Balancing Quality and Budget Considerations in Mobile Crowdsourcing

Chunyan Miao[a,c], Han Yu[c], Zhiqi Shen[a,c], Cyril Leung[b,c]

[a]*School of Computer Science and Engineering (SCSE), Nanyang Technological University (NTU), Singapore 637659*
[b]*Department of Electrical and Computer Engineering, The University of British Columbia (UBC), Vancouver, BC, Canada*
[c]*Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly (LILY), NTU, Singapore 637659*

Abstract

Mobile/spatial crowdsourcing is a class of crowdsourcing applications in which workers travel to specific locations in order to perform tasks. As workers may possess different levels of competence, a major research challenge for spatial crowdsourcing is to control the quality of the results obtained. Although existing mobile crowdsourcing systems are able to track a wide range of performance related data for the participating workers, there still lacks an automated mechanism to help requesters make key task allocation decisions including: 1) to whom should a task to allocated; 2) how much to pay for the result provided by each worker; and 3) when to stop looking for additional workers for a task. In this paper, we propose a budget-aware task allocation approach for spatial crowdsourcing (*Budget-TASC*) to help requesters make these three decisions jointly. It considers the workers' reputation and proximity to the task locations to maximize the expected quality of the results while staying within a limited budget. Furthermore, it supports payments to workers based on how their track records. Extensive experimental evaluation-

s based on a large-scale real-world dataset demonstrate that Budget-TASC outperforms the state-of-the-art significantly in terms of reduction in the average error rate and savings on the budget.
Keywords: Budget allocation; reputation; trust; mobile crowdsourcing; crowdsensing.

---

1. Introduction

Crowdsourcing refers to the arrangement in which contributions (such as services, observations, contents, etc.) are solicited from a large group of unrelated people [1, 2]. Many online crowdsourcing platforms, such as the *Amazon's Mechanical Turk (mTurk)* [3], have emerged to provide commercial crowdsourcing services.

As smart phones with easy access to the Global Positioning System (G-PS) and wireless Internet become increasingly ubiquitous, a new breed of crowdsourcing - *mobile crowdsourcing* (a.k.a. *spatial crowdsourcing*) - has emerged. In the rest of this paper, we use the terms "mobile crowdsourcing" and "spatial crowdsourcing" interchangeably. In spatial crowdsourcing, *requesters* crowdsource results for *spatial tasks* from *workers* who are required to travel to specific locations in order to complete the tasks. Examples of spatial crowdsourcing include monitoring traffic flows and monitoring environmental conditions at selected locations [4]; DARPA Network Challenge [5]; and the *Gigwalk* mobile crowdsourcing app (which engages workers to gather information about the locations and products, and provides business intelligence to requesters[1]).

[1] `http://www.gigwalk.com/crowdsourcing`

Social mobilization through mobile crowdsourcing faces significant challenges. One of the most important is how to obtain high quality results from workers with diverse capabilities [6]. The quality of results (i.e., how accurately the collective results from all workers involved in a task reflect the real state of the subjects being studied) obtained from mobile crowdsourcing is influenced by the workers' capabilities and situations. In addition, as workers are required to travel to specific locations to complete the tasks in spatial crowdsourcing, they incur extra costs. This may discourage the workers from participation or providing high quality results. In the mobile crowdsourcing research field, the quality of the result is regarded to be negatively affected by the distance between a worker and the task location [7, 8]. The research problem is to maximize the expected quality of the results obtained, while staying within a limited total budget.

Reputation modelling is a useful technique for estimating a worker's reliability. According to [9], reputation is refers to what is generally believed about a person's character or standing. Many computational approaches for inferring candidates' reputation based on their past performance have been proposed [10, 11, 12, 13]. Considering workers' reputation information and physical distances to task locations while taking into account requesters' budget constraints is a promising approach to improving the quality of spatial crowdsourcing results.

