

Review Article

Key Research Issues and Related Technologies in Crowdsourcing Data Collection

Yunhui Li ¹, Liang Chang ², Long Li ², Xuguang Bao ² and Tianlong Gu ³

¹School of Information and Communication, Guilin University of Electronic Technology, Guilin 541004, China

²Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin 541004, China

³College of Information Science and Technology, Jinan University, Guangzhou 510000, China

Correspondence should be addressed to Tianlong Gu; gutianlong@jnu.edu.cn

Received 8 June 2021; Accepted 28 September 2021; Published 16 October 2021

Academic Editor: Lihua Yin

Copyright © 2021 Yunhui Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Crowdsourcing provides a distributed method to solve the tasks that are difficult to complete using computers and require the wisdom of human beings. Due to its fast and inexpensive nature, crowdsourcing is widely used to collect metadata and data annotation in many fields, such as information retrieval, machine learning, recommendation system, and natural language processing. Crowdsourcing helps enable the collection of rich and large-scale data, which promotes the development of researches driven by data. In recent years, a large amount of effort has been spent on crowdsourcing in data collection, to address the challenges, including quality control, cost control, efficiency, and privacy protection. In this paper, we introduce the concept and workflow of crowdsourcing data collection. Furthermore, we review the key research topics and related technologies in its workflow, including task design, task-worker matching, response aggregation, incentive mechanism, and privacy protection. Then, the limitations of the existing work are discussed, and the future development directions are identified.

1. Introduction

Machine learning and deep learning technologies have increasingly become a research topic in many fields, including computer vision, natural language processing, and other fields related to artificial intelligence. The study of these techniques requires large-scale, high-quality data (raw and/or labeled data) to train algorithms, and the quantity and quality of the data directly affect the performance of the trained algorithm. How to collect large-scale, high-quality data is an urgent problem to be solved.

Crowdsourcing [1] provides a distributed data collection solution. We call this solution “crowdsourcing data collection” and define it as

“Crowdsourcing data collection is the scheme of undertaking collecting data tasks by an undefined, potentially large group of online workers in an open recruit format.”

There are many examples of crowdsourcing data collection. ImageNet (<http://www.image-net.org/about-stats>), a dataset of more than 14 million images, was labeled by 50,000 online users on Amazon Mechanical Turk (AMT)

(<https://deepmind.com/research/open-source/kinetics>). Kinetics (<https://deepmind.com/research/open-source/kinetics>), a dataset of human behavior that includes 700 motion categories and nearly 650,000 video clips, was collected via YouTube. LibriSpeech (<http://www.openslr.org/12/>), a speech corpus containing about 1000 hours of English, is from the LibriVox project. The famous Yelp dataset, from the largest public comments on Yelp (<https://www.yelp.com/dataset>), contains more than 8 million user comments and more than 20 images of over 200,000 businesses in 10 cities.

Crowdsourcing helps enable the collection of rich and large-scale data, which promotes the development of researches driven by data. However, crowdsourcing data collection relies on the uncertain crowd; the differences in people's ability and understanding of questions, as well as the motivation to participate in the task, will affect the effectiveness and efficiency of crowdsourcing, as well as harm the privacy of requesters and workers. Some technologies are applied to the crowdsourcing process to control the quality, cost, efficiency, and preserving privacy. These techniques focus on solving the following key issues: how to design a

task, how to select a worker (i.e., people who perform tasks), how to aggregate workers' responses, how to design an incentive mechanism, and how to protect privacy from disclosure. This survey describes the process of crowdsourcing data collection, reviews the key research topics and related technologies in its workflow, and discusses the limitations of the existing work and open problems.

This paper is organized as follows. Section 2 introduces the crowdsourcing data collection process, Sections 3–7 review the technologies adopted from five key aspects, respectively, including (1) task design, (2) task-worker matching, (3) response aggregation data, (4) incentive mechanism design, and (5) privacy-preserving. Section 8 discusses the limitations of the existing work and the future research direction. Section 9 concludes this paper.

2. Crowdsourcing Data Collection Process

Crowdsourcing data collection infrastructure comprises three major components: requester, worker, and crowdsourcing platform. A requester is a task owner, such as a person or an organization that requests a particular data collection task to be completed by workers (see Figure 1). A worker is an online user who potentially performs an assigned/selected task, motivated by interest or reward. A crowdsourcing platform is a server that manages requesters, workers, and tasks.

Figure 1 shows the process of crowdsourcing data collection. First, the requester submits the designed task and the corresponding reward to the platform (Step 1 in Figure 1). Then, the crowdsourcing platform publishes tasks (Step 2 in Figure 1). Then, the worker performs the assigned/selected tasks (Step 3 in Figure 1) and responds to the platform with collected data (Step 4 in Figure 1). Then, the platform aggregates the responses from workers and delegates them to the requester (Step 5 in Figure 1). Finally, the requester validates the task responses and determines whether to accept them, and once accepted, the reward is paid to the worker who has responded to the task (Step 6 in Figure 1).

Crowdsourcing data collection has the advantages of cheap price, fast collection speed, and large scale of data obtained. However, it still faces many challenges:

- (1) *Control of the Crowdsourcing Result's Quality.* The crowdsourcing result's quality refers to the extent to which the data obtained meets and/or exceeds the requestor's expectations. The quality is affected by two aspects: task and worker. First, the task design (including whether a task description is clear and whether a task design is reasonable) has a direct influence on the worker's understanding of the task. If workers cannot accurately understand the task, it is difficult to provide high-quality data. Secondly, crowdsourcing mainly uses the online worker to collect data. Because the objective ability and subjective motivation of the worker may affect the reliability and/or the correctness of the collected data, it is certainly difficult to ensure the quality of their submitted data.

- (2) *Control of the Cost.* From the task owner's perspective, the costs of crowdsourcing refer to the payment required to accomplish the task. Most crowdsourced tasks require the task owner to pay rewards to workers who have completed the task. On the AMT platform, the reward is usually a few cents per task. However, the total payment is a considerable expense, if the scale of the task is large. For example, if the price to tag a single image is 5 cents, the price to tag 50,000 images is \$2,500, so it is important to consider not only the quality of the crowdsourced data but also the cost of doing so. In contrast, from the worker's point of view, since they need to expend the cost (including time, energy, and resources) to participate in tasks, they usually consider whether the reward is worthwhile compared with the cost.

- (3) *Efficiency, which Refers to the Time between Publishing the Task and Completing the Task.* Efficiency is affected by the enthusiasm to participate in the task and the quality of the task completed by the workers. For example, 100 workers tag a set of photos faster than 10 workers. However, among the data submitted by workers, if the amount of qualified data is lower than the task requestor expects, a secondary publication task is required, increasing the overall time for the task to complete.

- (4) *Privacy Threat, which Is an Important Issue in Crowdsourcing.* The data collected through crowdsourcing may contain a large amount of sensitive information, which is directly related to user privacy, such as the user's geographical location, travel trajectory, and personal preferences. This would cause serious security threats. For example, based on the personal information collected and tracked, Egyptian government officials' harassment and retaliation on Ushahidi reporters in 2011 can be seen as both physical intrusions to those protestors' solitude and an interference against their ideas and public demonstrations.

To address the above challenges, recently, research on crowdsourcing data collection has sprung up, for example, controlling the quality of crowdsourcing results [2–4], balancing between budget and quality [5–7], and aggregating the responses generated by workers to produce accurate results [8–10].

These challenges exist in the entire crowdsourcing process. Next, according to the crowdsourcing process, a variety of studies are reviewed from 5 aspects, including (1) task design, (2) task-worker matching, (3) response aggregation, (4) incentive mechanism design, and (5) privacy-preserving.

