}_1\mathfrak{A}\varpi + \mathfrak{T}\mathfrak{B} + \mu\mathfrak{C}_2\varpi}. \tag{17}$$

This contradicts Assumption 1, so the original assumption is invalid. Therefore, under Assumption 1, introducing UAVs cannot obtain a higher Accuracy-Cost-Ratio. In other words, in most cases, introducing UAVs cannot achieve better results. Therefore, this theorem holds.

■

4.2.2 Introducing highly trustworthy workers and data recovery methods can lead to higher benefits

In data recovery methods, highly trustworthy workers' data is often used for truth inference. Then data recovery methods are used to obtain the inferred truth of tasks lacking highly trustworthy worker participation. In this paper, if a worker's DOT is greater than the threshold Θ, we consider the worker to be highly trustworthy, and only their data is considered trustworthy. Meanwhile, the accuracy of the inferred truth obtained from highly trustworthy workers is $\mathcal{Q}(\Theta)$. We denoted the coverage rate of highly trustworthy workers as $\mathbb{H}(\Theta)$. As the threshold Θ increases, many tasks will lack highly trustworthy worker participation, i.e., the coverage rate of highly trustworthy workers will decrease. Eq. (15) shows the relationship between the coverage rate of highly trustworthy workers $\mathbb{H}(\Theta)$ and the threshold Θ.

$$\mathbb{H}(\Theta) = (\sigma - 1)\Theta^3 + 1, \tag{18}$$

where σ is a constant that represents the coverage rate of trustworthy workers when $\Theta = 1$. Because few workers' DOT can achieve 1, it is satisfied that $\sigma \approx 0$. $\mathbb{H}(\Theta)$ monotonically decreases in the interval [0, 1], and satisfies $\mathbb{H}(0) = 1,\ \mathbb{H}(1) = \sigma$.

We define the accuracy of data recovery methods' result as $\mathcal{Q}_{Recover}(\Theta)$, which has a strong positive correlation with the quality of the inferred truth obtained by highly trustworthy workers and the coverage rate of highly trustworthy workers. $\mathcal{Q}_{Recover}(\Theta)$ and it can be calculated by

$$\begin{aligned}\mathcal{Q}_{Recover}(\Theta) &= \mathbb{H}(\Theta)\mathbb{E}\big(\mathcal{Q}(\Theta)\big) + \mathbb{e}_{Recover} \\ &= [(\sigma-1)\Theta^3 + 1]\left[1 - \frac{e_0 e_1}{(e_0 - e_1)\Theta^3 + e_1}\right] + \mathbb{e}_{Recover}.\end{aligned} \tag{19}$$

Here, $\mathbb{e}_{Recover}$ is the random error of the data recovery effect, and $\mathbb{E}(\mathbb{e}_{Recover}) = 0$. Therefore, its expectation can be obtained as follows.

$$\mathbb{E}\big(\mathcal{Q}_{Recover}(\Theta)\big) = [(\sigma-1)\Theta^3 + 1]\left[1 - \frac{e_0 e_1}{(e_0 - e_1)\Theta^3 + e_1}\right]. \tag{20}$$

Assumption 2: $\sigma {e_1}^2 = 0, 1-\sigma = 1, e_0 - e_1 = e_0, e_0 - \sigma e_1 = e_0, 1 - 2\sqrt{e_1} > e_0, e_0 - \sqrt{e_1} > 0$.

Justification: Since $e_1 \approx 0,\ \sigma \approx 0$, it can be reasonably assumed that $\sigma {e_1}^2 = 0,\ 1-\sigma = 1, e_0 - \sigma e_1 = e_0$. Also, since $e_1 \ll e_0 < 1$, it can be rationally assumed that $e_0 - e_1 = e_0, 1 - 2\sqrt{e_1} > e_0, e_0 - \sqrt{e_1} > 0$.

Theorem 2: *There exists* $\Theta \in (0,1)$ *such that* $\mathbb{E}\big(\mathcal{Q}_{Recover}(\Theta)\big) > \mathbb{E}\big(\mathcal{Q}(0)\big)$.

Proof: When $\Theta = \sqrt[3]{\frac{\sqrt{\frac{e_0 e_1 (e_0 - \sigma e_1)}{1-\sigma}} - e_1}{e_0 - e_1}}$, $\mathbb{E}\big(\mathcal{Q}_{Recover}(\Theta)\big)$ takes its maximum value, which is proved in Lemma1. At this time, the maximum value $\max\Big(\mathbb{E}\big(\mathcal{Q}_{Recover}(\Theta)\big)\Big)$ can be calculated as below.

$$\max\Big(\mathbb{E}\big(\mathcal{Q}_{Recover}(\Theta)\big)\Big) = 1 + \frac{\sigma-1}{e_0 - e_1}\left(\sqrt{\frac{{e_0}^2 e_1 - \sigma e_0 {e_1}^2}{1-\sigma}} - e_1 - e_0 e_1\right) - e_0 e_1 \left(\frac{e_0 - \sigma e_1}{e_0 - e_1}\right)^2 \sqrt{\frac{1-\sigma}{{e_0}^2 e_1 - \sigma e_0 {e_1}^2}}. \tag{21}$$

According to Assumption 2, we know that $\sigma {e_1}^2 = 0, 1-\sigma = 1, e_0 - e_1 = e_0, e_0 - \sigma e_1 = e_0, 1 - 2\sqrt{e_1} > e_0$, so

$$\max\left(\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right)\right) = 1 + \frac{-1}{e_0}\left(\sqrt{\frac{{e_0}^2 e_1 - 0}{1}} - e_1 - e_0 e_1\right) - e_0 e_1 \left(\frac{e_0}{e_0}\right)^2 \sqrt{\frac{1}{{e_0}^2 e_1}}$$
$$= 1 - 2\sqrt{e_1} + e_1 + \frac{e_1}{e_0}$$
$$> 1 - 2\sqrt{e_1}$$
$$> e_0. \tag{22}$$

Therefore, according to Eq. (22), introducing data recovery methods can improve revenue and obtain more accurate inferred truth without additional costs.

■

Lemma 1: *$\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right)$ attains its maximum value when $\Theta = \sqrt[3]{\frac{\sqrt{\frac{e_0 e_1 (e_0 - \sigma e_1)}{1-\sigma}} - e_1}{e_0 - e_1}}$.*

Proof: For convenience, we let $\Theta' = \Theta^3$, and use Θ' to represent $\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right)$, as shown below:

$$\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right) = [(\sigma - 1)\Theta' + 1]\left[1 - \frac{e_0 e_1}{(e_0 - e_1)\Theta' + e_1}\right]. \tag{23}$$

By simple calculation, we can obtain

$$\frac{d\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right)}{d\Theta'} = (\sigma - 1) + \frac{e_0 e_1 (e_0 - \sigma e_1)}{[(e_0 - e_1)\Theta' + e_1]^2}, \tag{24}$$

$$\frac{d^2\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right)}{d{\Theta'}^2} = -2\frac{e_0 e_1 (e_0 - \sigma e_1)}{[(e_0 - e_1)\Theta' + e_1]^3} < 0. \tag{25}$$

Therefore, $\frac{d\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right)}{d\Theta'}$ is monotonically decreasing, and thus has a unique zero point at $\frac{d\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right)}{d\Theta'}$, which is proved to be in the interval $(0, 1)$ in Lemma 2. Thus, $\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right)$ attains its maximum value when $\Theta' = \frac{\sqrt{\frac{e_0 e_1 (e_0 - \sigma e_1)}{1-\sigma}} - e_1}{e_0 - e_1}$, i.e., when $\Theta = \sqrt[3]{\frac{\sqrt{\frac{e_0 e_1 (e_0 - \sigma e_1)}{1-\sigma}} - e_1}{e_0 - e_1}}$.

■

Lemma 2 : $\sqrt[3]{\frac{\sqrt{\frac{e_0 e_1 (e_0 - \sigma e_1)}{1-\sigma}} - e_1}{e_0 - e_1}} \in [0, 1]$.

Proof: From Assumption 2, we know that $\sigma {e_1}^2 = 0, 1 - \sigma = 1, e_0 - e_1 = e_0, e_0 - \sigma e_1 = e_0, 1 - 2\sqrt{e_1} > e_0$, so

$$\frac{\sqrt{\frac{e_0 e_1 (e_0 - \sigma e_1)}{1-\sigma}} - e_1}{e_0 - e_1} = \frac{\sqrt{{e_0}^2 e_1} - e_1}{e_0}$$
$$= \frac{e_0\sqrt{e_1} - e_1}{e_0}. \tag{26}$$

Here, $\frac{e_0\sqrt{e_1} - e_1}{e_0} = \frac{\sqrt{e_1}(e_0 - \sqrt{e_1})}{e_0}$, and according to Assumption 2, we can assume that $e_0 - \sqrt{e_1} > 0$, which means that $\frac{\sqrt{\frac{e_0 e_1 (e_0 - \sigma e_1)}{1-\sigma}} - e_1}{e_0 - e_1} > 0$.

Meanwhile, $\frac{e_0\sqrt{e_1} - e_1}{e_0} < \sqrt{e_1} < 1$.

Therefore, $0 < \frac{\sqrt{\frac{e_0 e_1 (e_0 - \sigma e_1)}{1-\sigma}} - e_1}{e_0 - e_1} < 1$, which implies that $\sqrt[3]{\frac{\sqrt{\frac{e_0 e_1 (e_0 - \sigma e_1)}{1-\sigma}} - e_1}{e_0 - e_1}} \in (0, 1)$. Therefore, Lemma 2 holds.

■

4.2.3 Combining different benchmarks for truth inference is promising in a cost-controllable manner.

According to Theorem 1, in most cases, introducing UAVs cannot achieve a higher Accuracy-Cost-Ratio. However, in many cases, it is unavoidable for the MCS system to improve the quality of truth inference. In this case, dispatching UAVs is a promising method under the premise of using data recovery methods. Because the high-quality data obtained by UAVs can simultaneously improve the quality of data recovery methods, dispatching UAVs can obtain higher benefits. Specifically,

under the premise of using data recovery methods, dispatching ϖ UAVs can improve the accuracy of data recovery methods by $\frac{\varpi}{m}(1-e_1)$, which has been proven in Theorem 3.

Theorem 3: *In the premise of using data recovery methods, introducing ϖ UAVs can improve the quality obtained by data recovery methods by $\frac{\varpi}{m}(1-e_1)$.*

Proof: In the solution that simultaneously utilizes UAVs, highly trustworthy workers, and data recovery methods, we assume that the number of tasks completed by UAVs is ϖ, and the coverage range of highly trustworthy workers is $\mathbb{H}(\Theta)$. Therefore, there are still $\left[\left(1-\mathbb{H}(\Theta)\right)m-\varpi\right]$ tasks that need to be completed through data recovery methods.