In this paper, we focus on situations where workers are required to travel to a given location in order to complete a spatial task. We focus on the budget allocation problem in which a decision support mechanism incorporated into a mobile crowdsourcing system automatically allocates the task to a number of workers on behalf of the task requester such that the total cost does not exceed a pre-specified budget. The goal is to find an efficient budget allocation that maximizes the expected quality of the collective result provided by the selected workers.

To address this problem, we propose a Budget-aware trustworthy Task Allocation approach for Spatial Crowdsourcing (*Budget-TASC*). It maximizes the expected quality of information from the workers by jointly considering the workers' track records and distances from the task locations. It helps spatial crowdsourcing requesters make decisions on 1) to whom should a task to allocated; 2) how much to pay for the result provided by each worker; and 3) when to stop looking for additional workers for a task. The paper advances the state-of-the-art in the following ways:

1. We formulate the problem of spatial crowdsourcing task allocation with budget constraints as a *Multiple Choice Knapsack Problem (MCKP)* [14];
2. We propose Budget-TASC, a greedy branch and bound algorithm that efficiently solves the spatial crowdsourcing MCKP in $O(N^2)$ time by leveraging on the heuristics related to workers' locations and reputation;
3. Budget-TASC supports spatial crowdsourcing platforms in which rewards for a worker are partially determined by his past performance [15, 16, 17];
4. Budget-TASC eliminates the need for task requesters to pre-specify the number of workers required for a task.

By comparing the performance of Budget-TASC with existing approaches through extensive experimental evaluations based on a real-world data from the *Foursquare* location-based social network, we demonstrate that the proposed approach outperforms the state-of-the-art approaches by over 45% in terms of reduction in the average error rate, and results in more than 15% of savings on the budget.

The rest of this paper is organized as follows. In Section 2, the significance of our work is put into context through a review of related work. Section 3 provides a formalization of the problem. The proposed Budget-TASC approach and theoretical analysis of its performance are presented in Section 4. Section 5 further evaluates the performance of Budget-TASC through extensive numerical experiments. Finally, Section 6 concludes the paper and outlines future research directions.

2. Related Work

A significant amount of research on task and budget allocation has been carried out in classic (i.e., non-spatial) crowdsourcing. We discuss this category of work in Section 2.1. Then, in Section 2.2, we focus on task and budget allocation research in spatial crowdsourcing.

2.1. Task Allocation in Crowdsourcing

As workers in a crowdsourcing situation may possess different capabilities and behavior tendencies, the research on finding efficient task allocation solutions given the reputation information of each worker has attracted significant interest. In [18, 19, 20, 21, 22, 23, 24], the authors jointly considered the workers' resource constraints and reputation to derive task allocation plans based on network queueing theories. However, they did not consider the impact of the spatial separation between the workers and the task locations, or the constraint of a limited budget.

Research works in budget optimization in crowdsourcing have typically focused on maximizing the expected task quality given the number of workers to be assigned to the task. A number of works have used Bayesian Learning-based [25] and POMDP-based techniques [26] to estimate the ground truth. However, they do not accommodate having different costs for different tasks. In [27] and [28], the authors proposed recursive reverse auction-based mechanisms to make workers bid for tasks to save cost and improve quality. However, such approaches generally incur significant overhead in communications and make it complicated for workers to participate.

A representative work in the area of budget optimization for classic crowdsourcing is *CrowdBudget* [29]. It allocates the budget in advance through analysis of cost and expected quality of the results. This approach is a budget allocation approach that divides a given budget among different tasks to achieve low estimation error in crowdsourced classification tasks. As this approach demands prior input for the number of workers, n_k, to which a task needs to be allocated. Since this approach is not designed for spatial crowdsourcing, it did not specify how location information should be considered. CrowdBudget also assumes that all workers assigned the same spatial task are paid equally regardless of their individual track record.

2.2. Task Allocation in Spatial Crowdsourcing

Compared to classic crowdsourcing, task allocation research in spatial crowdsourcing has received less attention. In [7], a crowdsourcing platform which allocate tasks to workers based on their locations have been proposed. The authors also observed that workers main prefer solving tasks in close proximity of their homes. Nevertheless, the proposed approach does not take workers' past performance into account, nor does it support rewarding workers differently based on their track records.