3. Task Design

Crowdsourcing task design is to design a task with a clear description and appropriate size to improve the readability

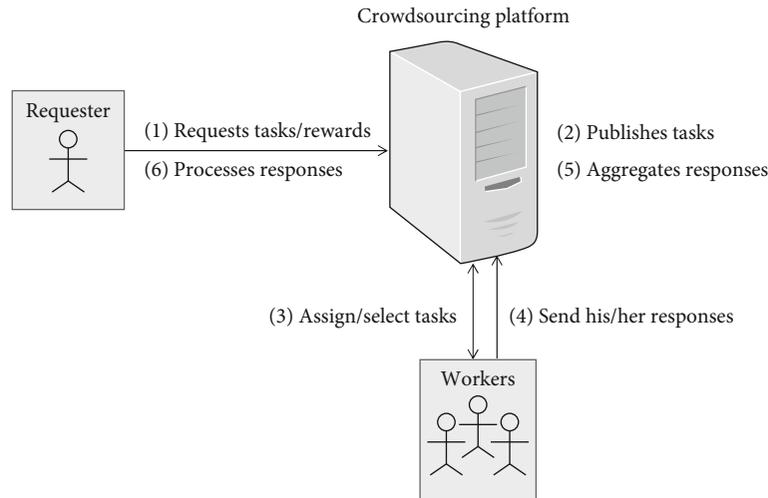


FIGURE 1: Crowdsourcing data collecting scenario.

of the task, to help workers complete the task quickly and correctly.

3.1. Task Description. Task description describes the task basic information (e.g., title, keywords, task content, task requirement, and task goal) and the task instructions of how to perform the task. The clarity of the task description affects the way workers perform the task and hence the quality of the crowdsourcing results [11–14].

Few task description studies have been conducted at this time. Gadiraju et al. [15] studied the quantification of task clarity. They published 71,000 microtasks on the CrowdFlower platform, including six task types: CC (Content Creation), IF (Information Finding), IA (Interpretation and Analysis), VV (Verification and Validation), CA (Content Access), and SU (Surveys). They collected workers' ratings on task goal clarity, task role clarity, and task clarity, then used the features (e.g., task type and task content) and the acquired labels to train and validate a supervised machine learning model for task clarity prediction. Gillier et al. [16] studied the influence of task instruction orientation on the quality of task (i.e., crowdsourcing innovative ideas) completion. They compared the quality of task completion under three types of task instructions: unbound, suggestive, and prohibitive. Suggestive task instruction leads to lower quality of idea originality, probably because it limits people's thinking. Wang et al. [17] believed that, if the samples in the suggestive instruction were highly original, it would motivate workers to produce high-quality original works. The research of Ipeirotis [18], on the AMT (Amazon Mechanical Turk) platform, suggests that task completion times were constrained by the way tasks were selected and followed a power-law distribution. Most tasks take 12 hours or 7 days to complete. Besides, graphic design [15] and gamification design [19, 20] not only make tasks more attractive to workers but also enhance workers' understanding of the task.

3.2. Task Decomposition. The size of a task affects the speed and quality with which it is completed. Microtasks with low

granularity, such as image tagging and text recognition, generally do not require much professional skill and can be completed quickly. Macrotasks with great granularity, such as editing an article and writing a travel guide, are complex, require specific professional skills, and are difficult to accomplish by one person alone. Macrotasks usually need to be decomposed into multiple subtasks to reduce the difficulty and granularity of tasks, so as to improve the quality of task results and shorten the completion time [21–23].

According to the participants involved in task decomposition, the task decomposition method is divided into independent task decomposition and cooperative task decomposition. Independent task decomposition means that the task decomposition is completed independently by the task requestor. Collaborative task decomposition means that task decomposition is accomplished collaboratively by task requesters and workers. For example, Kulkarni et al. [23] designed an editable visual tool Turkomatic to allow workers to participate in the decomposition of crowdsourced tasks.

From the content of task decomposition, the task decomposition method can be divided into vertical task decomposition and horizontal task decomposition. The vertical decomposition method decomposes the task into multiple subtasks in a sequential sequence. The output of the former subtask is taken as the input of the latter subtask, and the output of the last subtask is the final output of the original task. For example, Bernstein et al. [21] split the text editing task into three simple subtasks: (1) find, finding what needs to be fixed; (2) fix, fixing what needs to be fixed; and (3) verify, verifying the correctness of the fix. The horizontal method divides tasks into multiple subtasks that can be done in parallel. The final output of the original task is obtained by aggregating the output of all subtasks. For example, Kitur et al. [22], based on the MapReduce framework [24], studied the decomposition of complex tasks and the integration of responses to subtasks and split the task writing an article into three simple subtasks: Partition-Map-Reduction. The "Partition" subtask is to create an outline. The "Map" subtask is to collect materials required for a chapter. Multiple instances of a "Map" subtask can be done

in parallel. The “Reduction” subtask writes paragraphs based on the collected materials. Finally, the “Reduction” subtask merges all the outlines and chapters into a single article.

4. Task-Worker Matching

There are two ways of matching between tasks and workers: (1) worker selection task and (2) platform assignment task.

- (i) *Worker-selected tasks (WST)*: workers select data collection tasks of their interest from a given list published by the crowdsourcing platform.
- (ii) *Platform-assigned tasks (PAT)*: the crowdsourcing platform selects available appropriate workers for a given data collection task based on various parameters, such as the quality of the worker and the budget of the task, to ensure that certain goals are achieved.

4.1. Worker-Selected Tasks (WST). With regard to WST, a worker searches the list of tasks in some sort or by entering keywords. For example, AMT, the most popular crowdsourcing platform for microtasks, sorts tasks by “recently released,” “reward,” and “most HITS” [25] and allows inputting keywords for searching tasks [18]. WAT helps workers find tasks quickly, but it has some limitations:

- (1) Workers generally focus on the first 1~2 pages of search results, which means some tasks will not be completed for a long time, i.e., hungry task
- (2) The tasks found are likely not suitable for the worker. However, to save task search time, some workers choose tasks randomly from the search results, resulting in (i) the quality of the workers’ contribution being low and (ii) requestors losing contributions from other workers who are better suited to the task, and spend extra time dealing with suboptimal contributions

To complement the above search methods, some researchers propose task recommendation algorithms [26–30] based on worker characteristics and task characteristics, in order to provide workers with more appropriate tasks to choose from. Ambati et al. [28], based on the historical interactions between workers and tasks, built a preference model for workers and learned workers’ preferences through “Bag-of-Words Approach” and “Classification Based Approach,” so as to recommend tasks that might be of interest to workers. However, Ambati et al. [28] cannot solve the cold-start problem of a lack of historical information on new workers and tasks. To address the cold-start problem, Yuen et al. [26] proposed a task recommendation framework TaskRec based on Unified Probability Matrix Factor Decomposition, which is aimed at recommending tasks for workers in dynamic scenarios. In the real world, the time spent on completing a crowdsourcing data collection task is usually short (several minutes or even seconds), so it is possible that a task has been completed by other workers before it has been recommended to the right one.

Safran et al. [30] proposed a real-time recommendation algorithm that recommends the task within milliseconds, including the following: (1) Top-K-T algorithm, recommending the most suitable K tasks for specific workers; (2) Top-K-W algorithm, recommending the most suitable K workers for a specific task.

4.2. Platform-Assigned Tasks (PAT). PAT involves assigning a given task to suitable workers based on various conditions, aimed at achieving optimization goals benefitting the requester, such as maximizing the number of tasks assigned, minimizing the cost currently, and improving the quality of task responses, or goals benefitting the worker, such as maximizing the reward received by the worker. These goals are related to the quality of workers; therefore, how to assess the quality of workers is vital. Section 4.2.1 reviews the factors influencing the quality of workers, Section 4.2.2 introduces the assessment of the quality of workers, and Section 4.2.3 introduces the assignment algorithms.

4.2.1. Factors Influencing the Quality of Workers. The quality of workers is influenced by both the workers and the tasks, specifically, including the worker’s ability, the human factor of the worker, and the difficulty of the task.

The quality of workers is affected by their professional ability [25]. Workers perform better on crowdsourced tasks in areas of expertise they excel at [31]. A worker’s professional ability refers to the knowledge and skills acquired by the worker through previous studies and work, which reflects the ability level of the worker in a certain field. In general, the worker’s professional ability is evaluated based on his/her credentials (such as academic certificates, language level, and professional qualifications) and experience.