At this point, due to the introduction of UAVs, the coverage rate of highly trustworthy workers and UAVs is denoted as $\mathbb{H}'(\Theta)$, which can be calculated by

$$\mathbb{H}'(\Theta)=(\sigma-1)\Theta^3+\frac{\varpi}{m}+1. \tag{27}$$

And the expected average accuracy of both UAVs and highly trustworthy workers can be obtained by

$$\mathbb{E}'\left(\mathcal{Q}(\Theta)\right)=\frac{\varpi\mathbb{E}_{UAV}(\mathcal{Q})+\mathbb{H}(\Theta)m\mathbb{E}\left(\mathcal{Q}(\Theta)\right)}{\varpi+\mathbb{H}(\Theta)m}. \tag{28}$$

From this, we can obtain the accuracy of data recovery methods, which is denoted as $\mathcal{Q}'_{Recover}(\Theta)$, based on Eq (29).

$$\begin{aligned}\mathcal{Q}'_{Recover}(\Theta)&=\mathbb{H}'(\Theta)\mathbb{E}'\left(\mathcal{Q}(\Theta)\right)+\mathbb{e}_{Recover}\\&=\left[(\sigma-1)\Theta^3+\frac{\varpi}{m}+1\right]\left[\frac{\varpi\mathbb{E}_{UAV}(\mathcal{Q})+\mathbb{H}(\Theta)m\mathbb{E}\left(\mathcal{Q}(\Theta)\right)}{\varpi+\mathbb{H}(\Theta)m}\right]+\mathbb{e}_{Recover}.\end{aligned} \tag{29}$$

Therefore, the expected accuracy of data recovery methods be calculated as follows.

$$\mathbb{E}\left(\mathcal{Q}'_{Recover}(\Theta)\right)=\left[(\sigma-1)\Theta^3+\frac{\varpi}{m}+1\right]\left[\frac{\varpi\mathbb{E}_{UAV}(\mathcal{Q})+\mathbb{H}(\Theta)m\mathbb{E}\left(\mathcal{Q}(\Theta)\right)}{\varpi+\mathbb{H}(\Theta)m}\right]. \tag{30}$$

By subtracting $\mathbb{E}\left(\mathcal{Q}'_{Recover}(\Theta)\right)$ and $\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right)$, the following results can be obtained by

$$\mathbb{E}\left(\mathcal{Q}'_{Recover}(\Theta)\right)-\mathbb{E}\left(\mathcal{Q}_{Recover}(\Theta)\right)=\frac{\varpi}{m}(1-e_1). \tag{31}$$

Thus, it can be concluded that introducing ϖ UAVs can improve the accuracy of data recovery methods by $\frac{\varpi}{m}(1-e_1)$, which proves Theorem 3.

■

Furthermore, in the initial stage of the system startup, the platform often does not have sufficient understanding of the workers' DOT, and $\mathbb{H}(\Theta)$ cannot accurately represent the coverage rate of highly trustworthy workers, which is at a very low level. This leads to poor performance of the truth inference and further affects the performance of the system's worker recognition, leading to the coverage rate of highly trustworthy workers remaining at a low level for a long time. Currently, introducing UAVs as a supplement to the coverage rate of highly trustworthy workers greatly improves the performance of data recovery methods, avoiding the occurrence of the cold start phenomenon.

Finally, introducing UAVs at this point is cost-controllable. UAVs play the role of further improving data quality and speeding up system startup, rather than simply using them to obtain high-quality data. Therefore, the number of dispatched UAVs is controllable, and the platform can dynamically plan the number of UAVs based on the requirements for system startup speed and truth inference quality, rather than blindly dispatching UAVs. In this way, the cost of dispatching UAVs is controlled by DPA according to their requirements, so introducing UAVs at this point is cost-controllable.

4.3. Proposed muti-level GTD system

In this paper, we design a multi-level GTD system to achieve a complete posteriori GTD matrix so that all the reported data has a comparable GTD to evaluate the sensing quality. Specifically, our GTD is divided into 3 main levels, which are gold, silver, and bronze GTD, as shown in Fig. 4.

The gold GTD is the data collected by the UAVs assigned to complete tasks simultaneously with workers. In general, the UAVs are dispatched to finish different tasks in different rounds, and the number of gold GTD is much smaller than the number of tasks, as defined in Def. 8. Therefore, the gold GTD is defined as Eq. (32). If task w_j has no UAVs dispatched in the t-th round, then its gold GTD is marked as 0 directly, indicating the missing.

$$v_{j,t}^{Gold}=\begin{cases}v_{j,t}^{UAV},\varphi_{j,t}^{Gold}=1,\\0,\varphi_{j,t}^{Gold}=0.\end{cases} \tag{32}$$

The silver GTD is the weighted average of the reported data of each highly trustworthy worker, where the weight is their composite DOT, shown as Eq. (33). Obviously, there is a silver GTD in this round of the task if there is at least one highly trustworthy worker.

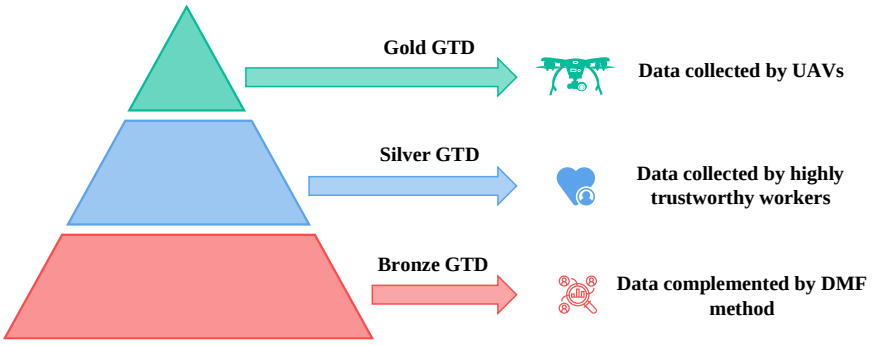


Fig. 4. The three-level GTD in DLFTI.

$$v_{j,t}^{Silver} = \begin{cases} \frac{\sum_{s_i \in S_k^{HT}} \hat{q}_{i,k} v_{j,t}^{i}}{\sum_{s_i \in S_k^{HT}} \hat{q}_{i,k}}, \varphi_{j,t}^{Silver} = 1, \\ 0, \varphi_{j,t}^{Silver} = 0. \end{cases} \tag{33}$$

Combining $v_{j,t}^{Gold}$ and $v_{j,t}^{Silver}$, an incomplete gold and silver GTD matrix $v_{t,j}^{*}$ can be obtained as follows.

$$v_{j,t}^{*} = \begin{cases} v_{j,t}^{Gold}, \varphi_{j,t}^{Gold} = 1, \\ v_{j,t}^{Silver}, \varphi_{j,t}^{Gold} = 0, \varphi_{j,t}^{Silver} = 1, \\ 0, else. \end{cases} \tag{34}$$

Because the number of UAVs and highly trustworthy workers is small, it is generally not possible to obtain gold or silver GTD for all tasks in a cycle. Therefore, we use the obtained gold GTD and silver GTD to compose the incomplete GTD matrix and use the matrix completion method to obtain the complete GTD matrix as the judging benchmark for all workers. Traditional matrix completion methods assume that the data follow linear relationships, and that the correlation between matrix rows is strong and can be projected onto a lower-dimensional linear subspace. However, traditional methods are linear models that cannot reflect non-linear relationships or recover unobserved data from observed data with non-linear relationships. The DMF method uses a multi-layer neural network to learn a non-linear matrix factorization model, which can better handle data with complex non-linear structures. Moreover, DMF has been shown to be effective in capturing complex dependencies and patterns in high-dimensional data. Therefore, we employ the DMF method to complete the incomplete GTD matrix. Therefore, we use DMF to make the GTD matrix completion, for DMF can learn the relationship between the data of different rounds and subregions to complete the sparse matrix every cycle. Then we can get an accurate and complete benchmark matrix as the bronze GTD, which is defined as follows:

Definition 9 (Bronze GTD). Based on the mentioned DMF method, the data of each task without gold GTD and silver GTD is recovered as the bronze GTD. And the bronze GTD of task w_j in the t-th round is denoted as $v_{j,t}^{Bronze}$.

Another feature of this paper is the employment of the DMF method to recover the data matrix every cycle. Compared to traditional methods, DMF can better exploit the nonlinear characteristics via deep neural networks [42]. Therefore, using DMF, matrix completion with better results can be performed, thus further improving the performance of the MCS system.

Since the above incomplete GTD matrix $v_{j,t}^{*}$ cannot evaluate all the data reported by workers, the GTD matrix needs to be recovered through DMF at the end of each cycle. Specifically, for each cycle, we will construct a GTD matrix Y_k of $\tau \times m$. The values in this matrix represent the multi-level GTD of m tasks for τ rounds in the k-th cycle.

Existing data inference methods typically assume linear spatiotemporal correlation between data and complement the matrix with the inherent low-rank property of the sparse matrix [42]. However, the DMF technique is adopted due to the complex spatiotemporal correlations and nonlinear structures of real-world sensed data, i.e., using a nonlinear function $f(\cdot)$ to indicate this kind of relationship, where $f(\cdot)$ is performed on each column of Z individually, as shown in Eq. (35).

$$y_k = f(z), Y_k = f(Z). \tag{35}$$

There should be three components of the optimization function we want to minimize. On the one hand, $f(Z)$ does not have to be exactly equal to Y in practical applications. What we need to do is minimize the difference between the two variables. As a result, we should try to minimize $\|Y_k - f(Z)\|_F^2$. On the other hand, two regularization penalties, $\rho\pi(f)$ and $\frac{\lambda}{2n}\|Z\|_F^2$, are introduced to weaken some insignificant characteristic parameters, considering the impact of the matrix Z and the nonlinear function $f(\cdot)$ on the outcomes of the optimization problem. Thus, the optimization problem can be formulated as follows:

Minimize:

$$\frac{1}{2n}\|Y_k - f(Z)\|_F^2 + \rho\pi(f) + \frac{\lambda}{2n}\|Z\|_F^2. \tag{36}$$

Subject to:

$$f(Z)[i,j] = Y_k[i,j], [i,j] \in \Omega. \tag{37}$$

Algorithm 1: Muti-level GTD Framework.

Input: $\tau, \varphi_{j,t}^{UAV}, v_{j,t}^{UAV}, R_j^t, S_k^{HT}$,
$g^* \triangleq \{g^{(1)}(\cdot), g^{(2)}(\cdot), \dots, g^{(d+1)}(\cdot)\}$.

