Spatial Crowdsourcing: Challenges and Opportunities

Lei Chen [#], Cyrus Shahabi [$]
[#]Dept. of Computer Science & Engineering, HKUST
Hong Kong, P.R. China
leichen@cse.ust.hk
[$]Department of Computer Science, USC
California, U.S.A.
shahabi@usc.edu

Abstract

As one of the successful forms of using Wisdom of Crowd, crowdsourcing, has been widely used for many human intrinsic tasks, such as image labeling, natural language understanding, market predication and opinion mining. Meanwhile, with advances in pervasive technology, mobile devices, such as mobile phones, tablets, and PDA, have become extremely popular. These mobile devices can work as sensors to collect various types of data, such as pictures, videos, audios and texts. Therefore, in crowdsourcing, a requester can unitize power of mobile devices and their location information to ask for data related a specific location, subsequently, the mobile users who would like to perform the task will travel to the target location and collect the data (videos, audios, or pictures), which is then sent to the requester. This type of crowdsourcing is called spatial crowdsourcing. Due to the pervasiveness of mobile devices and their superb functionality, spatial crowdsourcing is gaining more attention than general crowdsourcing platforms, such as Amazon Turk(http://www.mturk.com) and Crowdflower (http://crowdflower.com/). However, to develop a spatial crowdsourcing platform, effective and efficient solutions for motivating workers, mining workers' profiles, assigning tasks, aggregating results and controlling data quality must be developed. Therefore, in this paper, we will discuss the challenges and opportunities related to these key techniques, including 1) effective incentive mechanisms to encourage mobile device users to participate in crowdsourcing tasks; 2) automatic user profile mining methods; 3) optimal task assignment solutions; 4) novel answer aggregation models; 5) intelligent data quality control mechanisms.

1 Introduction

Recent advances in Internet and pervasive computing technology make tasks such as image tagging, language translation, and speech recognition easier to achieve by humans than by machines. Crowdsourcing, as one of the successful forms of utilizing human intelligences, has become quite popular recently. Basically, there are three components in crowdsourcing, a requester, an open call platform and workers. The requester will submit his/her tasks through the open calls and workers who are ready are assigned to perform the task. The

Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

results returned from the workers are aggregated and returned to the requester. In fact, crowdsourcing is human computation [32]and social computing [25] powered by a crowdsourced workforce. Crowdsourcing overlaps with several other topics like social computing, human computing, collective/collaborative intelligence, etc. It provides a new problem-solving paradigm [2, 38] and has branched into several areas.

Recently, smart phones have become quite popular and they have become people's lifemate. People can easily identify and participate in some location-based tasks that are close to their current positions, such as taking photos/videos, repairing houses, or preparing a gathering at a specific location. Therefore, a new framework, namely spatial crowdsourcing [23], for employing workers to conduct spatial tasks, has emerged in both academia (e.g., gMission [7] and MediaQ [24]) and industry (e.g., TaskRabbit). A typical spatial crowdsourcing platform assigns a number of moving workers to perform spatial tasks nearby, which requires workers to physically move to some specified locations and accomplish these tasks. With spatial crowdsourcing, people can well utilize the human intelligence in any places at any time.

Though many benefits can be brought through spatial crowdsourcing, building such a platform is not trival. In this paper, we will present challenges and opportunities related to the core techniques for spatial crowdsourcing along four dimensions: worker and task registration, task assignment, result aggregation, and data quality control. Specifically, we will discuss effective incentive mechanisms to motivate registered workers, automatic user profile mining tools, optimal task assignment mechanisms, market-driven data aggregation solutions, and movement pattern-based data quality control methods. Finally, we will present some privacy-related issues related to spatial crowdsourcing.

2 Overview of Spatial Crowdsouring Platform

Figure 1 shows a general overview of a spatial crowdsourcing system, such as gMission [7] and MediaQ [24]. Basically, there are four phases: 1) Registration (requesters and workers), 2) Task Assignment, 3) Answer Aggregation, and 4) Response & Quality Control. In the registration phase, the task requesters submit their spatial crowdsourcing tasks to the spatial crowdsourcing server (SCS) and workers will submit their profile information, including their locations and expertise, to the SCS and indicate their willingness to work on the tasks. The challenging issues in this phase are how to motivate workers' participation and how to adjust system configuration dynamically based on the current setting. In the task assignment phase, the SCS will assign the tasks to the workers based on the budget, location and other constraints. The task assignment is very important, and directly affects the system throughput, i.e., how many tasks can be well handled simultaneously. In this phase, we have to design solutions to address the challenges of insufficient knowledge about workers' profiles and expensive cost (NP-hard) to finding the optimal assignment. In the answer aggregation phase, the SCS will use different data integration models to aggregate the answers collected from the workers. Since the workers have different backgrounds, abilities, and different understanding of the tasks, the results returned by them may vary much. Therefore, we need to find a suitable model and design efficient solutions to aggregate the results. In addition, we need to address the challenges to handle fake results returned by malicious works, such as images or videos downloaded from Internet not captured by the worker himself/herself. In the response and quality control phase, the SCS will send the final answers to the requesters and estimate the quality of the returned results using various methods. In the same phase, the SCS will determine whether more workers should be hired to perform the same task to improve the quality. The challenging issue in this phase is that we often do not have the ground truth (gold standard) for the crowdsourcing tasks. Although there are a few previous works on spatial crowdsourcing [1, 23, 28] all of them focused only on task assignment. In the rest of this paper, we highlight the research problems and possible solutions to the challenges for each phase of the SCS.