In [30] and [8], the authors discussed the special characteristics of spatial crowdsourcing, and proposed an efficient heuristic-based greedy approach - *GeoTruCrowd* - that can produce close to optimal task allocation solutions. GeoTruCrowd maximizes the number of spatial crowdsourcing tasks assigned to a set of workers while satisfying the confidence levels required for the task results. It gives preference to solutions that require the least aggregate travel distance for the workers involved. However, they focused on volunteer-based spatial crowdsourcing in which people are self-motivated to perform the tasks without expecting rewards, and did not consider the budget limitations.

In contrast, Budget-TASC not only considers workers' reputation and locations when allocating tasks, but can also support crowdsourcing platforms which distinguish workers' rewards based on their track records.

3. Problem Formulation

3.1. Motivating Scenario

The problem of efficient large-scale sensing of the physical world at low cost, which is the focus of this paper, has its roots in many real-world applications. One of the most significant is crowdsensing for disaster response. For example, during the Nepal earthquake in April 2015, the ORCHID[2] project from the UK set up a system leveraging crowdsensed reports about damages in various locations across the affected area to coordinate rescue efforts [31]. In this type of applications, workers may have different capabilities in terms of assessing the severity of damage, or they may wish to exaggerate the damage to certain extent in order to channel rescue resources to the locations they prefer. Thus, multiple crowd workers often need to be dispatched to each location in order to obtain redundant reports which can be aggregated to improve the information accuracy. Reputation modelling is also needed to keep track of each worker's performance over time. Also, as the crowd workers incur travel costs, which can be significant in disaster situations due to damages to the transport network, this factor has to be taken into account when allocating workers to task locations.

3.2. System Model

In this section, we formalize the research problem spatial crowdsourcing task allocation. For the readers' convenience, symbols used in this paper are listed in Table 1 in alphabetical order.

We consider the problem of delegating a spatial crowdsourcing task τ_i proposed by a requester i to N candidate workers subject to a budget limit of $B^{\tau_i} \in \mathbb{R}^+$. A spatial task τ_i is represented as a tuple of the form $\langle l^{\tau_i}, R^{\tau_i}, B^{\tau_i}, p_H^{\tau_i}, p_M^{\tau_i} \rangle$. Similar to the model in [8], information related to a spatial task includes l^{τ_i}, which is the location of the task specified by a point in 2D space (e.g., represented by a latitude-longitude coordinate), and R^{τ_i},

[2] `http://www.orchid.ac.uk/`

which is the radius of the spatial region within which workers are most likely to accept the assigned task.

In our model, we assume that a reputation mechanism is available to track the past performance of the workers as envisioned in [1]. A worker's

Table 1: List of Symbols

Symbol	**Meaning**
B^{τ_i}	The total budget for task τ_i
$c_j^{\tau_i}$	Worker j's credibility in the context of task τ_i
D_c	The diameter of a given area c (e.g., a city/town)
i	A mobile crowdsourcing requester
j	A mobile crowdsourcing worker
l^{τ_i}	The location of a task τ_i
N	The total number of workers in a mobile crowdsourcing system
N^{τ_i}	The number of workers involved in task τ_i
$o_j^{\tau_i}$	The actual result provided by worker j for task τ_i
O^{τ_i}	The aggregated result for task τ_i provided by involved workers
P_{τ_i}	The task allocation plan calculated by Budget-TASC for a given task τ_i
$p_H^{\tau_i}$	The reward for a worker with high reputation for completing task τ_i
$p_M^{\tau_i}$	The reward for a worker with medium reputation for completing task τ_i
r_j	Worker j's reputation at the time when Budget-TASC tries to allocate tasks
R^{τ_i}	The radius of the spatial region within which to recruit workers for τ_i
τ_i	A task proposed by requester i
T^{τ_i}	The ground truth of task τ_i
Th_{HM}	The threshold value separating high and medium reputation levels
Th_{ML}	The threshold value separating medium and low reputation levels

reputation for performing different types of tasks can easily be tracked separately and will not affect the operation of the proposed approach. The workers' reputation information is used by the spatial crowdsourcing system to determine how much reward they should receive for each task. The reward can be any form of incentive or combination of incentives such as monetary payment or points which can be exchanged for goods or services.