Output: $\hat{v}_{j,t}^*$.

for each cycle k:
  for each round t in cycle k:
    for each task w_j:
      $\varphi_{j,t}^{Gold} = \varphi_{j,t}^{UAV}$;
      Calculate $v_{j,t}^{Gold}$ by Eq. (32);
      if exists s_i in $R_j^t \cap S_k^{HT}$:
        $\varphi_{j,t}^{Silver} = 1$;
      else:
        $\varphi_{j,t}^{Silver} = 0$;
      end if
      Calculate $v_{j,t}^{Silver}$ by Eq. (33);
      Obtain $v_{j,t}^*$ by Eq. (34);
    end for
  end for
  Construct GTD matrix Y_k by $v_{j,t}^*$;
  Random Init Z, W^*, b^*;
  Build the Neural Networks by g^*, b^*, W^* and Z;
  Init $count = 0$;
  Init MAX_ITER for the DMF iteration;
  while $count < MAX\_ITER$ **do:**
    Calculate $L(Z, W^*, b^*)$ by Eq. (36);
    Fix Z, W^*, and $b^*$to reduce $L(Z, W^*, b^*)$
    $count = count + 1$
    if $convergent$:
      break;
    end if
  end while
  Obtain $\hat{Y}_k = f(Z)$;
  Calculate $v_{j,t}^{Bronze}$ by Eq. (44);
  Complete $\hat{v}_{j,t}^*$ by Eq. (45);
end for

In Eq. (36), $\pi(\cdot)$ denotes a penalization or constraint on $f(\cdot)$, while $\frac{\lambda}{2n}\|Z\|_F^2$ is a regularization penalty term to limit the matrix Z, and Ω denotes the positions of observed entries of Y_k. The nonlinear function $f(\cdot)$ is a neural network containing d hidden layers, then we have

$$W^* \triangleq \{W^{(1)}, W^{(2)}, \dots, W^{(d)}, W^{(d+1)}\}. \tag{38}$$

$$b^* \triangleq \{b^{(1)}, b^{(2)}, \dots, b^{(d)}, b^{(d+1)}\}. \tag{39}$$

The corresponding activation function set is

$$g^* \triangleq \{g^{(1)}(\cdot), g^{(2)}(\cdot), \dots, g^{(d)}(\cdot), g^{(d+1)}(\cdot)\}. \tag{40}$$

The $(d+1)th$ term in sets W^*, b^*, g^*, represents the activation function or parameter from hidden layer d to the output layer. And the nonlinear function $f(\cdot)$ can be denoted as

$$f(Z) = g^{(K+1)}\left(W^{(k+1)}g^{(k)}\left(W^{(k)}, \dots, g^{(1)}\left(W^{(1)}z + b^{(1)}\right)\dots + b^{(k)}\right) + b^{(k+1)}\right). \tag{41}$$

We can denote the loss function of Eq. (36) as $L(Z, W^*, b^*)$. $\pi(f)$ is related to nonlinear function $g(\cdot)$, matrix $W^{(j)}$, and bias vector $b^{(j)}$ $(j = 1, 2, 3 \ldots K, K+1)$ where the matrix $W^{(j)}$ has the greatest influence. Consequently, the function $\pi(f)$ can be expressed as

$$\rho\pi(f) = \frac{1}{2}\sum_{j=1}^{k+1}\left\|W^{(j)}\right\|_F^2. \tag{42}$$

The DMF's output vector is a complete vector whose accuracy is cared only in the parts without GTD. As a result, the output vector is viewed as a sparse vector when the neural network is trained. The expression of the nonlinear mapping $f(\cdot)$ can be derived from the parameters W^* and b^* obtained after training the neural network, and then Eq. (43) is introduced to complete the inferred data $\hat{Y}_k$.

$$\hat{Y}_k = f(Z). \tag{43}$$

Where Z is also obtained from the DMF training. It is worth noting that during training and testing, Z is regarded as one of the neural network's parameters. When training the neural network, Z will have a randomly chosen initial value, and after many iterations, Z will eventually converge to a constant.

The DMF neural network differs significantly from conventional neural networks. In order for traditional neural networks to function, a mapping between the output and input vectors must be found. However, the DMF neural network has the input vector as one of the parameters of the neural network, and there is no need to define the input vector. Instead, we only need to consider whether the output vector converges to our sparse vector, because the outcome after the neural network converges is what we need.

From the above analysis, it is clear that the $\hat{Y}_k$ obtained through DMF is the multi-level GTD for all tasks in the k-th cycle, from which $v_{j,t}^{Bronze}$ is calculated by Eq. (44). For the sake of narration, we can let $\hat{v}_{j,t}^*$ denote the multi-level GTD for the task w_j in the t-th round, which contains the gold, silver, and bronze GTD, as defined in Eq. (45).

$$v_{j,t}^{Bronze} = \hat{Y}_{[t/\tau]}[t \bmod \tau, j]. \tag{44}$$

$$\hat{v}_{j,t}^* = \begin{cases} v_{j,t}^{Gold}, \varphi_{j,t}^{Gold} = 1, \\ v_{j,t}^{silver}, \varphi_{j,t}^{Gold} = 0, \varphi_{j,t}^{Silver} = 1, \\ v_{j,t}^{Bronze}, else. \end{cases} \tag{45}$$

The pseudocode of constructing the multi-level GTD is described in detail in Algorithm 1. Based on multi-level GTD, we can obtain the workers' DOT for each cycle, and then update their composite DOT by their historical and current DOT. Dynamic and accurate worker recognition based on a small number of UAVs is achieved by this method, which is explained in detail in the next section.

4.4. Proposed worker recognition system

In our DLFTI mechanism, we aggregate all the data information in each cycle after the tasks are completed, and form the multi-level GTD $\hat{v}_{j,t}^*$. After that, the sensing quality of all the data reported by the workers is calculated by $\hat{v}_{j,t}^*$. We then compute the workers' DOT at the end of the cycle. Finally, composite DOT is updated by their historical and current DOT. The sensing quality is calculated by the distance between the data submitted by workers and the multi-level GTD, as shown in Eq. (46). When the distance between them is large, it means that the data reported by the worker is likely to be wrong, while the opposite means that the data reported by the worker is comparatively trustworthy.

$$q_{i,j}^t = \frac{1}{1 + \alpha_{j,t}\left(\hat{v}_{j,t}^* - v_{j,t}^i\right)^2}. \tag{46}$$

Where $\alpha_{j,t}$ is a hyperparameter that defines the strength of the constraint on sensing quality. The larger the value of $\alpha_{j,t}$, the more stringent the sensing quality requirement is, and the small error between the reported data and the multi-level GTD may cause the sensing quality to drop rapidly. According to the three different levels of the GTD, $\alpha_{j,t}$ will take three different values, as shown in Eq. (47). Since we assume that the accuracy of gold GTD is the largest, silver GTD is the second and bronze GTD is the smallest, $\alpha_1 > \alpha_2 > \alpha_3$ should be satisfied.

$$\alpha_{j,t} = \begin{cases} \alpha_1, \varphi_{j,t}^{Gold} = 1, \\ \alpha_2, \varphi_{j,t}^{Gold} = 0, \varphi_{j,t}^{Silver} = 1, \\ \alpha_3, else. \end{cases} \tag{47}$$

After getting $q_{i,j}^t$ for a whole cycle, the worker's DOT can be obtained. In this paper, we choose to calculate the DOT $q_{i,k}$ for worker s_i by weighting the average of all the sensing qualities in the k-th cycle, defined as below.

$$q_{i,k} = \frac{\sum_{t=(k-1)\tau+1}^{k\tau} \varphi_{j,t}^i \omega_{j,t} q_{i,j}^t}{\sum_{t=(k-1)\tau+1}^{k\tau} \varphi_{j,t}^i \omega_{j,t}}. \tag{48}$$

Algorithm 2: Fast worker recognition by trust computing.

Input: $\tau, \theta_1, \theta_2, \alpha_{j,t}, \omega_{j,t}, \beta$.
Output: $\hat{q}_{i,k}, S_k^{HT}, S_k^{DT}$.

for each cycle k:
  for each round t in cycle k:
    for each task w_j:
      for each recruited s_i in R_j^t:
        Evaluate all the $q_{i,j}^t$ by Eq. (46);
      end for
    end for
  end for
  for each worker s_i:
    Calculate $q_{i,k}$ by Eq. (48);
    Calculate $\hat{q}_{i,k}$ by Eq. (50);
  end for
  Update S_k^{HT} and S_k^{DT};
end for

The $\omega_{j,t}$ is also a hyperparameter that defines the weight of the sensing quality for task w_j in the t-th round. Similar to $\alpha_{j,t}$, the $\omega_{j,t}$ takes three different values, as shown in Eq. (49). Since we assume that the gold GTD is the most credible, and the silver and bronze ones are less credible in order. So, $\omega_1 > \omega_2 > \omega_3$ should be satisfied.

$$\omega_{j,t} = \begin{cases} \omega_1, \varphi_{j,t}^{Gold} = 1, \\ \omega_2, \varphi_{j,t}^{Gold} = 0, \varphi_{j,t}^{Silver} = 1, \\ \omega_3, else. \end{cases} \tag{49}$$

Finally, the composite DOT $\hat{q}_{i,k}$is updated by their historical composite DOT $\hat{q}_{i,k-1}$ and current DOT $q_{i,k}$, as Eq (50).

$$\hat{q}_{i,k} = \beta q_{i,k} + (1-\beta)\hat{q}_{i,k-1}. \tag{50}$$

Here, β is a hyperparameter used to control the update rate of the composite DOT. When β is larger, it indicates that this update strategy is more aggressive and the current DOT has a greater impact on the overall composite one; vice versa, it indicates that the update strategy is more conservative and the historical composite DOT has a greater impact.

The pseudocode of worker recognition by trust computing is described in detail in Algorithm 2. In this way, $q_{i,j}^t$, $q_{i,k}$ and $\hat{q}_{i,k}$ are updated by the above method after each round of workers reporting data. And as the number of cycles gradually increases, each worker's DOT will converge progressively to close to its real level.

4.5 Proposed muti-level ETD system

After obtaining the gold, silver and bronze GTD, we can evaluate the data reported by the workers to obtain the DOT of each worker. After that, we expect to fully use each worker's DOT and the reported data to gain the estimated truth as accurately as possible for downstream applications.

Although we can initially identify the trustworthy and untrustworthy workers through multi-level GTD and worker recognition. It is still inevitable for the platform to recruit some untrustworthy workers who submit fake data, especially in the early stages of system startup. Therefore, we need to further conduct truth discovery in the same cycle for downstream applications from the conflicting data reported by workers.

It is transparent that due to the rarity of highly trustworthy workers, especially at the beginning of the system startup, the number of silver GTD is very scarce. In this case, most of the data in the multi-level GTD is supplemented by the DMF, which did not fully utilize the data reported by each trustworthy worker, thus leading to a lack of accuracy and huge data waste. Therefore, in order to improve the accuracy of truth discovery, we should increase the proportion of the data from the crowd.

Based on the above considerations, we want to make full use of the data provided by trustworthy workers in the process of truth discovery to achieve higher accuracy. Specifically, we define double trustworthy workers as follows.

Definition 10 (Double Trustworthy Worker). When the current DOT $q_{i,k}$ and the composite DOT $\hat{q}_{i,t}$ of a worker s_i are both greater than or equal to the threshold θ_2, i.e., $min(q_{i,k}, \hat{q}_{i,k}) \geq \theta_2$. Worker s_i is called the double trustworthy worker in the k-th cycle. Obviously, double trustworthy workers are a special case of trustworthy workers when $q_{i,k} \geq \theta_2$ is satisfied as well.