The quality of workers is affected by human factors [32, 33]. Kazai et al. [33] investigated the influence of human factors on the accuracy of labeling from six aspects: workers’ participation motivation, familiarity with the subject involved in the task, awareness of the difficulty of the task, satisfaction with reward, and enjoyment of the task, and found that the accuracy of labeling was related to human factors.

The quality of workers is affected by task difficulty [33, 34]. Wei et al. [34] divided tasks into easy and difficult categories and learned the difficulty of tasks according to the workers’ scores on the difficulty of tasks. The results of the image tagging experiment show that compared with easy tasks, difficult tasks have higher tagging accuracy, but this is related to the workers’ perception of the difficulty of the task [33].

4.2.2. Assessment of Worker’s Quality. An accurate assessment of the quality of workers is required before task assignment. Specifically, it can be divided into the following three evaluation methods shown in Table 1:

- (1) Evaluate the quality of workers according to their reputation (EQWR)
- (2) Evaluate the quality of workers by using the gold standard (EQWG)

TABLE 1: Assessment of worker's quality.

| Method/literature | Traits | Limitation |
|----------------------------------|---|--|
| EQWR [35, 37, 38, 36, 2, 33, 39] | Assume that the platform has obtained the worker's history of completion of the task. | The reputation value of the new user cannot be obtained. |
| EQWG [8, 40, 41, 33, 42] | Suppose the answer to the gold standard is known. | Increases the cost of the task requester and the time the worker takes to answer the question. Whether the gold standard is set suitable. |
| EQWA [8, 10] | Without knowing the correct answer to the task. The quality of the worker is inferred based on the response of the worker via data aggregation. | The limitation of the first two methods is solved, but the problem of long computation time may exist. |

- (3) Evaluate the quality of workers by the aggregation result of workers' responses (EQWA)

The quality of the worker is usually modeled by the reputation of the worker [35]. Reputation values are based primarily on explicit feedback (i.e., ratings of workers' contributions) from members of the crowdsourced community about workers' activities. For example, Xie et al. [36] evaluated workers' reputations based on the correctness of workers' responses. Allahbakhsh et al. [37] evaluated the reputation of the workers by using their timeliness and reliability in answering questions and their relationship with other workers or requestors. However, explicit feedback evaluation methods cannot avoid the influence of human factors such as the personal preferences or biases of the evaluator, which may result in an inaccurate assessment of the true quality of the worker [38]. Therefore, the evaluation method of implicit feedback appears, which is based on the worker's historical task completion and the worker/task profile. For example, the AMT platform typically considers highly reputable workers with more than 100 completed HIT (Human Intelligent Tasks) and more than 95% of those tasks accepted by the requester [2, 39]; these threshold values may be adjusted to accommodate your request. Reference [33] studied the accuracy of labeling under conditions of restricted and unrestricted worker qualifications; the results revealed that the accuracy is higher in the former condition.

Qualification tests for workers, using the gold standard contained in a task, is another way to assess the quality of workers [8, 40]. The gold standard refers to the known answers to the questions, usually used in qualification tests. The quality of workers is evaluated by the correct completion rate of the gold standard, so as to effectively evaluate the ability of workers to answer questions or the degree of attention to questions, thus filtering out low-quality workers. References [33, 41, 42] filtered out inattentive workers using Attention Check Questions (ACQs). The study [8] shows that the gold standard can effectively improve the accuracy of data annotation, by filtering out spammers. However, the addition of a gold standard would lead to additional tasks, resulting in increased costs for the requestor or an increase in unpaid work for the worker, which might make both parties reluctant to add a gold standard to the task.

Although the above two schemes realize the evaluation of worker quality, they both have some limitations: (1) EQWR assumes that the platform has obtained worker information; however, no information was available on the new worker or new task; (2) EQWG adds an additional cost to the requester and time spent for completion. Besides, the rationale for the gold standard is worth considering. EQWA addresses these limitations. Reference [10] uses a truth inference algorithm [8] to aggregate the workers' responses, infer the ground truth of the task, and evaluate the quality of the workers based on the ground truth. See Section 5 for a detailed description of the various aggregation methods.

4.2.3. Task Assignment. Crowdsourcing workers vary in their professional ability, work motivation, etc., resulting in different quality of workers when completing specific tasks, which makes it difficult to ensure the quality of crowdsourcing results. Although task redundancy and other methods improve the quality of crowdsourcing results [3], it will also lead to an increase in the cost paid by the task requestor and the time spent in response aggregation. Therefore, how to reasonably assign tasks to suitable workers has become one of the hottest research issues in crowdsourcing research.

At present, a large number of researches have been conducted on specific task assignment methods from the perspective of task requesters. The main idea is to balance the quality of crowdsourcing results, the number of tasks completed, and the cost (such as time and budget), to achieve the reasonable assignment of tasks.

Karger et al. [5–7] took classifying tasks as an example, aimed at obtaining reliable data annotation with minimum redundancy (the number of repeated assignments per task). In [5–7], the quality of a worker is modeled as a probability; the random regular bipartite graph is used to assign tasks to workers in the offline scenario. Ho et al. [43] proposed the exploration-exploitation algorithm in online scenarios, aimed at minimizing the total number of tasks assigned while the quality of crowdsourcing results is higher than the preset thresholds. Fan et al. [44] assumed that the quality of workers might differ in the different tasks they are engaged in and proposed an adaptive allocation framework, iCROWD. According to the similarity between tasks, a task is assigned to the workers who have performed better on similar tasks, to improve the quality and number of tasks completed as much as possible. The iRowd framework

TABLE 2: Factors considered and metrics of task assignments.

| Reference | Factors | | | Metrics | |
|-----------|-------------------------------|------------------|--------------------------|---------------------------------|----------------------|
| | Worker's professional ability | Worker's hobbies | Task's budget constraint | Quality of crowdsourcing result | Task assignment rate |
| [5] | × | × | √ | √ | × |
| [6] | × | × | √ | √ | × |
| [7] | × | × | √ | √ | × |
| [43] | √ | × | √ | √ | × |
| [44] | × | × | × | √ | × |
| [45] | √ | × | × | √ | √ |
| [46] | √ | × | × | √ | × |
| [47] | √ | √ | × | √ | × |
| [48] | √ | √ | × | √ | × |

includes a WarmUp component that is used to conduct qualification tests on new workers to assess their initial quality. Document [45] proposes an adaptive task allocation framework, Argo+, based on LDA (Latent Dirichlet Allocation) and Rocchio technology, to increase the success rate of task assignments. Reference [45] measures a worker's quality based on the similarity, calculated based on the worker's expertise and that required by a task. The new worker's expertise is provided by himself or set to any initial value. Literature [46] uses a decision tree to classify workers according to their expertise, then picks tasks for the worker he/she is good at, aimed at improving the quality of crowdsourcing results.

The above task assignment algorithms assume that the task to be assigned is a single task, which does not apply to the assignment of a combination task. A combination task is a task composed of several different microtasks. For example, a combination task might include making a city tour plan, selecting a book for a reading club, or rating a movie. A major feature of combined tasks is the diversity of task features [47]. Literature [48] studied the influence of task diversity on task assignment goals and proposed a task assignment method matching workers' professional abilities and hobbies.

Table 2 describes the factors considered and the metrics of various task assignments.

5. Response Aggregation

To improve the quality of a crowdsourcing result, the most commonly used crowdsourcing method based on redundancy is to assign one task to multiple workers and then aggregate the responses of multiple workers to produce a crowdsourcing result of the task [49, 50]. Much ground truth inference algorithms have been used for aggregating multiple workers' responses to infer the ground truth of the task [8, 9]. The ground truth, as a crowdsourcing result, is fed back to the requester. According to the calculation models [35], inference algorithms can be divided into the noniterative algorithm and iterative algorithm [10].