For simplicity of discussions, workers are classified as having *high*, *medium* or *low* reputation standings in this paper based on their reputation values. Nevertheless, our problem formulation is flexible enough to accommodate finer granularities of reputation division. It is reasonable to assume that a requester would not want to engage a worker with low reputation. $p_H^{\tau_i}$ and $p_M^{\tau_i}$ are the non-zero amounts of money the task requester is willing to pay for a result provided by a worker with high reputation and a worker with medium reputation, respectively ($0 \leqslant p_M^{\tau_i} \leqslant p_H^{\tau_i} \leqslant B^{\tau_i}$). The values of $p_H^{\tau_i}$ and $p_M^{\tau_i}$ can be different across different tasks.

The three reputation standings can be customized by each task requester or set by the crowdsourcing system operator using two threshold values: Th_{HM} for separating high and medium reputation levels, and Th_{ML} for separating medium and low reputation levels ($0 \leqslant Th_{ML} \leqslant Th_{HM} \leqslant 1$). Following the practice in existing crowdsourcing systems, such as mTurk [3], workers start from a medium reputation standing, and only workers with medium or high reputation standing are allowed to participate in tasks. A worker j's reputation any given time, r_j, can take values within the range of $[0, 1]$ (with 0 indicating the least trustworthy and 1 indicating the most trustworthy).

We assume that the ground truth of a task τ_i (i.e., the correct answer to the spatial crowdsourcing task) is binary. This assumption is reasonable as it is true in many real-world situations (e.g., whether the road work at a particular location has been completed). When recording a worker j's past performance, each time j provides a result for a task, the outcome $o_j^{\tau_i} \in \{0, 1\}$ is binary. Let $Tr^{\tau_i} \in \{0, 1\}$ be the ground truth of a task τ_i. In practice, as the ground truth of a task τ_i is unknown to the requester i, he cannot deterministically evaluate the performance of the participating workers. To circumvent this limitation, in applications such as *Baidu's Crowd Test*[3], the majority opinion from the selected workers, O^{τ_i}, is used to approximate the ground truth. O^{τ_i} can be formalized as:

$$O^{\tau_i} = \lfloor \frac{1}{N^{\tau_i}} \sum_{j=1}^{N^{\tau_i}} o_j^{\tau_i} - \frac{1}{2} \rfloor + 1 \tag{1}$$

where N^{τ_i} denotes the actual number of workers who participated in task τ_i. In other words, $O^{\tau_i} = 1$ if the average outcome is no less than $\frac{1}{2}$, and is 0 otherwise. There is a large number of reputation evaluation models available which can be used to compute a worker's reputation [32, 33, 34]. Budget-TASC can be used together with any reputation evaluation model as long as the computed reputation value can be normalized to the range of $[0, 1]$.

When workers' reputations need to be tracked, there is always the possibility that the workers would try to game the reputation mechanism in order to obtained unfair advantages. Existing research works such as [35, 36] have provided mechanisms to mitigate attempts to distort reputation information. As Budget-TASC is modularized to treat workers' reputation values as its

[3] `http://test.baidu.com/`

inputs, it can be used in conjunction with these mitigation mechanisms in order to deal with distorted reputation information.