The additional restriction $q_{i,k} \geq \theta_2$ is a requirement that not only does he perform well in the historical tasks, but the data reported by him at the current cycle k is also considered trustworthy. With this double guarantee, it is reasonable to assume that the data submitted by double trustworthy workers in the current cycle is highly trustworthy. We maintain a set S_t^{DT} to

contain all the double trustworthy workers in the k-th cycle. Therefore, we further propose a multi-level ETD based on the multi-level GTD, which is defined as follows.

Definition 11 (Gold ETD, Silver ETD, Bronze ETD). As with the multi-level GTD mechanism we proposed previously, we also construct multi-level ETD, denoted by $u_{j,t}^{Gold}$, $u_{j,t}^{Silver}$ and $u_{j,t}^{Bronze}$. And we let $\tilde{\varphi}_{j,t}^{Gold}$ and $\tilde{\varphi}_{j,t}^{Silver}$ indicate whether there is a gold or silver ETD for task w_j in the t-th round.

Specifically, we have established a three-level ETD system. The gold ETD is the data collected by the UAVs assigned to complete tasks. In fact, the definitions of Gold ETD and Gold GTD are the same, as Eqs. (51)-(52).

Algorithm 3: Accurate truth discovery based on muti-level ETD.

Input: $\tau, R_j^t, S_k^{DT}, \varphi_{j,t}^{Gold}, v_{j,t}^{Gold}, v_{j,t}^{Bronze}$.
Output: $u_{j,t}^{E}$.

```
for each cycle k:
  for each round t in cycle k:
    φ̃_{j,t}^{Gold} = φ_{j,t}^{Gold};
    u_{j,t}^{Gold} = v_{j,t}^{Gold};
    u_{j,t}^{Bronze} = v_{j,t}^{Bronze};
    if exists s_i in R_j^t ∩ S_k^{DT}:
      φ̃_{j,t}^{Silver} = 1
    else:
      φ̃_{j,t}^{Silver} = 0
    end if
    Calculate u_{j,t}^{Silver} by Eq. (53)
    Obtain u_{j,t}^{E} by Eq. (55)
  end for
end for
```

$$u_{j,t}^{Gold} = v_{j,t}^{Gold}. \tag{51}$$

$$\tilde{\varphi}_{j,t}^{Gold} = \varphi_{j,t}^{Gold}. \tag{52}$$

If there is any double trustworthy worker for one task, then there is a silver ETD in the current round. The silver ETD is the weighted average of each double trustworthy worker's reported data, where the weight is their composite DOT, as below.

$$u_{j,t}^{Silver} = \begin{cases} \dfrac{\sum_{s_i \in S_k^{DT}} \hat{q}_{i,k} v_{j,t}^{i}}{\sum_{s_i \in S_k^{DT}} \hat{q}_{i,k}}, \tilde{\varphi}_{j,t}^{Silver} = 1, \\ 0, \tilde{\varphi}_{j,t}^{Silver} = 0. \end{cases} \tag{53}$$

For tasks with neither gold ETD nor silver ETD, we still use the data recovered by DMF as bronze ETD, as Eq. (54).

$$u_{j,t}^{Bronze} = v_{j,t}^{Bronze}. \tag{54}$$

For the sake of narration, we can let $u_{j,t}^{E}$ denote the full ETD for the task w_j in the t-th round, which contains the gold, silver, and bronze ETD, as defined in Eq. (55).

$$u_{j,t}^{E} = \begin{cases} u_{j,t}^{Gold}, \tilde{\varphi}_{j,t}^{Gold} = 1, \\ u_{j,t}^{Silver}, \tilde{\varphi}_{j,t}^{Gold} = 0, \tilde{\varphi}_{j,t}^{Silver} = 1, \\ u_{j,t}^{Bronze}, else. \end{cases} \tag{55}$$

The pseudocode of truth discovery based on multi-level ETD is described in detail in Algorithm 3. It is easy to find that the essential difference between multi-level GTD and multi-level ETD is the silver ones. The main reason for such a design is to consider the different application scenarios of multi-level GTD and multi-level ETD. For multi-level GTD, we prefer to obtain high-precision evaluation metrics to set DOT for workers. But for multi-level ETD, we prefer to obtain more data within the error range of task accuracy, so we should try to avoid the appearance of bronze ETD that is not accurate enough.

Through the above methods, considering the data of UAVs, workers, and DMF method, we can complete truth discovery accurately and quickly, thus further improving the performance of MCS system.

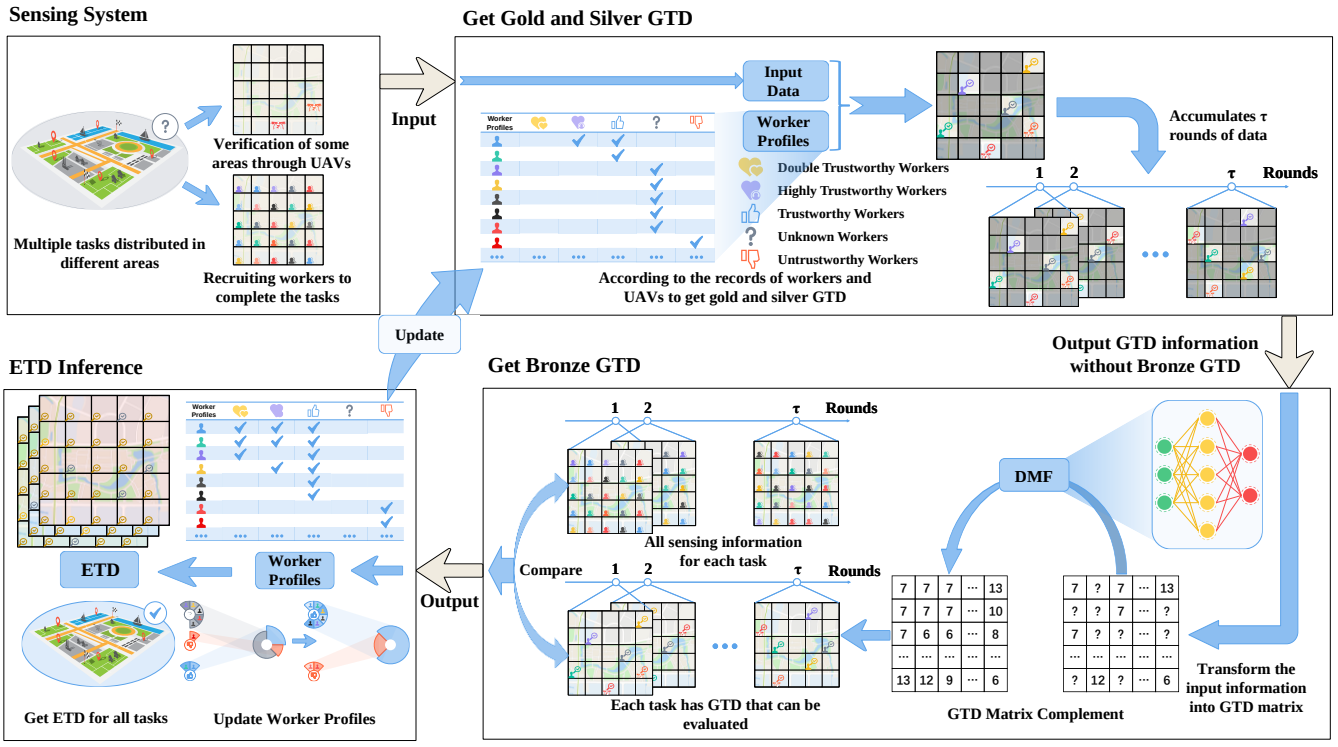


Fig. 5. The framework of DLFTI.

4.6. Summery of DLFTI

The framework of DLFTI is shown in Fig. 5. There are many different tasks distributed in different regions in the MCS system. First, the DPC recruits workers and dispatches UAVs to collect several completely trustworthy data. Secondly, the data provided by UAVs and highly trustworthy workers recorded in the worker profiles are treated as gold GTD and silver GTD, respectively, and an incomplete GTD matrix is obtained based on these gold and silver GTD. Thirdly, after accumulating data for τ rounds, the sparse GTD matrix is complemented by using DMF method to obtain the bronze GTD. At this point, the GTD matrix are completed, and all the data reported by the workers has a relatively accurate judgment benchmark. Finally, the platform updates the worker profiles and get the ETD for all tasks based on the complete GTD data. Then, the data reported by the workers are compared with the obtained GTD to determine whether a worker has completed the task honestly, and thus update the worker profiles. Finally, ETD for all tasks is obtained based on the data from UAVs, double trustworthy workers recorded in the worker profiles, and the DMF method.

5. Performance Analysis

5.1. Simulation setup

In this section, we evaluate the effectiveness of our DLFTI mechanism based on a real-world dataset. The task data comes from a real-world air quality reanalysis dataset [43] containing gridded ground-degree air quality data, at a high spatial resolution in China from 2013 to 2018. By simple comparison, the different types of air quality data have similar distribution trends. Therefore, we chose only the most representative part, i.e., PM2.5.

In the simulations, we take the sensing of location-aware ground PM2.5 as the tasks. In order to enable our experiments to simulate a real MCS environment, we assume that the data submitted by the workers conforms to a normal distribution, where the standard deviation is randomly distributed between [0,10] to simulate differences in sensing competence. The mean value of the normal distribution varies with the types of workers. Specifically, the mean value for trustworthy workers is the real GTD, while it is a task-dependent random value for untrustworthy workers. To simulate the fraudulent gang attacks, the distributions matched by the untrustworthy workers' data for the same task in the same round have the same mean value. By setting up workers in this way, we can fully depict various kinds of workers in the real world.

To control the variables, the thresholds θ_1 and θ_2 for worker recognition are set to 0.8 and 0.6, respectively. The hyperparameters α_1, α_2 and α_3 for calculating $q_{i,j}^t$ are fixed to 5×10^{-3} , 1×10^{-3} and 5×10^{-4}, while ω_1, ω_2 and ω_3 for calculating $q_{i,k}$ are fixed to 100, 50 and 1. Since the DMF algorithm is only a part of the DLFTI mechanism, the precise tuning of its parameters is not in our focus. Therefore, we directly fix the rank of Z as 15, and fix DMF's network structure as a simple two-layer nonlinear structure $g^* = \{tanh, linear\}$. The subsequent experimental results show that even with such a simple network structure, the DMF algorithm still shows its excellent performance.

In addition, to allow for easier exploration of the cost laws of different algorithms, we consider the costs per worker to complete a task as c_{Worker}, while the cost per UAV to complete a task as c_{UAV}. Due to the cost of additional energy and

management incurred by the UAVs, the c_{UAV} is usually a bit higher than c_{Worker}. For the sake of simplicity, we assume $c_{UAV} = \chi c_{Worker}$ is approximately satisfied. And we standardize $c_{Worker} = 1$ and set $\chi = 5$ as a reasonable example, which means that a UAV cost 5 times more to dispatch than a worker to recruit.