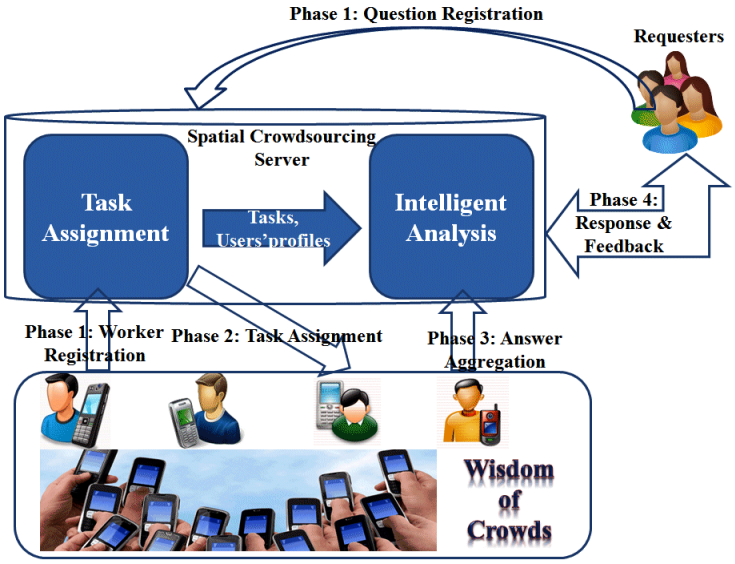

Figure 1: Overview of the Spatial Crowdsourcing System

3 Task and Worker Registration

For requesters, it is often not an issue for them to register their tasks in the SCS as long as a user friendly interface is provided. However, for normal mobile users, coming up with effective incentives to encourage their participation is quite challenging and an interesting problem, since participation often means consumption of their own resources, such as computation and communication resources. Generally, the existing incentive mechanisms can be divided into four categories: entertainment-based, service-based, social facilitation-based and monetary-based. For entertainment-based mechanisms [15, 20, 33], the system tries to gamify the tasks, such that users can contribute computation or sensing abilities of their smartphones when they play games. Here "gamify" means using game elements in non-game contexts [20]. This paradigm allows users to enjoy performing tasks, but the challenge lies in making sure that the gamified tasks are interesting enough. The service-based mechanisms are rooted in the mutual beneficial principle, where service consumers are also service providers. In other words, if a user wants to benefit from the service system, he/she also has to contribute to the service system [19, 27]. Social facilitation-based mechanisms take advantage of people's tendency to perform better in simple tasks if they are being watched. Thus to encourage people to participate in certain tasks, the effort of each participant should be easily observed by others [11, 30]. The last category is monetary incentives [18, 29, 31]. In this case, the system has to offer a certain amount of money to motivate potential participants such that the participants will use their resources, usually smartphone cameras or sensors, to complete the tasks. However, the existing monetary systems mostly assume static settings, i.e., when the interaction between a platform and users starts, there must exist a certain number of users, such that the current strategies can be applied. This setting is referred to as an offline setting. However, for most location-based services, an online setting is more practical. Therefore, it is necessary to investigate different incentive mechanisms in the online setting. Specifically, the following two issues should be further investigated:

a) Assigning different types of incentives for different tasks.

Spatial crowdsourcing tasks may involve various different tasks, ranging from simple text-based voting (decision making), image capturing, video or audio recording, to sophisticated product ranking and recommendation. Based on the characteristics of different tasks, especially the cost of performing such tasks (image or video uploading), it is necessary to design different incentives from which the requester can choose. Based on the collected log data, we can compare the similarity between the tasks in the same category and offer different

incentives. The challenge in performing such assignment is how to measure task difficulty. Only when the SCS knows the difficulty of the task can it offer suggestions regarding the type and amount of incentives that the requester should offer. Offering a small amount of incentives for a very difficult task may result in no workers taking up the task. On the contrary, giving out a large amount of incentives for a rather easy task may attract malicious workers. Therefore, how to measure the difficulty of the task is quite important. Some previous works consider the difficulty of a crowdsourcing task as a parameter in their models [26, 39]. However, it is still an open question as to how to quantify the difficulty of given crowdsourcing tasks. Even for the same class of tasks, the difficulty of individual tasks may be different. Given a set of crowdsourcing tasks, we first investigate the following measures for capturing the semantic of "difficulty" of text-based tasks.