As evaluating a worker's reputation and dealing with reputation distort are not the focus of this paper, we adopt the popular Beta Reputation System (BRS) [37] for simplicity of discussion. BRS tracks a worker's performance using two variables:

$$s_j = \sum_i 1_{[o_j^{\tau_i} = O^{\tau_i}]}, f_j = \sum_i 1_{[o_j^{\tau_i} \neq O^{\tau_i}]} \tag{2}$$

where s_j and f_j are the number of "correct" and "incorrect" results provided by j. Here, "correct" refers to conformity to the majority opinion. $1_{[\text{condition}]}$ is a function that evaluates to 1 if [condition] is true, and to 0 otherwise. Then, a worker j's reputation based on his performance in the N_j tasks he has performed in the past is:

$$r_j \triangleq \frac{\alpha + 1}{\alpha + \beta + 2} \in (0, 1) \tag{3}$$

where α and β are defined as:

$$\alpha = \sum_{t=0}^{N_j} s_j, \beta = \sum_{t=0}^{N_j} f_j. \tag{4}$$

It is well-known that BRS does not work well in the presence of malicious workers. There have been many research works in trustworthy computing addressing the problem of malicious workers [38]. As the approach proposed in this paper does not try to address this problem, but makes use of workers' reputation values produced by any reputation evaluation model to perform task allocation, the problem of malicious workers is not within the scope of this paper. BRS is used here to simplify subsequent discussions and avoid distracting readers with too much treatment on malicious worker behaviors. The proposed approach is one layer up from the reputation evaluation models layer, which should handle the malicious workers problem.

Similar to traditional crowdsourcing tasks, in spatial crowdsourcing, a requester i may wish to obtain corroborating results from multiple workers. However, in the case of spatial crowdsourcing, the quality of a result depends on two main factors:

1. Firstly, it is affected by a worker's intrinsic trustworthiness. A more trustworthy worker (as reflected by his good reputation standing) generally provides more reliable results than less trustworthy workers.
2. Secondly, the distance between a worker's location and the location of the spatial task influences a workers' cost of travel (and hence his willingness to travel to the task location) [7], which in turn, impacts the likelihood for the requester to receive quality results from this worker.

For example, the credibility of a result provided by Worker 1 who is at a distance d_1 from the task location should be decreased more than that provided by Worker 2 who is at a distance d_2 from the task location for $d_1 > d_2$. Based on the combined consideration of these two factors, a worker j's credibility in the context of a spatial task τ_i is:

$$c_j^{\tau_i} = r_j \cdot \delta(l_j, l^{\tau_i}) \tag{5}$$

where l_j is the worker's location at the time when the task request is delegated to him. $\delta(l_j, l^{\tau_i})$ is a function calculating the discount to the worker's reputation as a result of his proximity to the task location. In this paper, it is defined as:

$$\delta(l_j, l^{\tau_i}) = 1 - \max[0, \min[\log_{D_c}(d(l_j, l^{\tau_i})), 1]] \tag{6}$$

where $d(l_j, l^{\tau_i})$ calculates the Euclidean distance between two GPS coordinates in kilometers. This can be accomplished by using the *Haversine* formula [39]. As spatial crowdsourcing tasks tend to be micro-tasks which are not very complex and carry small amounts of monetary rewards, it is generally not worthwhile to recruit workers from too far away to travel to the task locations. Thus, we assume that the maximum range to look for candidate workers is within the same city as the proposed spatial task. The distance discount is a function of $\log_{D_c}$ where D_c is the diameter of a given area c (e.g., a city/town). Therefore, $\delta(l_j, l^{\tau_i}) \in [0, 1]$. The closer a worker j is to the location of a task τ_i, the closer $\delta(l_j, l^{\tau_i})$ is to 1. When worker j is at or beyond a distance of D_c km from τ_i, $\delta(l_j, l^{\tau_i}) = 0$.

This formulation is a simplification of real-world situations as terrain, noise interference, or direct line-of-sight may also play a role in affecting the quality of the results. Nevertheless, it can be extended to incorporate these factors in such a way that the presence of unfavourable factors effectively *increases* the separation between the worker and the task.