More specifically, we generate a total of 500 workers in the simulations, where the ratio of trustworthy workers to the total number of workers is called Trustworthy Rate (TR). In order to fully test the ability of different algorithms in each TR case, we simulated 9 sets of experiments with TR at 0.1, 0.2...0.9 respectively and the simulation parameters are listed in Table 3.

5.2. Algorithms for comparison

Since our DLFTI mechanism considers both worker recognition and truth estimation in MCS, no existing algorithm can be directly applied to our problems. We selected truth discovery methods for all numerical types listed in the latest review paper [25], which are Mean, Median, LFC, to compare DLFTI with the current state-of-the-art numerical truth discovery algorithms. And because these truth discovery algorithms cannot directly evaluate workers' DOT, we regard the ETD obtained

Table 3. Experimental parameter settings.

Parameter name	Values
m	100
n	500
The number of rounds	120
MAX_ITER for DMF	8000
Range of GTD	[0.01, 436.67]
μ	[4, 20]
τ	[4, 10]
β	[0.1, 0.5]
TR	[0.1, 0.9]

by these algorithms as the baseline to evaluate workers' DOT based on the same trust computing function as DLFTI. And to achieve worker recognition, we truncate the workers' group for different TRs, considering the workers with high DOT as trustworthy and the workers with low DOT as untrustworthy.

Moreover, we implement FTI method for better comparison. The biggest difference between the FTI method and our DLFTI method is that the FTI method does not use DMF to obtain the bronze GTD, but directly uses the estimated data obtained by the Mean method as the bronze GTD. We introduce FTI as a comparison here mainly for two purposes: On the one hand, we can observe the advantages of our three-level GTD and ETD mechanism over other traditional comparison algorithms through FTI. On the other hand, we can also compare the advantages of our DMF complementation method over the non-DMF.

In this simulation, we assume that the error of the data collected by UAVs is negligible. Except the gold GTD obtained by UAVs, DPC will not know the real ground truth of any task. But when calculating the mechanism's accuracy, we will compare the ETD with the real ground truth in the dataset to calculate the accuracy.

5.3 Evaluation results

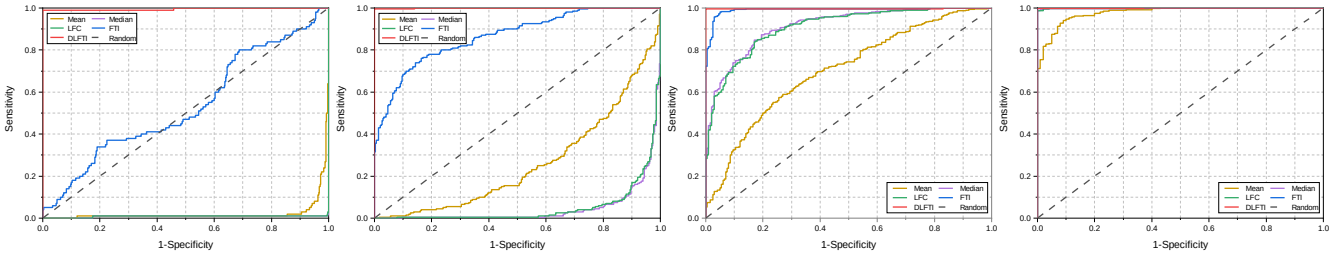
We investigate the performance of our algorithm from the perspective of worker recognition and truth discovery, respectively, and exhibit the performance of other algorithms.

First, ROC curve and AUC value are introduced to demonstrate the performance of our algorithm in worker recognition and compare it with other compared algorithms. The results indicate that our DLFTI algorithm performs significantly better than the comparison algorithms on the ROC curves for all cases of trustworthy rate. Remarkably, when the malicious workers are in the majority, the AUC achieved by DLFTI is at least 75% higher than Mean, Median and LFC. In the case where TR is equal to 0.2, the AUC value of our DLFTI algorithm is even more than 9 times higher than Mean, Median and LFC. In fact, the AUC of DLFTI is always close to 1 for all TRs.

Second, RMSE and FR are introduced to demonstrate the performance of our algorithm in truth discovery and similarly make comparisons with other compared algorithms. It can be seen that our DLFTI shows clearly better performance on RMSE and FR than the comparison algorithms for all TRs. Notably, when the malicious workers are in the majority, the RMSE achieved by the DLFTI algorithm is at least 20 times lower than the Mean, Median and LFC, and the FR by our DLFTI algorithm achieved is at least 104% higher than the other three algorithms. And it is revealed that the lower the TR, the more significant the advantage of our DLFTI's performance over other comparison algorithms.

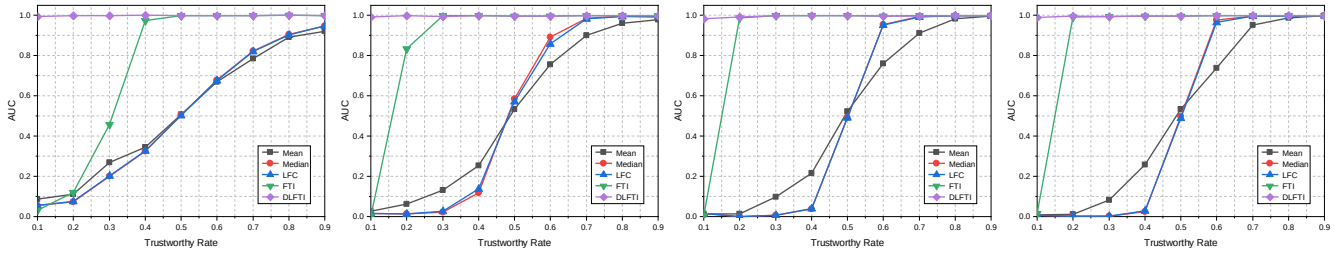
5.3.1 Worker recognition

To explore the algorithms' competence of worker recognition, we plot the ROC curves of our DLFTI and other comparison algorithms on worker recognition for different TRs, as shown in Fig. 6. It can be seen that our DLFTI shows significantly better performance than the comparison algorithms on the ROC curves for worker recognition in all TRs, and this advantage is especially dramatic when the TR is very small.



(a) $TR = 0.2$ (b) $TR = 0.4$ (c) $TR = 0.6$ (d) $TR = 0.8$

Fig. 6. ROC at different trustworthy rates. Mean [25], Median [25], LFC [25].



(a) $\mu = 5$ (b) $\mu = 10$ (c) $\mu = 15$ (d) $\mu = 20$

Fig. 7. AUC vs. the number of workers for each task μ. Mean [25], Median [25], LFC [25].

It is worth noting that the extreme case happens when the TR is less than 0.5. At this time the malicious workers occupy the majority of the alternative workers, and the compared methods are completely paralyzed. As shown by their ROC curves being completely below $y = x$, which suggests that their performance of worker recognition is even poorer than random judgment. Our algorithms maintain perfect results in all TRs, demonstrating that our algorithms have a very strong performance against gang fraudulent attacks. This is because:

(1) Our DLFTI mechanism fully considers the historical and current performance of the workers to weight their DOT, ensuring that we can recognize workers comprehensively to avoid recognition bias caused by chance.

(2) We can complete the missing GTD through DMF, so it is possible to evaluate the workers' DOT with high accuracy even when the gold and silver GTD are extremely scarce, which makes the performance of DLFTI stable for TR.

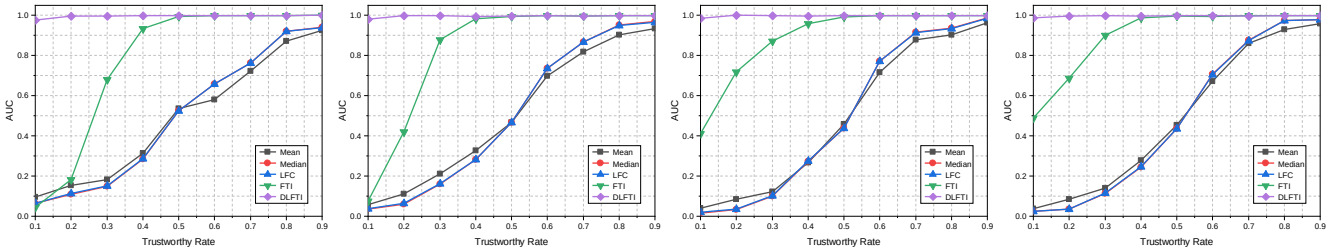
Then, we investigate the impact of TR from 0.1 to 0.9 with an increment of 0.1 and exhibit the performance of different algorithms in terms of AUC score with different μ and τ, as shown in Figs. 7-8.

In Fig. 7, we can observe that DLFTI and FTI both achieve much higher AUC scores than the Mean, Medium and LFC. At the same time, we can notice that as μ becomes larger, several algorithms for comparison tend to be more extreme, i.e., when μ is smaller, AUC varies more slowly with TR, while when μ is larger, AUC varies very significantly with TR. This phenomenon is because as more workers are selected for each task, ETD computation using mathematical statistics is easier to fall into the majority trap. So, when there are more malicious workers than trustworthy ones, mathematical statistics-based systems are more vulnerable to fraudulent gang attacks.

In Fig. 8, for the change of τ, Mean, Median and LFC do not change dramatically, but the overall curve becomes smoother as the τ increases. This phenomenon is because when the τ is small, the randomness is stronger, and therefore the overall system lacks stability, while when the τ is large, each algorithm makes full use of the statistical laws so the changes are smoother. It is worth noting that the FTI algorithm changes more significantly because FTI has a positive incentive for trustworthy workers. Therefore, when the τ is larger the FTI algorithm is more likely to perceive trustworthy workers, and this positive incentive promotes more quickly after the trustworthy people are identified. Hence, FTI is very sensitive to τ.

In Fig. 9, we evaluate the worker recognition performances of DLFTI by changing cycles and exhibit the performance of different algorithms in terms of trustworthy and untrustworthy workers' average DOT and Coverage Rate of Gold and Silver GTD (CRGSG). In the 10-th round, the trustworthy workers' average DOT have reached above 90, while the untrustworthy workers' average degrees of trust have dropped to 20, which shows the convergence speed of DLFTI is pretty fast.

Since worker recognition is related to GTD, we analyzed the CRGSG to study the worker recognition speed of DLFTI, and also analyzed the effects of different amounts of β, τ, and UAVs on the worker recognition speed, as shown in Figs. 10-12. It is observed that with the increase in the number of β, μ, and UAVs, the worker recognition speed of DLFTI gets significantly increased. Since DLFTI behaves more aggressively as β increases, it can perform worker recognition faster. As μ increases, more workers complete each task, leading to more highly trustworthy workers being identified, allowing DMF to get better results, thus speeding up worker recognition. As the number of UAVs increases, the number of gold GTD in each round increases, allowing DMF to get better results and thus perform worker identification faster.



(a) $\tau = 4$ (b) $\tau = 6$ (c) $\tau = 8$ (d) $\tau = 10$

Fig. 8. AUC vs. the number of rounds in a cycle τ. Mean [25], Median [25], LFC [25].