5.1. Noniterative Algorithm. The noniterative algorithm infers the ground truth of the task directly from the workers'

responses [51]. Majority Voting (MV) [3], a simple method, takes a response that is consistent with the majority of the workers' responses as the ground truth. If multiple responses have the same maximum number of votes, one of them is randomly selected as the ground truth. For example, given a binary task t_i , the label option $x_i \in \{0, 1\}$, N workers label the task t_i , and the response of the worker w_j is represented by $y_i^j = w_j(x_i) \in \{0, 1\}$. The ground truth of the task t_i is \hat{y}_i .

$$\hat{y}_i = \begin{cases} 1, & \frac{1}{N} \sum_{j=1}^N y_i^j > \frac{1}{2}, \\ \text{random guess}, & \frac{1}{N} \sum_{j=1}^N y_i^j = \frac{1}{2}, \\ 0, & \frac{1}{N} \sum_{j=1}^N y_i^j < \frac{1}{2}. \end{cases} \quad (1)$$

MV, if (1) more than half of the workers voted unanimously and (2) the error rate of workers is uniformly distributed, can effectively improve the accuracy of the ground truth [49].

The typical MV method is only suitable for the discrete decision task. Mean and median are generally regarded as the truth of a numerical task [8]. These methods are simple to calculate and easy to implement in applications. However, if spammers are in the majority, the ground truth may seriously deviate from the real answer of the task [52]. In addition, these methods assumed there was no difference in response quality among all workers.

HP (HoneyPot) [51], an advanced version of MV, is proposed. It first filters out low-quality workers based on the gold standard, then adopts MV to infer the truth based on the remainder of workers. Unlike the HP algorithm, ELICE (Expert Label Injected Crowd Estimation) [53] assumes that a worker's response is related to the difficulty of the task. Using labels provided by experts as the gold standard, the ratio of the number of workers who responded correctly to the total number of workers who participated in the gold standard is used to measure the difficulty of the task. ELICE and HP solved the problem of excessive spammers, but the

accuracy of the aggregation results depends heavily on the rationality of the gold standard and threshold setting.

5.2. Iterative Algorithm. The iterative algorithm iterates in two steps until the algorithm converges. Each iteration is divided into two steps: (1) update the aggregation truth of the task; (2) update the quality of workers. There are several iteration algorithms such as EM (Expectation Maximization), SLME (Supervised Learning from Multiple Experts), GLAD (Generative Model of Labels, Abilities, and Difficulties), FaitCrowd [54], TEST (Topic-missile-similar Tasks) [31], and ZenCrowd [55].

EM [56] carries out Maximum Likelihood Estimation (MLE) through iteration of the E (Expectation) step and the M (Maximization) step.

- (i) *E step*: infer the truth of a task based on the worker quality and labels provided by workers.
- (ii) *M step*: the worker quality based on the truth inferred in the Expectation step and labels provided by workers.

When the algorithm is stopped, the inferred ground truths of tasks and the confusion matrix representing the error rate of the worker's response are returned.

EM algorithm improves the accuracy of task aggregation results since it considers the variation of worker quality. However, EM has a high computational cost and long-running time because of its iterative running characteristics, and the clustering results are closely related to the initial values of the parameters. Moreover, for a large number of label categories and a small number of labels, the confusion matrix obtained is a sparse matrix, which means that the inaccuracy of the estimated results is very high.

Similar to EM algorithm, SLME algorithm [51] also obtains aggregation results through the alternating calculation of the E step and the M step. It is assumed that the quality of workers is proportional to their professional ability, and the sensitivity and specificity of statistics are used to measure the professional ability of workers. Therefore, SLME algorithm is only applicable to binary-class tasks.

GLAD [57] takes extra consideration of the difficulty of the task based on EM algorithm and assumes the adversarial labeler can be reversed. Each iteration updates the aggregation label of the task, the professional ability of the worker, and the difficulty of the task. GLAD outperforms the commonly used "Majority Vote" heuristic for inferring image labels and is robust to both noisy and adversarial labelers.

FaitCrowd [54] uses the topic model LDA to model the professional ability of workers in different topics and assumes that a task belongs to only one topic. Different from FaitCrowd, TEST [31] assumes that a task may belong to multiple topics.

The above inference algorithms attempt to model the worker quality from multiple sides, including the expertise of a worker and the difficulty of a task. However, due to the sparsity of samples, the accuracy of the inferences is subject to certain risks. Demartini et al. [55] argued that a simple model would perform better than a complex model on a

sparse dataset and therefore proposed the ZenCrowd algorithm [55]. Because the ZenCrowd algorithm uses fewer parameters, it avoids the problem of large deviation of variable estimation in the case of sparse data. Since the ZenCrowd algorithm uses maximum entropy to estimate the quality of a worker, its advantage is that it is suitable for a multiclass task that has more than two options.

Table 3 shows the main inference algorithms used in response aggregation. The factors considered by the algorithm, the suitable task type, and the efficiency are compared. Overall, the running time of the noniterative algorithm is lower than that of the iterative algorithm, but the accuracy of the aggregation result is related to the specific dataset.

6. Incentive Mechanism Design

Despite some workers being willing to work for free, most crowdsourcing workers want to be paid for their services. Hiring one user is cheap, but incentivizing extensive, reliable users to perform tasks is still crucial under a limited budget. Several studies have identified direct relations between incentives and workers' response quality and/or task execution speed [35, 58, 59]. Incentives may come in two different forms: extrinsic incentives (e.g., monetary [60] and virtual currency [61]) and intrinsic incentives (e.g., gamification point/leaderboards [19, 20]). Extrinsic incentives accelerate task execution speed [58]. Intrinsic incentives influence quality more significantly than extrinsic ones [37]. Task design typically combines extrinsic incentives and intrinsic incentives, to attract enough workers and ensure the quality of the response.

The goal of rational workers motivated by extrinsic incentives is to maximize the payoffs. A worker's payoff is the difference between the reward received by the worker and the cost incurred to complete the task. Maximizing payoffs implies minimizing the cost (e.g., effort to respond to tasks), which generally leads to the poor quality of responses. Much research on incentive mechanisms have been conducted, to design a payment rule trade-off between the number of tasks completed, the response's quality, and the payment paid by requesters (or the reward received by workers). Existing crowdsourcing incentive mechanisms can be divided into two categories: non-game theory-based incentive mechanisms and game theory-based incentive mechanisms.

6.1. Non-Game Theory-Based Incentive Mechanisms. Few studies on non-game theory-based incentive mechanisms are proposed. The existing incentive mechanisms are mainly designed from the perspective of the task requester. Reference [62] proposes a payment mechanism that takes a multiplicative form, where the worker's response to a golden question is evaluated using a score; the reward received by the worker is the sum of the minimum payment and bonus, which is the product of the score and unit bonus per task. This score is directly related to the quality of the workers' response; workers with lower response quality are given a lower score and hence paid less and vice versa. Thus, this

TABLE 3: Ground truth inference algorithms.

| Algorithms | Worker quality model | Task model | Task type | Efficiency | |
|------------------------|----------------------|--|---------------------|--------------|-----|
| Noniterative algorithm | MV [3] | × | Binary-class | High | |
| | HP [51] | The probability of correctly responds to the gold standard | × | | |
| | ELICE [53] | The probability of correctly responding to the gold standard | The task difficulty | | |
| Iterative algorithm | EM [56] | Confusion matrix | × | Binary-class | |
| | SLME [134] | Sensitivity, specificity | × | Binary-class | |
| | GLAD [57] | The probability of responding correctly | The task difficulty | Binary-class | Low |
| | FaitCrowd [54] | LDA | LDA | Binary-class | |
| | TEST [31] | LDA | LDA | Binary-class | |
| | ZenCrowd [55] | The probability of responding correctly | × | Multiclass | |

payment mechanism, on the one hand, prevents spammers from participating in the task and, on the other hand, encourages high-quality workers to actively participate in the task and ultimately achieves the goal of improving the quality of the response received by the requester.