4. The Budget-TASC Approach

Given the workers' reputation and location information, the objective is to find a task allocation plan on behalf of the task requester to maximize the expected quality of the result while satisfying budget limitations. As the expected result quality is affected by the credibility of selected workers, this objective can be expressed as an optimization problem:

Maximize:

$$\sum_{j=1}^{N} c_j^{\tau_i} \tag{7}$$

subject to:

$$\sum_{j=1}^{N}\sum_{g=1}^{G} \psi_{j,g} p_{j,g} \leqslant B^{\tau_i} \tag{8}$$

$$\psi_{j,g} \in \{0,1\}, \forall j \in \{1,...,N\}, \forall g \in \{1,...,G\} \tag{9}$$

$$\sum_{g=1}^{G} \psi_{j,g} = 1, \forall j \in \{1,...,N\} \tag{10}$$

where $p_{j,g} \in \{p_H^{\tau_i}, p_M^{\tau_i}, 0\}$ represents the actual reward for a worker j. G denotes the number of reward options for each task ($G = 3$ in this work). The value of $\psi_{j,g}$ depends on j's reputation. Thus, if $r_j \geqslant Th_{HM}$, then only $\psi_{j,1} = 1$ (i.e., j is paid $p_H^{\tau_i}$ dollar if the requester accepts his result); if $Th_{ML} \leqslant r_j < Th_{HM}$, then only $\psi_{j,2} = 1$ (i.e., j is paid $p_M^{\tau_i}$ dollars if the requester accepts his result); otherwise, only $\psi_{j,3} = 1$ (i.e., j is paid 0 dollars which means j is not invited to participate in task τ_i).

The problem of spatial crowdsourcing task allocation is an MCKP. In essence, given a set of workers, each with a credibility value and a cost, we aim to determine the combination of workers to include in a collection so that the total cost is less than or equal to a given budget limit and the total credibility value of the selected workers is as large as possible.

In order to reduce the complexity of the problem, we analyze the structures available in spatial crowdsourcing to identify possible simplifying heuristics. There are two heuristics that are useful in reducing the complexity of the problem:

1. For workers at the same distance from the location of a spatial task, the ones with higher reputation have higher credibility.
2. For workers with the same reputation, the ones closer to the location

---

Algorithm 1 Budget-TASC

Require: A spatial crowdsourcing system with N workers, a given task τ_i.

for $j = 1$ to N **do**
    Compute $c_j^{\tau_i}$ according to equation (5);
end for
Rank workers in descending order of their $c_j^{\tau_i}$;
for $j = 1$ to $\min\left[\left\lfloor \frac{B^{\tau_i}}{p_H^{\tau_i}} \right\rfloor, N\right]$ **do**
    $J = \min\left[N - j, \left\lfloor \frac{B^{\tau_i} - j \cdot p_H^{\tau_i}}{p_M^{\tau_i}} \right\rfloor\right]$;
    $\mathbf{W}_j = \{\emptyset\}$ and $\mathbf{S}_j = \{\emptyset\}$;
    for $k = 1$ to N **do**
        Add worker k to the set $\mathbf{W}_j$ if $d(l_k, l^{\tau_i}) \leqslant R^{\tau_i}$ and $r_k \geqslant Th_{ML}$;
        Break out of the for-loop if $|\mathbf{W}_j| \geqslant (j + J)$;
    end for
    for $k = 1$ to $|\mathbf{W}_j|$ **do**
        Add worker k to the set $\mathbf{S}_j$ if $r_k \geqslant Th_{HM}$;
    end for
    Choose payment option $p_H^{\tau_i}$ for all workers in $\mathbf{S}_j$;
    Choose payment option $p_M^{\tau_i}$ for all workers in $(\mathbf{W}_j - \mathbf{S}_j)$;
    $C_{S_j}^{\tau_i} = \sum_{j=1}^{|S_j|} c_j^{\tau_i}$;
end for
$P_{\tau_i} = \arg\max_{S_j} C_{S_j}^{\tau_i}$;
return P_{τ_i};

---

of a spatial task have higher credibility.

Based on these heuristics, we propose the Budget-TASC approach as shown in Algorithm 1.