- Entropy - If we consider the answer of a crowdsourcing task as a random variable, then each crowdsourced answer is simply the outcome of one experiment on the random variable. Under this model, difficult crowdsourcing tasks would lead to high randomness of the answer. Since entropy is a typical measure of randomness, it is a fair tool to estimate the magnitude of difficulty.

- Inter-rater reliability - Inter-rater reliability is the degree of agreement among workers. With the increase in difficulty, the degree of agreement among workers significantly decreases. Hence, the inter-rater reliability is also a reasonable measure of the difficulty. Given these two difficulty measures and their values computed from historical trace of tasks, we can estimate the difficulty of the current task by combining the difficulties of a few very similar historical tasks, The similarity function between two tasks can be a simple hamming distance or a sophisticated earth mover's distance. After completing the text-based tasks, we study how to measure the difficulty of capturing multimedia content (image, video and audio). Our initial idea is to measure the similarity between the current task and historical tasks, with the consideration of the size and resolution of the data.

b). Adjusting the online setting.

Since both tasks and workers in the system change dynamically, we need to propose solutions to address the data dynamics. Most of the workers in spatial crowdsourcing are mobile users. The characteristics of mobile devices (e.g. where a device is located) lead to the uniqueness of the spatial crowdsourcing workers. In [9], we have proposed a grid-based predication method to estimate spatial distributions of workers/tasks in the future. In particular, we consider a grid index, $\mathcal{I}$, over tasks and workers, which divides the data space $\mathcal{U}$ into γ^2 cells, each with the side length $1/\gamma$, where the selection of the best γ value can be guided by a cost model in [10]. Then, we estimate possible workers/tasks that may fall into each cell at a future timestamp, which can be inferred from historical data within the most recent sliding window of size w. After that, we assign workers with spatial tasks for multiple rounds (within a time interval P), based on the predicted location/quality distributions of workers/tasks.

Specifically, for each round at timestamp p, we first retrieve a set, T_p, of available spatial tasks, and a set, W_p, of available workers. Here, the task set T_p (or worker set W_p) contains both tasks (or workers) that have not been assigned in the last round and the ones that newly arrive at the system after the last round. In order to avoid the local optimality at the current timestamp p, we need to predict location/quality distributions of workers and tasks at a future timestamp $(p+1)$ that newly join the system, and obtain two sets W_{p+1} and T_{p+1}, respectively.

In particular, our grid-based prediction algorithm first predicts the future numbers of workers/tasks from the latest w worker/task sets in cells, and then generates possible worker (or task) samples in W_{p+1} (or T_{p+1}) for cells of the grid index $\mathcal{I}$. First, we initialize a worker set W_{p+1} and a task set T_{p+1} in the future round with empty sets. Then, within each cell $cell_i$, we can obtain its w latest worker counts, $|W_{p-w+1}^{(i)}|$, $|W_{p-w+2}^{(i)}|$, ..., and $|W_p^{(i)}|$, which form a *sliding window* of a time series (with size w). Due to the temporal correlation of worker counts in the sliding window, in this paper, we utilize *linear regression* over these w worker counts to predict the future number, $|W_{p+1}^{(i)}|$, of workers in cell, $cell_i$, that newly join the system at timestamp $(p+1)$. Note that,

other prediction methods can be also plugged into our grid-based prediction framework, and we would like to leave it as our future work. Similarly, we can estimate the number, $|T_{p+1}^{(i)}|$, of tasks in $cell_i$ for the next round at timestamp $(p+1)$. According to the predicted numbers of workers/tasks, we can uniformly generate $|W_{p+1}^{(i)}|$ worker samples (or $|T_{p+1}^{(i)}|$ task samples) within each cell $cell_i$, and add them to the predicted worker set W_{p+1} (or task set T_{p+1}). Finally, we return the two sets W_{p+1} and T_{p+1}.

Note that, random samples of workers/tasks in cells can approximately capture future location distributions of workers/tasks. However, in each cell, discrete samples may be of small sample sizes, which may lead to low prediction accuracy. Therefore, we alternatively consider continuous *probability density function* (pdf) for location distributions of workers/tasks.

4 Task Assignment

In task assignment, we need to design solutions to assign spatial crowd sourcing tasks to workers. Like traditional crowdsourcing, spatial crowdsourcing also aims to assign the tasks to workers economically and efficiently. By economically we mean that all the tasks are to be achieved within some budget, and by efficiently we mean that all the assigned tasks should be accomplished as soon as possible. To achieve these two goals, we need to address the following two key issues.

a) Building mobile users' (workers') profiles.