Reference [63] examines the problem: if the budget is not enough to support one response per task, is it to motivate more tasks to be completed or to motivate better quality acquisition of a single response? Requallo, a flexible budget allocation framework, is proposed based on the Markov decision process, to determine the number of annotation instances and payments, ultimately maximizing the number of annotation instances under a limited budget, while ensuring quality does not fall below a certain threshold.

The aforementioned non-game theory-based approach designs the incentive mechanism directly based on the task requestor’s estimate of the worker’s quality. The worker either accepts or rejects the task, and there is no negotiation with the task requestor over the quality or price of the task. To solve this problem, game theory was introduced into incentive mechanisms, and hence, a large number of incentive mechanisms based on game theory have been proposed.

6.2. Game Theory-Based Incentive Mechanisms. During the crowdsourcing process, the behavior and interests of task requesters and workers interact and constrain. Based on this, researchers have proposed a large number of incentive mechanisms based on game theory to solve the utility maximization problem of the parties involved. In economics, utility refers to the degree of satisfaction people get from a good or a service.

We will focus on two types of game theory-based incentives: auction-based incentives and Stackelberg-based incentives.

6.2.1. Auction-Based Incentive Mechanism. The auction-based incentive mechanism models the interaction between stakeholders as an auction process and examines the properties of auctions under the behavior of the stakeholders. An auction-based incentive mechanism is generally evaluated according to the following desirable auction properties.

- (1) *Individually Rational (IR)*. The utility of all participants is nonnegative.
- (2) *Budget Feasible (BF)*. The payment paid by the requestor/platform must be less than or equal to his budget.
- (3) *Compute Efficiently (CE)*. The computational complexity of the incentive mechanism algorithm is polynomial time.
- (4) *Truthful (T)*. During the game, players will not provide false personal information (such as the cost of participating in the task or the value they can bring to the other party) or manipulate strategically to gain more utility. In other words, the players maximize their utility only if they have truly reported their personal information.

The first three properties ensure the feasibility of the incentive mechanism. The fourth property eliminates the fear of market manipulation among participating users.

Reference [64] designs the incentive mechanism, MSensing auction, based on the antiauction model to determine the optimal time for users to participate in the task, so as to maximize users’ utility. MSensing auction is proved to be profitable (i.e., the platform should not incur a deficit.) instead of the property BF. Literature [65] models the interaction between workers and task requestors as an antiauction model and proposes an incentive mechanism under budget constraints. Initially, the workers submit bids, each of which is a task-price pair, to the platform. The platform then greedily selects bidders to maximize its utility and determines how much to pay. Bayesian inference is used to estimate worker quality and selection, and Myerson’s lemma is used to determine the reward to be paid to the winner. Literature [66] considers the real-time arrival and departure of workers and proposes dynamic incentive mechanisms OMZ and OMG based on the online auction model. In the current period, workers first bid with the reserve price of accepting the task and the time of arrival and departure. The platform then decides whether to accept the service of the worker and the reward to be paid, under the remaining budget, so as to

TABLE 4: An auction-based incentive mechanism.

| Reference | Quality | IR | BF | CE | T | Other properties | Goal | Scenario |
|-----------------------|---------|----|----|----|---|--|--|----------|
| MSensing auction [64] | × | √ | × | √ | √ | Profitability | Maximize the user's utility | Offline |
| [65] | √ | √ | √ | √ | √ | No | Maximize platform's utility | Offline |
| [66] | × | √ | √ | √ | √ | Consumer sovereignty, constant competitiveness | Maximize platform's utility | Online |
| TM [67] | × | √ | × | √ | √ | Frugality | Minimize payment | Offline |
| [68] | √ | √ | √ | √ | √ | Competitiveness | Maximize the total number of satisfied tasks | Online |

maximize its utility. In addition to the four properties mentioned above, the incentive mechanisms also have the properties of *consumer sovereignty*, which guarantees that each participating user has a chance to win the auction, and *constant competitiveness*, which ensures that the mechanism has an approximate optimal solution in an offline scenario. Unlike the above work, literature [67] proposes a frugal auction-based mechanism (i.e., a nonuniform truthful mechanism, TM), which is committed to saving payment paid by the requester. Reference [68] considers that the worker quality may change over time and proposes a long-term, dynamic, quality-sensitive incentive mechanism, Melody, which models the interaction between requestors and workers as a reverse-auction model running continuously. Melody is proven to satisfy the competition property, which means that the ratio of this mechanism's solution to the optimal solution (OPT) that is computed in the offline case is $O(1)$.

Table 4 compares auction-based incentive mechanisms.

6.2.2. Stackelberg-Based Incentive Mechanism. A Stackelberg game is used to model the competition between one player, called the *leader*, and a set of players, called the followers. A Stackelberg game consists of two stages: the leader first takes actions and knows the actions will be observed by the followers. The followers then take actions according to the actions observed. Both parties choose their own strategies according to the strategies of the other party, so as to maximize their utility under the strategies of the other party and, hence, achieve Stackelberg Equilibrium.

Literature [64] designed a truthful incentive mechanism based on the Stackelberg game, where the platform is a leader and each worker is a follower. First, the platform announces the total reward of the task. Next, each worker decides sensing time to maximize its utility. By solving the Stackelberg equilibrium of the utility function of the workers and the platform, the optimal strategy (i.e., the reward or the time) of both parties is obtained when the utility is maximum.

Reference [67] resolved the problem of minimizing payment in a scenario where both the cost of workers and the value they contribute are heterogeneous. CS-MECH, an incentive mechanism based on the Stackelberg game, was proposed. In the first stage, the requester, as a leader, announces the total payment that will be allocated to all participants. In the second stage, the workers, as followers, learn the task and other workers' information then decide their

participation level which maximizes their utility. Reference [67] compares the two incentive mechanisms proposed: CS-MECH and TM, and finds that CS-MECH performs better than TM in terms of reducing the payment.

Reference [69], highlighting the collaboration between requestors and workers, proposes a novel framework, in which workers and requestors observe each other's strategies and share their information to maximize their benefit. First, the worker, as the leader, reports the optimal strategy maximizing its utility. The worker's strategy is the number of tasks it plans to complete. Next, each requester, as a follower, identifies the optimal strategy (i.e., the unit price of the task) based on the observed strategy and the private information owned (i.e., the worker's reputation) and then shares the information with the workers. Finally, based on the information observed, the worker adjusts the number of tasks to be accomplished, to maximize its utility.

Besides, privacy and security are the challenges of crowdsourcing. Given concerns about privacy disclosure and security threats, workers may be reluctant to participate in tasks [70]. Li and Cao [71] proposed two privacy-oriented incentive mechanisms, in which users are encouraged by credits to upload data without being disclosed. One scheme is implemented by a trusted third-party platform using a hashing verification equation; the other scheme is implemented by blind signature and delegate technology. Xiong et al. [72] have proposed a secure framework for reward-based spatial crowdsourcing (SECRSC), which uses homomorphism encryption technology to prevent the disclosure of information uploaded by workers.

7. Privacy-Preserving

Crowdsourced data collection faces three types of privacy threats:

- (1) *Threat to Data Privacy of Workers.* The data collected may relate to the privacy of workers, such as the social activities, travel trajectory, political views, and health. The disclosure of such information probably harms workers' privacy.
- (2) *Threats to Personal Information Privacy of Workers.* The personal information of the worker is uploaded by the worker when he/she registers in the crowdsourcing platform, including the worker's ID and

gender. Personal information can be stolen and used to commit crimes.

- (3) *Threats to Task Privacy.* The task uploaded by the requester includes the information about the requester. An attacker may infer valuable requester information from the task description, thus endangering the requester's privacy.

To protect crowdsourcing from privacy security threats, several methods have been applied to task allocation, data aggregation, and incentive mechanisms.