5.3.2 Truth discovery

To explore the algorithms' competence in truth discovery, we investigate the impact of trustworthy rate from 0.1 to 0.9

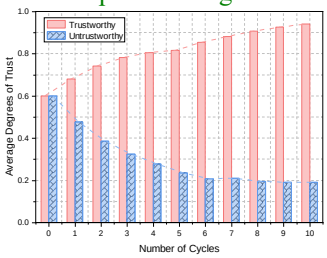


Fig. 9. Average DOT vs. the number of cycles.

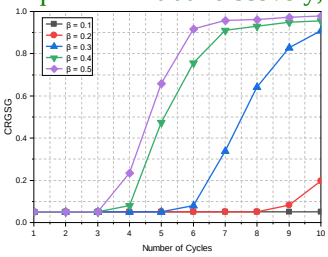


Fig. 10. CRGSG in different β.

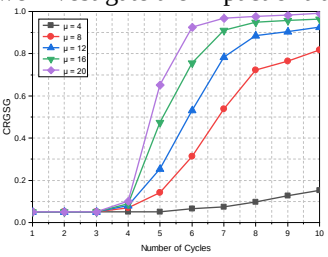


Fig. 11. CRGSG in different μ.

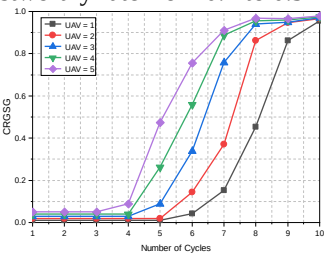
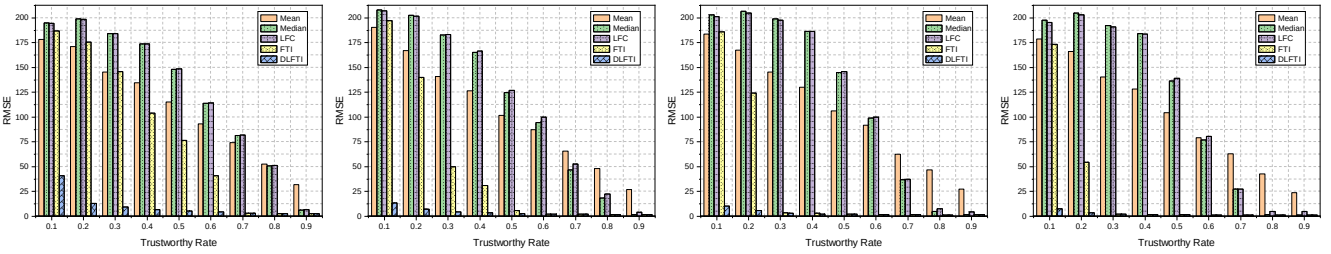
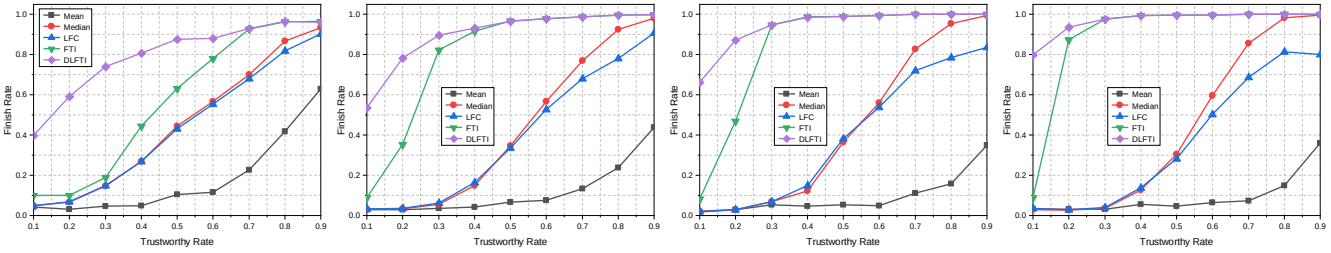


Fig. 12. CRGSG in different numbers of UAVs.



(a) $\mu = 5$ (b) $\mu = 10$ (c) $\mu = 15$ (d) $\mu = 20$

Fig. 13. RMSE vs. the number of workers for each task μ. Mean [25], Median [25], LFC [25].



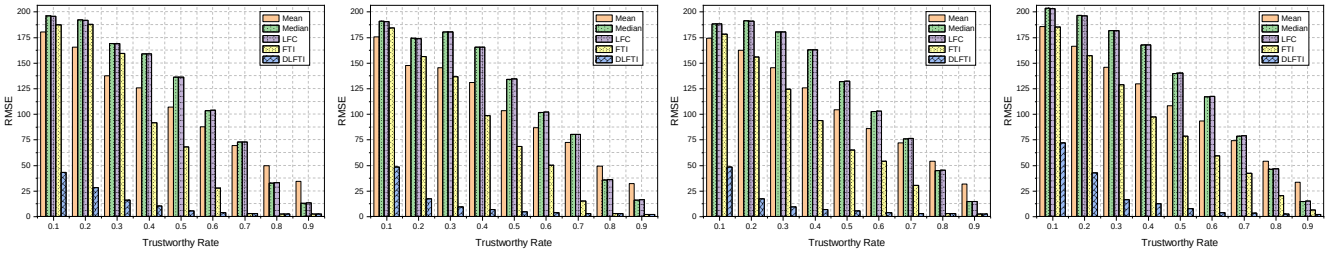
(a) $\mu = 5$ (b) $\mu = 10$ (c) $\mu = 15$ (d) $\mu = 20$

Fig. 14. FR vs. the number of workers for each task μ. Mean [25], Median [25], LFC [25].

with an increment of 0.1 and exhibit the performance of different algorithms in terms of RMSE and FR with different μ and τ, as shown in Figs. 13-16.

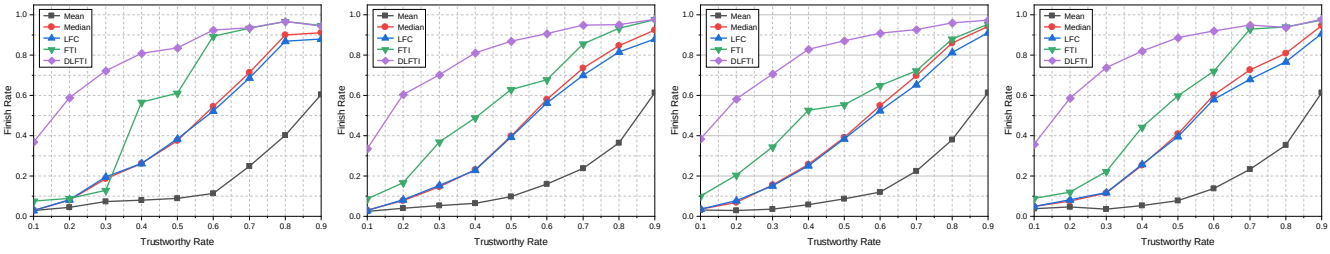
From Figs. 13-14, we can see that the change of μ has little effect on the RMSE and FR of Mean, Median and LFC, but it has an impact on our DLFTI algorithm. Specifically, as μ increases, the overall RMSE decreases and the FR increases on our DLFTI algorithm. This is because the comparable algorithms mainly rely on statistics, but for our DLFTI and FTI algorithms, when μ increases, it means that the number of our silver GTD and silver ETD will increase accordingly, which dramatically improve our truth discovery performance.

In Figs. 15-16, it is clear that our DLFTI always occupies the best truth discovery performance, and it does not change so significantly as τ is changed. Meanwhile, the RMSE and FR metrics of Mean are greater than Median and LFC when TR is less than 0.7, while the opposite is true when TR is greater than or equal to 0.7. For the FTI algorithm, when TR is less than 0.2, its RMSE performance does not differ much from mean, median and LFC, but as TR increases, the RMSE of FTI drops rapidly, while the FR metric increases rapidly and far exceeds the other three compared algorithms. It shows the effectiveness of our multi-level GTD and multi-level ETD; even without DMF for complementation, FTI still achieves outstanding combined characteristics with the multi-level GTD and multi-level ETD architecture.



(a) $\tau = 4$ (b) $\tau = 6$ (c) $\tau = 8$ (d) $\tau = 10$

Fig. 15. RMSE vs. the number of rounds in a cycle τ. Mean [25], Median [25], LFC [25].



(a) $\tau = 4$ (b) $\tau = 6$ (c) $\tau = 8$ (d) $\tau = 10$

Fig. 16. FR vs. the number of rounds in a cycle τ. Mean [25], Median [25], LFC [25].

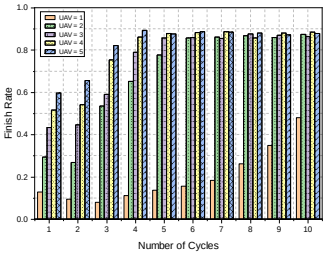


Fig. 17. Evaluation of the number of UAVs.

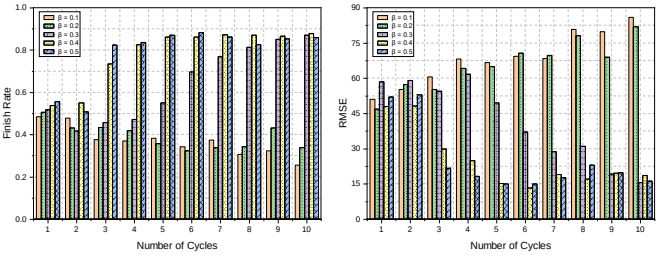
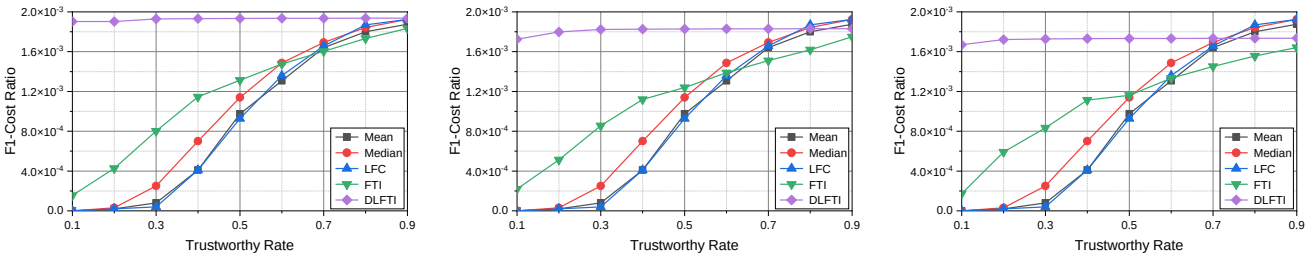


Fig. 18. Evaluation of the parameter β.