After the mobile users register with the SCS as workers, other than the registration data that they provide, we need to enhance their profiles by mining the tasks they have performed. This mining step is critical since the workers often do not express explicitly what type of questions (domains) they can answer during the registration. Furthermore, without feedback from the requesters, we cannot know whether the workers will perform a good job. In addition, based on the time that workers spend to answer various tasks, we can derive a latency model for each worker. Therefore, in this step, for each user, we can conduct mining on both tasks that he/she has performed and the evaluation reports (satisfactory/unsatisfactory and time cost to solve the tasks) to generate a preferred task domain and a time latency model of the worker. In our previous work [4]. we have used data mine methods to mining social users' profile. A similar idea can be applied to build worker's profile for spatial crowdousrcing platform. In gMission [7], we treat each well-performed task as a sample data point and conduct an EM algorithm to derive the task (topic) distribution of the worker. We can also infer the user's ability to conduct various event capturing tasks such as movement speed of the worker, resolution of the camera and precision of GPS devices [9].

For the latency model, we develop a two-phase model [6]. The first phase is the period from the task being published to the task being chosen by a worker; the second phase is the period from the task being accepted by a worker to the answer being returned. Statistical research has been conducted on several crowdsourcing platforms to capture the traits of such latencies [17, 29, 42], such as the one defined as follows.

Definition 1 (Latency): The On-hold Latency L_o of a task (or a batch of tasks) is the clock time from when the task is published to the time when it is accepted by a worker. The Processing Latency L_p of a task (or a batch of tasks) is the clock time from when the task is accepted to the time when the answer is returned and collected by the system. The Overall Latency L is the sum of L_o and L_p: $L = L_o + L_p$.

b) Task Assignment

Based on publishing modes of spatial tasks, the spatial crowdsourcing problems can be partitioned into another two classes: *worker selected tasks* (WST) and *server assigned tasks* (SAT) [23]. In particular, WST publishes spatial tasks on the server side, and workers can choose any tasks without contacting the server; SAT collects location information of all workers to the server, and directly assigns workers with tasks. For example, in the WST mode, some existing works [1,14] allowed users to browse and accept available spatial tasks. However,

there often exist the cases that tasks with low reward or far from the workers will not be served. On the other hand, in the SAT mode, our previous works [23] assumed that the server decides how to assign spatial tasks to workers and the solutions only consider simple metrics such as maximizing the number of assigned tasks on the server side. Compared with WST, SAT is more popular and can control the system throughput and offer balanced workload for each worker.

In SAT, to assign a task to the right crowd, the first question to address is what makes the answers from the crowd wise. Fortunately, a clear solution is provided in [23]. As indicated in [23], four key elements separate wise crowds from irrational ones: Diversity - Each person should have private information even if it's just an eccentric interpretation of the known facts. Independence - People's opinions are not determined by the opinions around them. Decentralization - People are able to specialize and draw on local knowledge. Aggregation- Some mechanism for turning private judgments into a collective decision. Intuitively, independence, decentralization and aggregation can be easily achieved with an appropriate question-answering framework. For instance, a requester may push a yes-no task to a deterministic group of users. Without publishing any answers during the crowdsourcing process, the users would work on the tasks with independence and decentralization. Therefore, the major challenge left is to find an appropriate group of users with high diversity. To achieve this, we need to measure the similarity among users, which again relates to the profiles of users. Considering user's profile as a multi-attribute vector, we can apply different similarity measures for different attributes and combine them by assigning different weights. For example, we can use the Euclidean distance to measure the similarity of users' positions and Cosine similarity for users' expertise which are represented by a set of keywords.

Based on users' profiles and measures about users' diversity, we can work on solving the issue of task assignment. Specifically, given a set of spatial crowdsourcing tasks, we want a strategy to assign tasks to workers to achieve the maximum number of task assignment in real time and meanwhile reduce the cost of each participating workers in terms of their travel distances to the specified crowdsourcing location. Clearly, due to the dynamic properties of workers, we cannot obtain a global picture of all workers at a specific point in time. Therefore, we try to obtain local optimal assignment solutions. Moreover, there are many constraints to assigning tasks, such as the locations (specifying where the task should be performed), the expertise (the domains that a worker knows well), the latency (the time taken to complete the task), reliability (the success rate or the confidence with which a worker can perform certain tasks) and diversity of the workers (dissimilar and independent workers). [23] studied the spatial crowdsourcing with the goal of static maximum task assignment, and proposed several heuristics approaches to enable fast assignment of workers to tasks. Similarly, Deng et al. [14] tackled the problem of scheduling spatial tasks for a single worker such that the number of completed tasks by this worker is maximized. As proved in [23], the task assignment problem in SAT is NP-hard and we can focus on designing various heuristic and approximate solutions to conduct the optimal assignment.

We further study the spatial-temporal diversity and reliability in task assignment [10]. The reliability indicates the confidence that at least some worker can successfully complete the task, whereas the spatial/temporal diversity reflects the quality of the task accomplishment by a group of workers, in both spatial and temporal dimensions (e.g., taking photos from diverse angles and at diverse timestamps). We prove that task assignment with the consideration of spatial-temporal diversity and reliability is also NP-hard, and thus we propose three approximation algorithms (i.e., greedy, sampling, and divide-and-conquer). In addition to handle tasks which require single-skilled workers, we investigate solutions to handle complex tasks requiring multi-skilled workers [8]. For such a complex task assignment, we want to maximize an assignment score (i.e., flexible budget, given by the total budget of the completed tasks minus the total travelling cost of workers). Moreover, we also need to consider several constraints, such as skill-covering, budget, time, and distance constraints, which make the complex assignment problem more challenging. Again, we prove that the multi-skilled complex task assignment problem is NP-hard and intractable. Therefore, we propose three effective heuristic approaches, including greedy, g-divide-and-conquer and cost-model-based adaptive algorithms to get worker-and-task assignments.