- (1) *Task Assignment with Privacy.* To et al. [73] introduced a trusted third party to protect worker location privacy based on differential privacy. Shen [74] designed a secure task assignment protocol using additively homomorphic encryption with the introduction of a semihonest third party. In contrast, [75–78] intend to protect both task privacy and worker privacy. References [76, 78] proposed the task assignment based on the encrypted locations of workers and requesters by homomorphic encryption. In [75, 77], a dual privacy-preserving algorithm based on anonymity is proposed in task matching in spatial crowdsourcing.
- (2) *Data Aggregation with Privacy.* In the data aggregation stage, there are two common methods to protect privacy: homomorphic encryption [79, 80] and the addition of random noise perturbation data [81]. In [79], a data aggregation scheme based on additional homomorphic identity encryption (IBE) was proposed, in which data reported to SP should be encrypted using the worker's private key. This ensures that workers' data is not decrypted (except by the trusted third party). Zhuo et al. [80] proposed a data aggregation scheme that supports privacy protection and data integrity. Zhuo et al. [80], based on Brakerski–Gentry–Vaikuntanathan, proposed a verifiable homomorphic encryption scheme.
- (3) *Incentive Mechanism with Privacy.* To protect personal information privacy, some incentive schemes try to support differential privacy by adding random disturbance to bidding information [82–84], which can protect workers' personal information well. Meanwhile, It also ensures that no worker can gain more benefit by claiming false bids. In addition, [85] applied homomorphic cryptography to protect the privacy of personal information (bidding) and considered the verification of incentive results. Sun and Ma [86] proposed a verifiable incentive mechanism for privacy protection based on signature and homomorphic encryption.

8. Discussion

In this section, we discuss some important open problems in crowdsourcing data collection.

8.1. More Effective Incentive Mechanism. Crowdsourcing data collection tasks involve three entities, including task requesters, workers, and crowdsourcing platforms. One of the most fundamental questions is how to recruit extensive appropriate workers. Existing research focuses on how to design tasks to attract enough reliable workers to participate in tasks. One of the most important components of the task design is the incentive mechanism. However, existing work mainly designs the incentive mechanism from the requester's perspective, in which the reward paid to workers is mainly based on the task requestor's evaluation of worker responses; the drawback is that the evaluation may not be accurate, because the task requestor may be malicious and deceptive. How to design an incentive mechanism from the perspective of the system, with constraints on both workers and requestors, is worth further study.

8.2. More Accurate Selection of Workers. Worker quality is the key basis of task assignment. Some workers exclude low-quality workers through the gold standard. However, the gold standard hidden in a crowdsourced task can lead to an increase in the cost (e.g., the payment by the requester and the time to complete tasks) of the task. Besides, if the gold standard is too difficult or not relevant to the real task, honest, professional workers may be eliminated, thus wasting the requester's resources. Therefore, further research is needed to ensure the elimination of spammers during task assignments and to ensure the accuracy of response aggregation.

8.3. Privacy-Preserving. Privacy is an important issue in crowdsourcing. The data collected through crowdsourcing may contain a large amount of sensitive information, which is directly related to user privacy, such as the user's geographical location, travel trajectory, and personal preferences. This would cause serious security threats, although some studies have incorporated privacy-preserving techniques into task assignment [76, 77], response aggregation [87], and incentive mechanisms [83, 85]. However, in crowdsourcing, malicious participants or the platform may deceive other stakeholders. Hence, the privacy threat has the unique characteristic of deceptive practices [88]. Further, developing effective strategies for protecting user privacy remains an open research problem in crowdsourcing.

8.4. Practical Application. Finally, researches on crowdsourcing data collection have been mainly carried out theoretically or verified on the prototype system of researchers. Applying theoretical research to the real world is an aspect of the development of crowdsourcing platforms that remains to be explored.

9. Conclusion

This paper summarizes the key issues faced in the process of crowdsourcing data collection, reviews relevant technologies proposed over the past decade, and discusses the similarities and differences between these technologies. Then, the present situation of crowdsourcing research and the problems that can be further studied are discussed. Finally,

we hope that our work can provide a reference for relevant researchers.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was funded by the Guangxi Key Laboratory of Trusted Software (No. kx201727), Project to Improve the Scientific Research Basic Ability of Middle-Aged and Young Teachers (No. 2019KY0226), Natural Science Foundation of China (Nos. 62066010, U1811264, and U1711263), and Natural Science Foundation of Guangxi Province (Nos. 2019GXNSFB245049, 2019GXNSFB245059, 2018GXNSFDA281045, and 2020GXNSFAA159055).

References

- [1] M. Sahlins, "The conflicts of the faculty," *Wired Magazine*, vol. 35, no. 4, pp. 997–1017, 2009.
- [2] E. Peer, J. Vosgerau, and A. Acquisti, "Reputation as a sufficient condition for data quality on Amazon Mechanical Turk," *Behavior Research Methods*, vol. 46, no. 4, pp. 1023–1031, 2014.
- [3] W. Wang and Z.-H. Zhou, "Crowdsourcing label quality: a theoretical analysis," *Science China Information Sciences*, vol. 58, no. 11, pp. 1–12, 2015.
- [4] P. G. Ipeirotis, F. Provost, and J. Wang, "Quality management on Amazon Mechanical Turk," in *HCOMP '10: Proceedings of the ACM SIGKDD Workshop on Human Computation*, pp. 64–67, Washington, D.C., USA, 2010.
- [5] D. R. Karger, S. Oh, and D. Shah, "Efficient crowdsourcing for multi-class labeling," *ACM SIGMETRICS Performance Evaluation Review*, vol. 41, no. 1, pp. 81–92, 2013.
- [6] D. R. Karger, S. Oh, and D. Shah, "Iterative learning for reliable crowdsourcing systems," in *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011*, pp. 1953–1961, Granada, Spain, 2011.
- [7] D. R. Karger, S. Oh, and D. Shah, "Budget-optimal crowdsourcing using low-rank matrix approximations," in *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 284–291, Monticello, IL, USA, 2011.
- [8] Y. Zheng, G. Li, Y. Li, C. Shan, and R. Cheng, "Truth inference in crowdsourcing," *Proceedings of the VLDB Endowment*, vol. 10, no. 5, pp. 541–552, 2017.
- [9] J. Zhang, X. Wu, and V. S. Sheng, "Learning from crowd-sourced labeled data: a survey," *Artificial Intelligence Review*, vol. 46, no. 4, pp. 543–576, 2016.
- [10] N. Q. V. Hung, N. T. Tam, L. N. Tran, and K. Aberer, "An evaluation of aggregation techniques in crowdsourcing," in *Web Information Systems Engineering – WISE 2013*, vol. 8181 of Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp. 1–15, Springer, 2013.
- [11] J. Chandler, G. Paolacci, and P. Mueller, "Risks and rewards of crowdsourcing marketplaces," in *Handbook of Human Computation*, pp. 377–392, Springer, 2013.
- [12] M.-h. Wu and A. J. Quinn, "Confusing the crowd : task instruction quality on Amazon Mechanical Turk," in *InProceedings of the AAAI Conference on Human Computation and Crowdsourcing*, Quebec, Canada, 2017.
- [13] L. C. Irani and M. Six Silberman, "Turkopticon: interrupting worker invisibility in Amazon Mechanical Turk," in *CHI '13: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Paris, France, 2013.
- [14] A. Kulkarni, P. Gutheim, P. Narula, D. Rolnitzky, T. Parikh, and B. Hartmann, "Mobileworks: designing for quality in a managed crowdsourcing architecture," *IEEE Internet Computing*, vol. 16, no. 5, pp. 28–35, 2012.
- [15] U. Gadiraju, J. Yang, and A. Bozzon, "Clarity is a worthwhile quality-on the role of task clarity in microtask crowdsourcing," in *HT 2017- Proceedings of the 28th ACM Conference on Hypertext and Social Media*, pp. 5–14, New York, NY, USA, 2017.
- [16] T. Gillier, C. Chaffois, M. Belkhouja, Y. Roth, and B. L. Bayus, "The effects of task instructions in crowdsourcing innovative ideas," *Technological Forecasting and Social Change*, vol. 134, pp. 35–44, 2018.
- [17] K. Wang, J. Nickerson, and Y. Sakamoto, "Crowdsourced idea generation: the effect of exposure to an original idea," *Creativity and Innovation Management*, vol. 27, no. 2, pp. 196–208, 2018.
- [18] P. G. Ipeirotis, "Analyzing the Amazon Mechanical Turk marketplace," *XRDS: Crossroads, The ACM Magazine for Students*, vol. 17, no. 2, pp. 16–21, 2010.
- [19] B. Morschheuser, "Gamification in crowdsourcing : a review," in *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pp. 4375–4384, Koloa, HI, USA, 2016.
- [20] L. von Ahn, "Games with a purpose," *Computer*, vol. 39, no. 6, pp. 92–94, 2006.
- [21] M. S. Bernstein, G. Little, R. C. Miller et al., "Soylent: a word processor with a crowd inside," in *UIST '10: Proceedings of the 23rd annual ACM symposium on User interface software and technology*, pp. 313–322, New York, 2010.
- [22] A. Kittur, B. Smus, S. Khamkar, and R. E. Kraut, "CrowdForge: crowdsourcing complex work," in *UIST'11- Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, pp. 43–52, Santa Barbara, CA, 2011.
- [23] A. Kulkarni, M. Can, and B. Hartmann, "Collaboratively crowdsourcing workflows with Turkomatic," in *CSCW '12: Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, pp. 1003–1012, Seattle, Washington, USA, 2012.
- [24] J. Dean and S. Ghemawat, "MapReduce," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.
- [25] F. Daniel, P. Kucherbaev, C. Cappiello, B. Benatallah, and M. Allahbakhsh, "Quality control in crowdsourcing," *ACM Computing Surveys*, vol. 51, no. 1, pp. 1–40, 2018.
- [26] M.-C. Yuen, I. King, and K.-S. Leung, "TaskRec: probabilistic matrix factorization in task recommendation in crowdsourcing systems," in *Neural Information Processing*, vol. 7664 of Lecture Notes in Computer Science, pp. 516–525, 2012.
- [27] D. Geiger and M. Schader, "Personalized task recommendation in crowdsourcing information systems – current state of the art," *Decision Support Systems*, vol. 65, pp. 3–16, 2014.
- [28] V. Ambati, S. Vogel, and J. Carbonell, "Towards task recommendation in micro-task markets," in *Proceedings of the 25th*