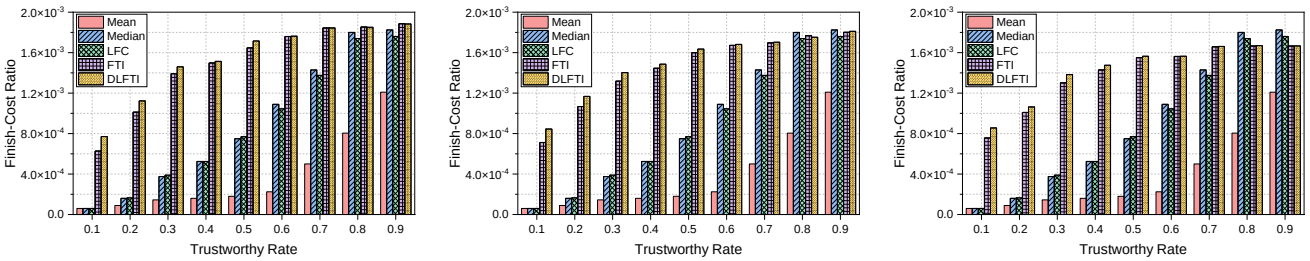
Fig. 17 shows the effect of dispatching different numbers of UAVs in a round on the variation of FR and RMSE with cycles. Overall, the FR metric of DLFTI gradually increases with increasing cycles, while RMSE gradually decreases with increasing cycles. It is worth mentioning that at the beginning of the DLFTI run (the first 4 rounds), both RMSE and FR show a clear stepped pattern, i.e., the more UAVs, the better the performance. As the cycles continue to increase, this stepped pattern begins to saturate slowly, and the FR and RMSE of DLFTI for all cases converge. This indicates that UAVs can speed up the convergence performance of DLFTI, thus solving the problem of the slow start. The more UAVs, the faster the DLFTI algorithm can converge to saturation performance. At the same time, this also guides us to dynamically change the UAVs' settings according to the actual situation. Specifically, in the initial few cycles, we can increase the number of UAVs to allow the system to converge quickly, and when the system converges to saturation, the number of UAVs can be reduced to reduce costs. Another interesting case is that when UAV=1, its RMSE curve appears a small increase and then gradually decreases. This case is because the number of UAVs is too small, which causes the multi-level GTD to be mostly dominated by inaccurate bronze GTD in the initial stage, so the system's performance will be impaired. However, as the truth discovery proceeds, the algorithm can still converge to the near saturation value, except that this process will be slower compared to the case of a larger number of UAVs.

Fig. 18 shows the effect of FR and RMSE at truth discovery with cycles when different β are set in a round. Intuitively, we can understand that the algorithm is biased towards conservative when β is smaller, while the algorithm is biased towards aggressive when β is larger. We can clearly see that when β is less than 0.3, the performance of its FR and RMSE decreases instead with the increase of cycles, which is because the system is too slow in learning the workers' DOT, resulting in a serious shortage of highly trustworthy workers in the early stage, and thus a lot of bronze ETD is used as inferring the truth. However, when β reaches above 0.3, both its FR and RMSE performance converge quickly to saturation values as the cycle increases. But too large β also brings the risk that the overall model is not stable enough and will have a large impact with minor fluctuations in workers' reported data. In fact, our hyperparameter β is similar to the learning rate in deep learning. When the hyperparameter β is too large or too small, it will lead to the degradation of the truth discovery performance of the whole MSC. A relatively nice strategy is to give a dynamic change to β. In the beginning, β can be set larger because of the need for fast start-up, and after the system has been stabilized and saturated, β can be adjusted smaller to improve the system's stability and avoid the damage to system performance caused by chance factors.



(a) The number of UAVs = 3 (b) The number of UAVs = 9 (c) The number of UAVs = 15

Fig. 19. F1-Cost Ratio vs. Different number of UAVs. Mean [25], Median [25], LFC [25].



(a) The number of UAVs = 3 (b) The number of UAVs = 9 (c) The number of UAVs = 15

Fig. 20. Finish-Cost Ratio vs. Different number of UAVs. Mean [25], Median [25], LFC [25].

5.3.3 Cost analysis

There is no doubt that by employing UAVs to collect data as GTD, our DLFTI approach can identify workers and discover truth more precisely. Therefore, we develop F1-Cost Ratio and Finish-Cost Ratio, to further characterize the utility for worker recognition and task completion at unit cost, respectively. Based on this, we compared the performance of DLFTI and other methods on F1-Cost Ratio and Finish-Cost Ratio when dispatching different number of UAVs, as shown in Figs 19-20.

In Fig. 19, it is clear that UAVs can have a significant effect on worker identification. Although the introduction of UAVs will increase the costs, the utility it can generate is much higher than the costs. Specifically, as shown in Fig. 19 (a), dispatching only a small number of UAVs (equals to 3) can result in significant gains for DLFTI. And it is clear that at this point, the proposed DLFTI can achieve state-of-art results at all trustworthy rates. However, as the number of UAVs introduced increases, the marginal utility of the F1-Cost Ratio gradually decreases in Fig. 19 (b) and Fig. 19 (c). As a result, DLFTI may not be able to continue its overwhelming advantage in high trustworthy rates.

In Fig. 20, it is illustrated that the utility of UAVs for truth discovery is very similar to that for worker identification. When the number of UAVs is proper, as shown in Fig. 20 (a), DLFTI can achieve the state-of-art effect in each trustworthy rate. However, when the number of UAVs reaches a certain level, its Finish-Cost Ratio decreases instead. Specifically, as shown in Fig. 20 (b) and Fig. 20 (c), DLFTI and FTI are rather less effective than the traditional Median and LFC methods when the trustworthy rate is greater than 0.8, for the ongoing expenses of dispatching excess UAV.

Summarizing with the above analysis, there is a phenomenon of diminishing marginal utility of dispatching UAVs. This inspires the platform to keep the number of UAVs in a reasonable range when adapting DLFTI. Often, just a small number of UAVs can obtain an optimal result, and it is not advisable to blindly dispatch an excessive number of UAVs.

5.4 Engineering applications

Since it is difficult to detect and collect distributed spatiotemporal data continuously, the data collected is often grossly insufficient to characterize the complete spatiotemporal distribution of the tasks accurately. With the continuous surge of mobile devices today, the sensing of large amounts of spatiotemporal data can be achieved by leveraging the sensing devices of crowds, making MCS a promising data collection paradigm. However, facing so many unknown workers and the massive spatiotemporal data they provide, the traditional MCS model is vulnerable to gang attacks by malicious workers. So, we propose the DLFTI method to achieve fast worker recognition and accurate truth discovery, which can robustly resist malicious attacks, by dispatching UAVs and using DMF method to mine the nonlinear relationships among the spatiotemporal data.

In the fields of climate, environment, and traffic, it is essential to collect, process and analyze the distributed spatiotemporal data. Therefore, by using our DLFTI, we can quickly and accurately get the trustworthy data we want, as shown in Fig. 20. The DLFTI approach makes full use of UAVs and the nonlinear relationships among the distributed spatiotemporal data to eliminate the interference of malicious workers. Specifically, the DPC first issue the spatiotemporal sensing tasks to workers and dispatch a fraction of UAVs to collect data with workers simultaneously. Then the data reported by UAVs and highly trustworthy workers are treated as gold and silver GTD. And next, the sparse GTD matrix is complemented by DMF to obtain bronze GTD, so that all the sensing tasks have a high-accuracy GTD for comparison. Finally, the worker profiles are updated and the double trustworthy workers are found, so the three-level ETD is established. Thus, the distributed spatiotemporal data is quickly and accurately obtained at a low cost.

In terms of the climate, we can recruit workers to monitor the atmospheric composition and use DLFTI to obtain the spatiotemporal climate data on a large scale to assist in making climate predictions for the distributed locations. Based on the

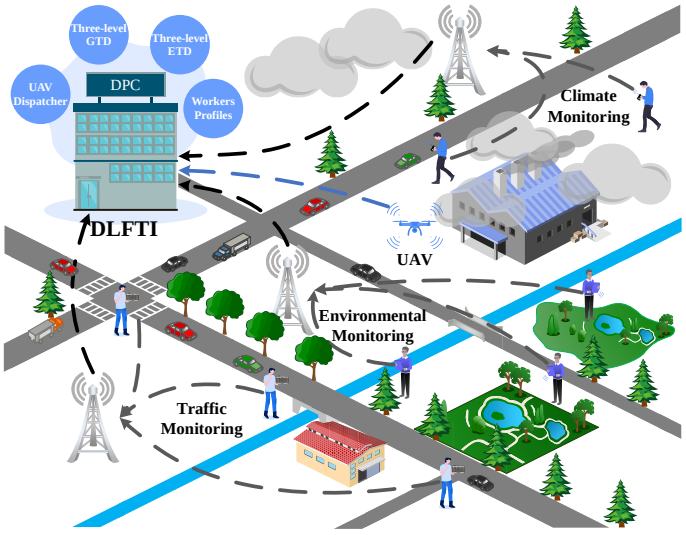


Fig. 20. Engineering applications of DLTFI.

climate prediction, not only can we issue more accurate warning information to reduce the impact of severe weather on people's lives. We can also provide more timely and accurate weather services to the people engaged in agriculture and pastoralism, thus, creating greater economic value.

In terms of the traffic, we can use MCS to let workers collect traffic flow in different spatiotemporal distributions, and purify the data by DLFTI method. Thus, the traffic situation of each spatiotemporal distribution is obtained, and an effective traffic circulation monitoring mechanism can be established. We can use the monitoring mechanism to provide reasonable support for the state of public traffic circulation, which can facilitate us to learn the spatiotemporal characteristics of vehicle travel and master the traffic flow state to provide support for road navigation, travel guide, parking selection and so on.

In terms of the environment, due to the large amount of industrial development and human activities at this stage, the forms of environmental pollution have become more and more abundant, and the traditional environmental monitoring methods have been difficult to meet the requirements today. Therefore, we can adopt the paradigm of MCS, and recruit workers to detect the diverse spatiotemporal distribution of environmental component data such as soil composition and water quality. And the data reported by workers can be purified by DLFTI method to obtain the environmental spatiotemporal data accurately. This will help us to establish a sounder environmental monitoring system, enabling us to achieve better water sources management, environmental management, and soil restoration. It can also help us to obtain early warnings of future environmental problems, providing more feasible and efficient pollution prevention solutions.

6. Conclusion and future works

In this paper, we focus on the worker recognition and truth discovery problem in MCS while simultaneously considering the unknown quality and gang of malicious workers. We propose a truth inference mechanism resistant to malicious gang attacks called DLFTI, which builds the multi-level GTD and multi-level ETD framework based on UAVs, already recognized trustworthy workers, and the DMF. Based on multi-level GTD, our DLFTI can achieve fast and efficient worker recognition, which can solve the slow start or even cold start problem caused by the unknown quality in the early stage of MCS. Meanwhile, based on multi-level ETD, our DLFTI can achieve accurate truth discovery to accomplish the expected goal of Crowd Sensing in high quality. Finally, we conduct extensive simulations on a real dataset to demonstrate the effectiveness of our mechanisms. The results show that our algorithm significantly outperforms the compared algorithms in three metrics, AUC score, RMSE and FR. And the advantages are especially dramatic when the malicious workers occupy most of the alternative workers.