The task assignment solutions that we discussed so far focus on offline scenarios, where the server knows all the spatiotemporal information of tasks and workers in advance. However, this offline scenario is quite un-

realistic since tasks and workers in real applications appear dynamically and their spatiotemporal information are often unknown in advance. Thus, in our recent work [41, 44], we have investigated solutions for online task assignment. In [41], to address the shortcomings of existing offline approaches, we first identify a more practical micro-task allocation problem, called the Global Online Microtask Allocation in spatial crowdsourcing (GOMA) problem, then, we extend the state-of-art algorithm for the online maximum weighted bipartite matching problem to the GOMA problem as the baseline algorithm. We further develop a two-phase-based framework targeting on the average performance of online assignment algorithms. Based on the proposed framework, we present an efficient algorithm with $1/4$-competitive ratio under the online random order model. To improve its efficiency, we further design the TGOA-Greedy algorithm, which runs faster than the TGOA algorithm but has lower competitive ratio of $1/8$. In [44], aiming at offering mutual benefit to both workers and tasks, we propose a task assignment framework, called task assignment with mutual benefit awareness (TAMBA). TAMBA captures both sides preferences based on the historical data. With the awareness of both sides preferences, the tasks are assigned to the workers in order to achieve the maximum mutual benefit. Specifically, TAMBA is built on top of a crowdsourcing platform (e.g. gMission), and consisted of two core components: the preference estimator and the AMMB assignor. The preference estimator is built based on a probabilistic graphical model, which is trained offline with historical data (the training of the graphical model is offline conducted and NOT a part of the assignment process). When tasks are submitted, it generates the preference scores (presented in the next section) between the submitted tasks and all the workers. With the preference scores, the AMMB assignor allocate the tasks to the workers based on the output of AMMB assignor. To optimize the performance, we have developed algorithms for the offline and online settings of the same assignment, respectively.

5 Answer Aggregation

After collecting the data from the workers, our next task is to aggregate the results from the workers. Depending on the types of crowdsourcing tasks, different data aggregation models should be investigated. Markets have been proven to be an effective institution for aggregating beliefs of users and yielding reliable answers. For example, in racetrack betting [16], investors are only allowed to choose from two options, and the promised rewards are only given to the investors whose investment meets the market opinion (the majority of all investors). We apply this idea to the aggregation model of workers' answers [5]. Here, workers are considered as investors in this market, thus a wise market includes a set of investors, each of which is associated with a probability (individual confidence) that he or she will give the correct answer. This confidence can be mined from the historical results contributed by the workers. The SCS sever maintains a pool of candidate workers while evaluating the worker' confidence simultaneously. When a particular task comes along, the SCS server uses the solutions proposed in the task assignment step to find an optimal subset of workers and releases the task to them with promise of a reward. After receiving their choices, the answer preferred by a majority of the workers will be given as the market opinion and the ones who make the same choice as the market are granted a reward. Note that the output is the majority of the market, which is not necessarily the same as the ground truth. Compared to simple majority voting, only the workers who make the same choice as the majority will get the reward. This mechanism helps prevent malicious workers, who might return random answers for the reward. With this market aggregation model, the key issue is how to compute the result confidence efficiently. The market confidence is the probability that the aggregation result of C investors (C is larger than half of the selected crowd) is the same as the ground truth. We can use divide-and-conquer strategy to recursively divide the crowd into two groups and compute the confidence for each group. Note that the above discussed aggregation model is designed for simple decision tasks. In [3], we extend the idea of changing "workers" to "traders" to a more complex problem, opinion elicitation. We use Bayesian updating, beginning with our initial guess as the prior, to obtain a posterior distribution that reflects the weighted opinions of all the traders in the market. The payoff for each trader is proportional to her contributed modification from the prior to the posterior.

However, for spatial crowdsourcing, there may exist needs to perform more complicated tasks, such as ranking or capturing events. Therefore, we need to investigate the following issues:

1. Aggregation module for the ranking tasks: since each worker will only return partial ranks, we can apply different voting rules, such as Copeland, maximin, Bucklin, and ranked pairs, or even considering the distance from the worker to the task as the weighting factor [21], to aggregate the answers. However, it was proved that aggregation under these rules is NP-complete [43]. Thus we need to investigate efficient heuristic or sampling solutions.

2. Aggregation module for capturing some event or scenery: since malicious workers may post fake images or videos possibly downloaded from Internet to obtain the reward, we need to design solutions to aggregate them effectively and efficiently. To find such images or videos, we can extend our solutions in detecting near duplicated images or videos [45] to remove those that have appeared a few times.