- AAAI Workshop in Human Computation, pp. 80–83, San Francisco, USA, 2011.
- [29] Y. Gong, L. Wei, Y. Guo, C. Zhang, and Y. Fang, “Optimal task recommendation for mobile crowdsourcing with privacy control,” *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 745–756, 2016.
- [30] M. Safran and D. Che, “Real-time recommendation algorithms for crowdsourcing systems,” *Applied Computing and Informatics*, vol. 13, no. 1, pp. 47–56, 2017.
- [31] S. S. Gong, W. Hu, W. Y. Ge, and Y. Z. Qu, “Modeling topic-based human expertise for crowd entity resolution,” *Journal of Computer Science and Technology*, vol. 33, no. 6, pp. 1204–1218, 2018.
- [32] S. Basu Roy, I. Lykourantzou, S. Thirumuruganathan, S. Amer-Yahia, and G. Das, “Task assignment optimization in knowledge-intensive crowdsourcing,” *VLDB Journal*, vol. 24, no. 4, pp. 467–491, 2015.
- [33] G. Kazai, J. Kamps, and N. Milic-Frayling, “An analysis of human factors and label accuracy in crowdsourcing relevance judgments,” *Information Retrieval*, vol. 16, no. 2, pp. 138–178, 2013.
- [34] W. Wang, X.-Y. Guo, S.-Y. Li, Y. Jiang, and Z.-H. Zhou, “Obtaining high quality label by distinguishing between easy and hard items in crowdsourcing,” in *IJCAI’17 Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 2964–2970, Melbourne, Australia, 2017.
- [35] A. I. Chittilappilly, L. Chen, and S. Amer-Yahia, “A survey of general-purpose crowdsourcing techniques,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 9, pp. 2246–2266, 2016.
- [36] H. Xie, J. C. S. Lui, and D. Towsley, “Incentive and reputation mechanisms for online crowdsourcing systems,” in *In Proceeding 2015 IEEE 23rd International Symposium on Quality of Service, IWQoS 2015*, pp. 207–212, Portland, OR, USA, 2016.
- [37] M. Allahbakhsh, B. Benatallah, A. Ignjatovic, H. R. Motahari-Nezhad, E. Bertino, and S. Dustdar, “Quality control in crowdsourcing systems: issues and directions,” *IEEE Internet Computing*, vol. 17, no. 2, pp. 76–81, 2013.
- [38] H. Xie and J. C. S. Lui, “Incentive mechanism and rating system design for crowdsourcing systems: analysis, tradeoffs and inference,” *IEEE Transactions on Services Computing*, vol. 11, no. 1, pp. 90–102, 2018.
- [39] A. Carlson, J. Betteridge, R. C. Wang, E. R. Hruschka, and T. M. Mitchell, “Coupled semi-supervised learning for information extraction,” in *In WSDM 2010- Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, pp. 101–110, New York, USA, 2010.
- [40] C. Akkaya, A. Conrad, J. Wiebe, and R. Mihalcea, “Amazon Mechanical Turk for subjectivity word sense disambiguation,” in *In Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk, CSLDAMT’10*, pp. 195–203, Morristown, NJ, USA, 2010.
- [41] D. J. Hauser and N. Schwarz, “Attentive Turkers: MTurk participants perform better on online attention checks than do subject pool participants,” *Behavior Research Methods*, vol. 48, no. 1, pp. 400–407, 2016.
- [42] Y. S. Lee, Y. W. Seo, and E. Siemsen, “Running behavioral operations experiments using Amazons Mechanical Turk,” *Production and Operations Management*, vol. 27, no. 5, pp. 973–989, 2018.
- [43] C. J. Ho, S. Jabbari, and J. W. Vaughan, “Adaptive task assignment for crowdsourced classification,” in *In: Proceedings of the 30th International Conference on Machine Learning, ICML, 2013*.
- [44] J. Fan, G. Li, B. C. Ooi, K.-I. Tan, and J. Feng, “ICrowd: an adaptive crowdsourcing framework,” in *In Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pp. 1015–1030, SIGMOD’15. New York, NY, USA: ACM, 2015.
- [45] S. Castano, A. Ferrara, and S. Montanelli, “Crowdsourcing task assignment with online profile learning,” in *In Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pp. 226–242, Valletta, Malta, 2018.
- [46] P. Mavridis, D. Gross-Amblard, and Z. Miklós, “Using hierarchical skills for optimized task assignment in knowledge-intensive crowdsourcing,” in *In 25th International World Wide Web Conference, WWW 2016, Montréal Québec Canada, 2016*.
- [47] N. Kaufmann, T. Schulze, and D. Veit, “More than fun and money. Worker motivation in crowdsourcing – a study on Mechanical Turk,” in *Proceedings of the Seventeenth Americas Conference on Information Systems*, pp. 1–11, Detroit, Michigan, 2011.
- [48] M. Alsayasneh, S. Amer-Yahia, E. Gaussier et al., “Personalized and diverse task composition in crowdsourcing,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 1, pp. 128–141, 2018.
- [49] S. Alexander and F. David, “Utility data annotation with amazon mechanical turk,” in *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Anchorage, AK, USA, 2008.
- [50] R. Snow, B. O’Connor, D. Jurafsky, and A. Y. Ng, “Cheap and fast - but is it good? Evaluating non-expert annotations for natural language tasks,” in *In Proceedings of the 2008 Conference on Empirical Methods on Natural Language Processing*, Honolulu, Hawaii, 2008.
- [51] L. Nassar and F. Karray, “Overview of the crowdsourcing process,” *Knowledge and Information Systems*, vol. 60, no. 1, pp. 1–24, 2019.
- [52] P. G. Ipeirotis, F. Provost, V. S. Sheng, and J. Wang, “Repeated labeling using multiple noisy labelers,” *Data Mining and Knowledge Discovery*, vol. 28, no. 2, pp. 402–441, 2014.
- [53] F. Khan-Khattak and A. Salleb-Aouissi, *Quality Control of Crowd Labeling through Expert Evaluation*, Workshop Comput. Social Sci, Wisdom Crowds, 2011.
- [54] F. Ma, Y. Li, Q. Li et al., “Faitcrowd: fine grained truth discovery for crowdsourced data aggregation,” in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2015*, pp. 745–754, Sydney, Australia, 2015.
- [55] G. Demartini, D. E. Difallah, and P. Cudré-Mauroux, “ZenCrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking,” in *In: proceedings of the 21st international conference on world wide web, ACM*, pp. 469–478, Lyon, France, 2012.
- [56] A. P. Dawid and A. M. Skene, “Maximum likelihood estimation of observer error-rates using the em algorithm,” *Applied Statistics*, vol. 28, no. 1, pp. 20–28, 1979.
- [57] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, and J. Movellan, *Whose vote should count more: optimal integration of labels*