In the future, we are planning to investigate more efficient methods for truth inference to better improve the development of MCS. Meanwhile, we will also devote to integrating our truth inference method with worker selection based on the balance of exploration and exploitation, especially considering the Multi-Armed Bandit (MAB) algorithm and auction theory. We hope that the worker recognition and truth discovery obtained by DLFTI can be effectively fed back to the data platform and assist in selecting workers to increase the revenue.

CRediT authorship contribution statement

Jianheng Tang and **Kejia Fan**: Conceptualization, Methodology, Software, Investigation, Formal analysis, Writing-original draft. **Pengzhi Yin** and **Zhenzhe Qu**: Investigation, Review & editing. **Anfeng Liu**: Conceptualization, Funding acquisition, Resources, Supervision, Writing-original draft, Writing-review & editing. **N. Xiong**: Supervision, review & editing. **Tian Wang**, **Mianxiong Dong**, **Shaobo Zhang**: Supervision, Review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (62072475).

References

[1] M. Karaliopoulos, E. Bakali. Optimizing mobile Crowd Sensing platforms for boundedly rational users. IEEE Transactions on Mobile Computing, 21 (2022) 1305-1318.
[2] F. Yucel, M. Yuksel, E Bulut. QoS-based budget constrained stable task assignment in mobile Crowd Sensing. IEEE Transactions on Mobile Computing, 20 (2021) 3194-3210.
[3] H. Sedghani, D. Ardagna, M. Passacantando, et al. An incentive mechanism based on a Stackelberg game for mobile Crowd Sensing systems with budget constraint. Ad Hoc Networks, 123 (2021) 102626.
[4] X. Zhu, Y. Luo, A. Liu, W. Tang, MZA. Bhuiyan. A Deep Learning-Based Mobile Crowd Sensing Scheme by Predicting Vehicle Mobility, IEEE Transactions on Intelligent Transportation Systems, 22 (2021) 4648-4659.
[5] Y. Ren, W. Liu, A. Liu, T. Wang, A. Li. A privacy-protected intelligent crowdsourcing application of IoT based on the Reinforcement Learning, Future generation computer systems, 127 (2022) 56-69.
[6] W. Mo, Z. Li, Z. Zeng, N. Xiong, S. Zhang, A. Liu. SCTD: A Spatiotemporal Correlation Truth Discovery Scheme for Security Management of Data Platform, Future generation computer systems, 139 (2023) 109-125.
[7] N. Maisonneuve, M. Stevens, M. E. Niessen, and L. Steels, Noisetube: Measuring and mapping noise pollution with mobile phones, Information Technologies in Environmental Engineering, (2019) 215–228.
[8] R. K. Rana, C. T. Chou, S. S. Kanhere, N. Bulusu, and W. Hu, Ear-phone: an end-to-end participatory urban noise mapping system, in Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks, Stockholm, Sweden, April 2010.
[9] E. Koukoumidis, L.-S. Peh, and M. R. Martonosi, Signalguru: leveraging mobile phones for collaborative traffic signal schedule advisory, in Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services, Bethesda, Maryland, USA, June 2011.
[10] Y. Wang, X. Liu, H. Wei, G. Forman, C. Chen, and Y. Zhu, Crowdatlas: self-updating maps for cloud and personal use, in Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services, Taipei, Taiwan, June 2013.
[11] X. Wang, R. Jia, L. Fu, H. Jin, X. Tian, et al. Online Spatial Crowd Sensing With Expertise-Aware Truth Inference and Task Allocation. IEEE Journal on Selected Areas in Communications, 40 (2022) 412-427.
[12] X. Wang, R. Jia, X. Tian, X. Gan, L. Fu, et al. Location-aware Crowd Sensing: Dynamic task assignment and truth inference. IEEE Transactions on Mobile Computing, 19 (2018) 362-375.
[13] X. Zhang, Y. Wu, L. Huang, H. Ji, et al. Expertise-aware truth analysis and task allocation in mobile crowdsourcing. IEEE Transactions on Mobile Computing, 20 (2019) 1001-1016.
[14] P. Sun, Z. Wang, L. Wu, Y. Feng, X. Pang, et al. Towards personalized privacy-preserving incentive for truth discovery in mobile Crowd Sensing systems. IEEE Transactions on Mobile Computing, 21 (2020) 352-365.
[15] S. Lyu, W. Ouyang, Y. Wang, H. Shen, et al. Truth discovery by claim and source embedding. IEEE Transactions on Knowledge and Data Engineering, 33 (2021) 1264-1275.
[16] X. Gao, S. Chen and G. Chen. MAB-Based Reinforced Worker Selection Framework for Budgeted Spatial Crowdsensing, IEEE Transactions on Knowledge and Data Engineering, vol. 34 (2022) 1303-1316.
[17] C. Zhao, S. Yang, JA. McCann. On the data quality in privacy-preserving mobile Crowd Sensing systems with untruthful reporting. IEEE Transactions on Mobile Computing, 20 (2021) 647-661.
[18] C. Ye, H. Wang, K. Zheng, YK. Kong, et al. Constrained truth discovery. IEEE Transactions on Knowledge and Data Engineering, 34 (2022) 205-218.
[19] S. Ye, J. Wang, H. Fan, Z. Zhang. Probabilistic model for truth discovery with mean and median check framework. Knowledge-Based Systems, 233 (2021) 107482.
[20] L. Fu, J. Xu, S. Qu, Z. Xu, X. Wang, et al. Seeking the Truth in a Decentralized Manner. IEEE/ACM Transactions on Networking, 29 (2021) 2296-2312.
[21] X. Wu, YE. Sun, Y. Du, G. Gao, H. Huang, X. Xing, et al. An Anti-Malicious Task Allocation Mechanism in Crowd Sensing Systems. Future Generation Computer Systems, 127 (2022) 347-361.
[22] H. Shao, D. Sun, S. Yao, L. Su, Z. Wang, et al. Truth discovery with multi-modal data in social sensing. IEEE Transactions on Computers, 70 (2020) 1325-1337.
[23] X. Gao, H. Huang, C. Liu, F. Wu and G. Chen. Quality Inference Based Task Assignment in Mobile Crowdsensing, IEEE Transactions on Knowledge and Data Engineering, 33 (2021). 3410-3423.
[24] I. Rasheed. Enhanced privacy preserving and truth discovery method for 5G and beyond vehicle crowd sensing systems. Vehicular Communications, 32 (2021) 100395.

[25] Y. Zheng, G. Li, Y. Li, C. Shan, R. Cheng. Truth inference in crowdsourcing: Is the problem solved? Proceedings of the VLDB Endowment, 10 (2017) 541-552.
[26] XS. Fang, QZ. Sheng, X. Wang, WE. Zhang, et al. From appearance to essence: comparing truth discovery methods without using ground truth. ACM Transactions on Intelligent Systems and Technology (TIST), 11 (2020) 1-24.
[27] H. Lu, X. Gao, G. Chen. Efficient Crowdsourcing-Aided Positioning and Ground-Truth-Aided Truth Discovery for Mobile Wireless Sensor Networks in Urban Fields. IEEE Transactions on Wireless Communications, 21 (2022) 1652-1664.
[28] Y. Du, YE. Sun, H. Huang, L. Huang, H. Xu, et al. Bayesian co-clustering truth discovery for mobile crowd sensing systems. IEEE Transactions on Industrial Informatics, 16 (2020) 1045-1057.
[29] T. Li, A. Liu, S. Zhang, T. Wang, N. Xiong. A Trustworthiness-based Vehicular Recruitment Scheme for Information Collections in Distributed Networked Systems, Information Sciences, 545 (2021) 65-81.
[30] Z. Liu, K. Li, X. Zhou, N. Zhu, Y. Gao, K. Li. Multi-stage complex task assignment in spatial crowdsourcing, Information Sciences, 586 (2022) 119-139.
[31] R. Zhang, Z. Li, N. Xiong, S. Zhang, A. Liu. TDTA: A Truth Detection based Task Assignment Scheme for Mobile Crowdsourced Industrial Internet of Things, Information Sciences, 610 (2022) 246-265.
[32] Y. Dong, L. Jiang, C. Li. Improving data and model quality in crowdsourcing using co-training-based noise correction, Information Sciences, 583 (2022) 174-188.
[33] D. Sarma, A. Das, P. Dutta and U. K. Bera. A Cost Minimization Resource Allocation Model for Disaster Relief Operations With an Information Crowdsourcing-Based MCDM Approach, IEEE Transactions on Engineering Management, 69 (2022) 2454-2474.
[34] M. Xiao, B. An, J. Wang, G. Gao, S. Zhang and J. Wu. CMAB-Based Reverse Auction for Unknown Worker Recruitment in Mobile Crowdsensing, IEEE Transactions on Mobile Computing, 21 (2022) 3502-3518.
[35] Z. Zheng, S. Yang, J. Xie, F. Wu, X. Gao and G. Chen. On Designing Strategy-Proof Budget Feasible Online Mechanisms for Mobile Crowdsensing With Time-Discounting Values, IEEE Transactions on Mobile Computing, 21 (2022) 2088-2102.
[36] K. Li, S. Wang, X. Cheng and Q. Hu. A Misreport- and Collusion-Proof Crowdsourcing Mechanism Without Quality Verification, IEEE Transactions on Mobile Computing, 21 (2022) 3084-3095.
[37] J. Guo, A. Liu, K. Ota, M. Dong, X. Deng, N. Xiong. ITCN: An Intelligent Trust Collaboration Network System in Industrial IoT, IEEE Transactions on Network Science and Engineering, 9 (2022) 203-218.
[38] J. Guo, G. Huang, Q. Li, N. N. Xiong, S. Zhang, T. Wang. STMTO: A Smart and Trust Multi-UAV Task Offloading System, Information Sciences, 573 (2021) 519-540.
[39] Y. Zhan, Y. Xia, J. Zhang, T. Li, Y. Wang. An incentive mechanism design for mobile crowdsensing with demand uncertainties, Information Sciences, 528 (2020) 1-16.
[40] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy. Learning from crowds. JMLR, 11 (2010) 1297–1322.
[41] J. Liang, W. Liu, N. Xiong, A. Liu, S. Zhang. An Intelligent and Trust UAV-assisted Code Dissemination 5G System for Industrial Internet-of-Things, IEEE Transactions on Industrial Informatics, 18 (2022) 2877-2889.
[42] E. Wang, M. Zhang, X. Cheng, Y. Yang, et al. Deep learning-enabled sparse industrial Crowd Sensing and prediction. IEEE Transactions on Industrial Informatics, 17 (2020). 6170-6181.
[43] L. Kong, X. Tang, J. Zhu, Z. Wang, J. Li, et al. A 6-year-long (2013–2018) high-resolution air quality reanalysis dataset in China based on the assimilation of surface observations from CNEMC, Earth Syst. Sci. Data, 13 (2021) 529–570, https://doi.org/10.5194/essd-13-529-2021.