6 Responses and Quality Control

In the last step, the aggregated results will be returned to the requester. However, there is a big and challenging problem about the quality of the returned results. From the database point of view, crowdsourcing applications break the Closed World assumption that data not available in the database can be considered as non-existing. Data quality measurement for Collected Contents varies according to the specific usage. To deal with the data quality challenge, many data-driven solutions have been proposed. The data-driven approaches are mostly compelled by the data collected so far or the data to be collected. The philosophies of designing such solutions vary according not only to specific tasks they are tackling, but also the form of the quality measurement. Amazingly, success in developing such solutions depends on the coherent matching between the reasonable understanding of the uncertainty, and the proper mathematical (probabilistic) model that agrees with the semantics. We can investigate these quality issues along the following two dimensions:

(a). Probabilistic Data Fusion

Probabilistic Data Fusion approaches are mostly adopted in the crowdsourcing scenario when the workers are asked to choose one or multiple answers from a set of candidates [26]. Two intrinsic observations of such scenario are:

1) There exists the ground truth (or real correct answer) for such decision-making problem, no matter whether the truth is latent or obvious;

2) There are workers with low quality, but there are no spam or biased workers. The two observations lead to a fundamental mathematical description of the uncertainty: The correctness of each worker (w_i)'s answer is a Bernoulli random variable, with identifiable accuracy ϵ_i. Given a voting scheme (e.g. the simplest one, majority voting) and a set of n workers, $W_n = (w_1, w_2, , w_n)$, the overall answer is:

$$OA(W_n) = \begin{cases} 1 & \text{if } \sum w_i \geq \lceil \frac{n}{2} \rceil \\ 0 & \text{if } \sum w_i < \lfloor \frac{n}{2} \rfloor \end{cases}$$

Not surprisingly, the aggregator does not rule out any error-prone ingredients, but installs a straightforward synthesizer. If the answers from different workers are i.i.d. the overall data quality (OQ) can be measured as the reliability of the outcome from Majority Voting:

$$\begin{aligned} OQ(W_n) &= \Pr(OA(W_n) = G) \\ &= \Pr(\big|C\big| \geq \lceil \frac{n}{2} \rceil) \end{aligned}$$

where C is the number of correct workers and G stands for the ground truth. Despite the rigorous definition of data quality, the challenge remains in a practical way. The number of correct workers is essentially a random variable following Poisson Binomial Distribution, on which even the probability density function is intractable.

(b). Data Cleaning

Data cleaning is another way to deal with conflicting crowdsourced data. Compared to data fusion approaches, the data cleaning methods rely relatively less on the probabilistic semantic, internal consistency and dependency receive more consideration,. In one word, the conflict among the data is due to the existence of wrong tuples, and the correct answers are just self-justified. The goal of data cleaning is thus to eliminate wrong items via the internal consistencies or dependencies. We can employ probabilistic graphical models [12, 13] to capture the data quality issues derived from intricate correlation and apply our previous work on matching dependencies [34–37] to clean the results.

Neither approach above takes into consideration the movement patterns of mobile users. Very often, the users follow the same movement pattern in weekdays (commuting from home to the office). Based on these movement patterns, we can easily tell whether the data provided by the worker are correct or not according to her usual location at that time. Thus, we can incorporate movement patterns (along spatio-temporal domain) into quality control solutions.

7 Privacy Issues

Other than the four topics that we have discussed above, privacy is an important issue that needs to be addressed along all the steps in spatial crowdsourcing applications. During the process of spatial crosscutting, users, especially workers' position information need to be released in order to assign the tasks to the nearby workers. However, workers' position and their moving trajectory information are very sensitive information, which can be used to identify individual workers. Therefore, some of our previous works [22,40] investigate how to tackle the privacy leakage problem in spatial crowdsourcing. In [22], we first formally define the problem of privacy leakage in spatial crowdsourincg system and identify its unique challenges due to the existence of an un-trusted central data server. We propose a privacy-aware framework by submitting group and independent queries instead of individual and correlated ones, which enables participation of users without compromising their privacy. In [40], again we assume the existence of a trusted server named *cellular service provider*, we propose a spatial crowdsourcing framework that can achieve an acceptable assignment success rate and total worker travel distance without privacy leakage. Workers first submit their accurate locations to the cellular service provider, then the provider desensitize these locations to a private spatial decomposition for the crowdsourcing server to access. When a new task comes, the crowdsourcing server issues some region queries to the Private Spatial Decomposition (PSD) to determine a specific region to notice. And at last we use a technique named geocast to send the task information to this region.

8 Conclusion

Spatial Crowdsourcing, as a new direction for crowdsourcing has become more and more popular and has been applied in many real applications. In this paper, we first give an overview of the spatial crowdsourcing framework and then illustrate the challenges and opportunities of the spatial crowdsourcing step by step, including incentive design, task assignment, answer aggregation and quality control. Finally, we have presented some works on privacy-preserved spatial crowdsourcing. Given the popular usage of smart phones and advanced development of AI, we believe there will be many challenges and opportunities for spatial crowdsourcing applications.