- from labelers of unknown expertise, *Advances in Neural Information Processing Systems*, 2009.
- [58] M. Buhrmester, T. Kwang, and S. D. Gosling, "Amazon's Mechanical Turk," *Perspectives on Psychological Science*, vol. 6, no. 1, pp. 3–5, 2011.
- [59] Y. Singer and M. Mittal, "Pricing mechanisms for crowdsourcing markets," in *In Proceedings of the 22Nd International Conference on World Wide Web, 1157–66. WWW'13*, New York, NY, USA: ACM, 2013.
- [60] I. Krontiris and A. Albers, "Monetary incentives in participatory sensing using multi-attributive auctions," *International Journal of Parallel, Emergent and Distributed Systems*, vol. 27, no. 4, pp. 317–336, 2012.
- [61] J. S. Lee and B. Hoh, "Dynamic pricing incentive for participatory sensing," *Pervasive and Mobile Computing*, vol. 6, no. 6, pp. 693–708, 2010.
- [62] N. B. Shah and D. Zhou, "Double or nothing: multiplicative incentive mechanisms for crowdsourcing," *Journal of Machine Learning Research*, vol. 17, no. 1, pp. 5725–5776, 2016, <http://dl.acm.org/citation.cfm?id=2946645.3053447>.
- [63] Q. Li, F. Ma, L. S. Jing Gao, and C. J. Quinn, "Crowdsourcing high quality labels with a tight budget," in *In WSDM 2016-proceedings of the 9th ACM international conference on web search and data mining*, San Francisco, California, USA, 2016.
- [64] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing," in *In proceedings of the annual international conference on mobile computing and networking*, MOBICOM, 2012.
- [65] Q. Zhang, Y. Wen, X. Tian, X. Gan, and X. Wang, "Incentivize crowd labeling under budget constraint," in *In proceedings-IEEE INFOCOM*, Hong Kong, China, 2015.
- [66] Z. Dong, X.-y. Li, and H. Ma, "How to crowdsource tasks truthfully without sacrificing utility : online incentive mechanisms with budget constraint," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, pp. 1213–1221, Toronto, Canada, 2014.
- [67] W. Wu, W. Wang, M. Li, S. Member, and J. Wang, "Incentive mechanism design to meet task criteria in crowdsourcing: how to determine your budget," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 2, pp. 502–516, 2017.
- [68] H. Wang, S. Guo, J. Cao, and M. Guo, "MELODY: a long-term dynamic quality-aware incentive mechanism for crowdsourcing," in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, pp. 933–943, Atlanta, GA, USA, 2017.
- [69] X. Wu, S. Wang, C. Liu, W. Sun, and C. Wang, "Stackelberg game based tasks assignment mechanism using reputation in crowdsourcing," in *In proceedings -2016 international conference on identification, information and knowledge in the internet of things, IIKI 2016*, pp. 332–339, Beijing, China, 2018.
- [70] W. Feng, Y. Zheng, H. Zhang, K. Zeng, X. Yu, and Y. T. Hou, "A survey on security, privacy, and trust in mobile crowdsourcing," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2971–2992, 2018.
- [71] Q. Li and G. Cao, "Providing efficient privacy-aware incentives for mobile sensing," in *In proceedings-international conference on distributed computing systems*, Madrid, Spain, 2014.
- [72] P. Xiong, D. Zhu, L. Zhang, W. Ren, and T. Zhu, "Optimizing rewards allocation for privacy-preserving spatial crowdsourcing," *Computer Communications*, vol. 146, pp. 85–94, 2019.
- [73] To, Hien, G. Ghinita, and C. Shahabi, "A framework for protecting worker location privacy in spatial crowdsourcing," *Proceedings of the VLDB Endowment*, vol. 7, no. 10, pp. 919–930, 2014.
- [74] Y. Shen, "Towards preserving worker location privacy in spatial crowdsourcing," in *in Proc. IEEE GLOBECOM*, pp. 1–6, San Diego, CA, USA, 2015.
- [75] J. Shu, X. Liu, X. Jia, K. Yang, and R. H. Deng, "Anonymous privacy-preserving task matching in crowdsourcing," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 3068–3078, 2018.
- [76] S. Han, J. Lin, S. Zhao et al., "Location privacy-preserving distance computation for spatial crowdsourcing," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7550–7563, 2020.
- [77] S. Wang, X. J. Xiang, and Q. Sang, "A dual privacy preserving algorithm in spatial crowdsourcing," *Mobile Information Systems*, vol. 2020, 6 pages, 2020.
- [78] A. Liu, W. Wang, S. Shang, Q. Li, and X. Zhang, "Efficient task assignment in spatial crowdsourcing with worker and task privacy protection," *GeoInformatica*, vol. 22, no. 2, pp. 335–362, 2018.
- [79] F. Günther, M. Manulis, and A. Peter, "Privacy-enhanced participatory sensing with collusion resistance and data aggregation," in *in Proc. Conf. Cryptol. Netw. Security (CANS)*, pp. 321–336, Heraklion, Greece, 2014.
- [80] G. Zhuo, Q. Jia, L. Guo, M. Li, and P. Li, "Privacy preserving verifiable data aggregation and analysis for cloud assisted mobile crowdsourcing," in *in Proc. Annu. IEEE Conf. Comput. Commun. (INFOCOM)*, pp. 1–9, San Francisco, CA, USA, 2016.
- [81] J. Chen, H. Ma, and D. Zhao, "Private data aggregation with integrity assurance and fault tolerance for mobile crowd-sensing," *Wireless Networks*, vol. 23, no. 1, pp. 131–144, 2017.
- [82] Y. Wen, J. Shi, Q. Zhang et al., "Quality-driven auction-based incentive mechanism for mobile crowd sensing," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 9, pp. 4203–4214, 2015.
- [83] H. Jin, L. Su, B. Ding, K. Nahrstedt, and N. Borisov, "Enabling privacy-preserving incentives for mobile crowd sensing systems," in *2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS)*, pp. 344–353, Nara, Japan, 2016.
- [84] T. Luo, S. K. Das, H. P. Tan, and L. Xia, "Incentive mechanism design for crowdsourcing," *ACM Transactions on Intelligent Systems and Technology*, vol. 7, no. 3, pp. 1–26, 2016.
- [85] Y. Zhang, H. Zhang, S. Tang, and S. Zhong, "Designing secure and dependable mobile sensing mechanisms with revenue guarantees," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 1, pp. 100–113, 2016.
- [86] J. Sun and H. Ma, "Privacy-preserving verifiable incentive mechanism for online crowdsourcing markets," in *in Proc. Int. Conf. Comput. Commun. Netw. (ICCCN)*, pp. 1–8, Shanghai, China, 2014.
- [87] X. Zhang, X. Chen, H. Yan, and X. Yang, "Privacy-preserving and verifiable online crowdsourcing with worker updates," *Information Sciences*, vol. 548, pp. 212–232, 2021.
- [88] H. Xia and B. McKernan, "Privacy in crowdsourcing: a review of the threats and challenges," *Computer Supported Cooperative Work: CSCW: An International Journal*, vol. 29, no. 3, pp. 263–301, 2020.