References

[1] Florian Alt, Alireza Sahami Shirazi, Albrecht Schmidt, Urs Kramer, and Zahid Nawaz. Location-based crowdsourcing: Extending crowdsourcing to the real world. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries*, NordiCHI '10, pages 13–22, 2010.

[2] D. C. Brabham. Crowdsourcing as a model for problem solving: An introduction and cases. *Convergence*, 14(1):75–90, 2008.

[3] Caleb Chen Cao, Lei Chen, and Hosagrahar Visvesvaraya Jagadish. From labor to trader: Opinion elicitation via online crowds as a market. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1067–1076, 2014.

[4] Caleb Chen Cao, Jieying She, Yongxin Tong, and Lei Chen. Whom to ask?: Jury selection for decision making tasks on micro-blog services. *Proc. VLDB Endow.*, 5(11):1495–1506, July 2012.

[5] Caleb Chen Cao, Yongxin Tong, Lei Chen, and H. V. Jagadish. Wisemarket: A new paradigm for managing wisdom of online social users. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 455–463, 2013.

[6] Chen Cao, Jiayang Tu, Zheng Liu, Lei Chen, and H. V. Jagadish. Tuning crowdsourced human computation. *Arxiv.org Technical Report*, page https://arxiv.org/submit/1693767, 2016.

[7] Zhao Chen, Rui Fu, Ziyuan Zhao, Zheng Liu, Leihao Xia, Lei Chen, Peng Cheng, Caleb Chen Cao, Yongxin Tong, and Chen Jason Zhang. gmission: A general spatial crowdsourcing platform. *Proc. VLDB Endow.*, 7(13):1629–1632, August 2014.

[8] Peng Cheng, Xiang Lian, Lei Chen, Jinsong Han, and Jizhong Zhao. Task assignment on multi-skill oriented spatial crowdsourcing. *IEEE Transactions on Knowledge and Data Engineering*, 28(8):2201–2215, 2016.

[9] Peng Cheng, Xiang Lian, Lei Chen, and Cyrus Shahabi. Prediction-based task assignment on spatial crowdsourcing. *Arxiv.org Technical Report*, page https://arxiv.org/abs/1512.08518, 2016.

[10] Peng Cheng, Xiang Lian, Zhao Chen, Rui Fu, Lei Chen, Jinsong Han, and Jizhong Zhao. Reliable diversity-based spatial crowdsourcing by moving workers. *Proc. VLDB Endow.*, 8(10):1022–1033, June 2015.

[11] Antin J. Cheshire, C. The social psychological effects of feedback on the production of internet information pools. *Comput. Mediat. Commun*, 13:705–727, 2008.

[12] Mausam Christopher H. Lin and Daniel S. Weld. Crowdsourcing control: Moving beyond multiple choice. In *Proceedings of the 2012 Uncertainty in Artificial Intelligence*, UAI '12, pages 491–500, 2012.

[13] Peng Dai, Mausam, and Daniel S. Weld. Decision-theoretic control of crowd-sourced workflows. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI'10, pages 1168–1174, 2010.

[14] Dingxiong Deng, Cyrus Shahabi, and Ugur Demiryurek. Maximizing the number of worker's self-selected tasks in spatial crowdsourcing. In *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, SIGSPATIAL'13, pages 324–333, 2013.

[15] Sebastian Deterding, Miguel Sicart, Lennart Nacke, Kenton O'Hara, and Dan Dixon. Gamification. using game-design elements in non-gaming contexts. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '11, pages 2425–2428, 2011.

[16] David A. Easley and Jon M. Kleinberg. *Networks, Crowds, and Markets - Reasoning About a Highly Connected World*. Cambridge University Press, 2010.

[17] Siamak Faridani, Björn Hartmann, and Panagiotis G Ipeirotis. What's the right price? pricing tasks for finishing on time. In *Hcomp 2011*.

[18] David Geer. E-micropayments sweat the small stuff. *Computer*, 37(8):19–22, August 2004.

[19] B. Hoh, T. Yan, D. Ganesan, K. Tracton, T. Iwuchukwu, and J.-S. Lee. Trucentive: A game-theoretic incentive platform for trustworthy mobile crowdsourcing parking services. In *Proceedings of International IEEE Conference on Intelligent Transportation Systems*, ITSC12, pages 160–166, 2012.

[20] Jeff Howe. *Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business*. Crown Publishing Group, New York, NY, USA, 1 edition, 2008.

[21] Huiqi Hu, Yudian Zheng, Zhifeng Bao, Guoliang Li, Jianhua Feng, and Reynold Cheng. Crowdsourced POI labelling: Location-aware result inference and task assignment. In *Proceedings of the 32nd IEEE International Conference on Data Engineering*, pages 61–72, 2016.

[22] Leyla Kazemi and Cyrus Shahabi. A privacy-aware framework for participatory sensing. *SIGKDD Explor. Newsl.*, 13(1):43–51, August 2011.

[23] Leyla Kazemi and Cyrus Shahabi. Geocrowd: Enabling query answering with spatial crowdsourcing. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '12, pages 189–198, 2012.

[24] Seon Ho Kim, Ying Lu, Giorgos Constantinou, Cyrus Shahabi, Guanfeng Wang, and Roger Zimmermann. Mediaq: Mobile multimedia management system. In *Proceedings of the 5th ACM Multimedia Systems Conference*, MMSys '14, pages 224–235, 2014.

[25] Irwin King, Jiexing Li, and Kam Tong Chan. A brief survey of computational approaches in social computing. In *Proceedings of the 2009 International Joint Conference on Neural Networks*, IJCNN'09, pages 2699–2706, 2009.

[26] Qiang Liu, Jian Peng, and Alexander Ihler. Variational inference for crowdsourcing. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, NIPS'12, pages 692–700, 2012.

[27] Tie Luo and Chen-Khong Tham. Fairness and social welfare in incentivizing participatory sensing. In *Proceedings of IEEE Communications Society Conference on Sensor, Mesh, and Ad Hoc Communications and Networks*, SECON '12, pages 425–433, 2012.

[28] Y. Yilmaz M. Bulut and M. Demirbas. Crowdsourcing location-based queries. In *Proceedings of Pervasive Computing and Communications Workshops (PERCOM Workshops)*, Percom'011, pages 513–518, 2011.

[29] Winter Mason and Duncan J. Watts. Financial incentives and the "performance of crowds". In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '09, pages 77–85, 2009.

[30] Prashanth Mohan, Venkata N. Padmanabhan, and Ramachandran Ramjee. Nericell: Rich monitoring of road and traffic conditions using mobile smartphones. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, SenSys '08, pages 323–336, 2008.

[31] Mohamed Musthag, Andrew Raij, Deepak Ganesan, Santosh Kumar, and Saul Shiffman. Exploring micro-incentive strategies for participant compensation in high-burden studies. In *Proceedings of the 13th International Conference on Ubiquitous Computing*, UbiComp '11, pages 435–444, 2011.

[32] Alexander J. Quinn and Benjamin B. Bederson. Human computation: A survey and taxonomy of a growing field. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1403–1412, 2011.

[33] Christoph Schlieder, Peter Kiefer, and Sebastian Matyas. Geogames: Designing location-based games from classic board games. *IEEE Intelligent Systems*, 21(5):40–46, September 2006.

[34] Shaoxu Song and Lei Chen. Differential dependencies: Reasoning and discovery. *ACM Trans. Database Syst.*, 36(3):16:1–16:41, August 2011.

[35] Shaoxu Song, Lei Chen, and Hong Cheng. Parameter-free determination of distance thresholds for metric distance constraints. In *Proceedings of the 28th IEEE International Conference on Data Engineering*, ICDE '12, pages 846–857, 2012.

[36] Shaoxu Song, Lei Chen, and Philip S. Yu. On data dependencies in dataspaces. In *Proceedings of the 27th IEEE International Conference on Data Engineering*, ICDE '11, pages 470–481, 2011.

[37] Shaoxu Song, Lei Chen, and Philip S. Yu. Comparable dependencies over heterogeneous data. *The VLDB Journal*, 22(2):253–274, April 2013.

[38] R. Laubacher T. W. Malone and C. Dellarocas. Harnessing crowds: mapping the genome of collective intelligence. *Technical Report, MIT Sloan Research Paper*, 4732(09), 2009.

[39] Yuandong Tian and Jun Zhu. Learning from crowds in the presence of schools of thought. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 226–234, 2012.

[40] Hien To, Gabriel Ghinita, and Cyrus Shahabi. A framework for protecting worker location privacy in spatial crowdsourcing. *PVLDB*, 7(10):919–930, 2014.

[41] Yongxin Tong, Jieying She, Bolin Ding, Libin Wang, and Lei Chen. Online mobile micro-task allocation in spatial crowdsourcing. In *Proceedings of the 32nd IEEE International Conference on Data Engineering*, ICDE'16, pages 49–60, 2016.

[42] Jing Wang, Siamak Faridani, and P Ipeirotis. Estimating the completion time of crowdsourced tasks using survival analysis models. *CSDM 2011*.

[43] Lirong Xia and Vincent Conitzer. Determining possible and necessary winners under common voting rules given partial orders. In *AAAI*, 2008.

[44] Liu Zheng and Lei Chen. Mutual benefit aware task assignment in a bipartite labor market. In *Proceedings of the 32nd IEEE International Conference on Data Engineering*, ICDE'16, pages 73–84, 2016.

[45] Xiangmin Zhou, Lei Chen, and Xiaofang Zhou. Structure tensor series-based large scale near-duplicate video retrieval. *IEEE Trans. Multimedia*, 14(4):1220–1233, 2012.