When the Budget-TASC approach receives a task request τ_i, it first computes the $c_j^{\tau_i}$ values of all N known workers according to equation (5), which jointly considers the aforementioned heuristics. The workers are then ranked in descending order of their $c_j^{\tau_i}$ values. In effect, workers who are close to the task location, or have high reputation values at the time, or display both attributes are ranked at the top. Next, the proposed approach identifies workers who are within the spatial region of the task. These three steps significantly reduce the number of candidate workers for forming task allocation plans. The algorithm then forms task allocation plans by looking for workers with high reputation standings first (i.e., those who should be paid $p_H^{\tau_i}$ each), and workers with medium reputation standings next if the budget allows it. Each plan has an overall credibility value $C_{S_j}^{\tau_i}$. The plan with the highest $C_{S_j}^{\tau_i}$ is selected as the solution. In the case of ties, preference is given to plans with a lower cost, or requiring less travel.

4.1. Analysis

By using Budget-TASC, the task requesters do not have to deterministically choose specific workers to perform the tasks. Instead, they only submit the proposed spatial tasks to the crowdsourcing system. This is the same as the practice adopted by many open crowdsourcing systems with large worker populations (e.g., 99designs or mTurk). As opposed to state-of-the-art approaches such as CrowdBudget [29], task requesters do not need to pre-specify the number of workers required for a task under Budget-TASC.

For each j value, the Budget-TASC algorithm finds $C_{S_j}^{\tau_i}$ which is the maximum overall credibility of the task allocation plan if j workers provide their results to the task requester. Since each worker's credibility is non-negative (i.e., $c_j^{\tau_i} \in [0, 1]$) and decreases with increasing distance from the task location, there exists an optimal solution, $P_{\tau_i}^*$, that utilizes the same set $\mathbf{W}$ of up to $(j + J)$ workers closest to the task location. For each subset $\mathbf{S}^* \subseteq \mathbf{W}$, let C_{S^*} represent the overall credibility of the task allocation plan following equation (7). Then, C_{S^*} is maximized by the subset of workers with the highest $c_j^{\tau_i}$ values. Thus, the solution found by Budget-TASC through maximizing over all possible j values is *one of the optimal solutions*. Nevertheless, as there is always uncertainty surrounding the quality of information provided by the selected workers in reality, the solution would not be the theoretical optimal solution produced by an Oracle, but rather a better effort solution under the given situations.

In Algorithm 1, the "for" loop between Line 1 and Line 3 iterates N times (where N is the total number of workers in a spatial crowdsourcing system). The "for" loop encompassing Line 5 to Line 18 and the "for" loop encompassing Line 8 to Line 11 both may iterate up to a maximum of N times in the worst case scenario (where B^{τ_i} and R^{τ_i} are large enough to include all N workers and $r_j \geqslant Th_{ML}$ for every worker j). Thus, the time complexity of the Budget-TASC algorithm is $O(N^2)$.

5. Experimental Evaluation

After analyzing the optimality of the solutions produced by Budget-TASC, we now turn to the practical aspects and examine its performance in realistic settings. To this end, we compare the performance of Budget-TASC against two state-of-the-art approaches on a range of spatial crowdsourcing tasks generated from a real-world location-based social network dataset. Specifically, the real-world dataset allows us to construct realistic spatial crowdsourcing scenarios to demonstrate that the proposed approach outperforms existing approaches in practice. The simulation allows us to better understand the behavior of the Budget-TASC by finely varying the parameter settings. We first describe our experiment design and evaluation metrics, and then analyze the results.

5.1. Experiment Design

To test the proposed approach under realistic settings, we use a dataset from the popular location-based social network *Foursquare* [40] with location information for over 2,000,000 users and over 1,000,000 venues.[4] Data for users and venues in the city of Singapore are extracted from this dataset. The resulting dataset contains 13,919 users and 430 venues. In the experiments, the locations of the venues and the users in this subset of data are used to generate the locations for the spatial crowdsourcing tasks and the workers, respectively. The locations of the workers and the tasks are illustrated in Figure 1. Although the dataset is not directly from spatial crowdsourcing, it gives a varied and realistic worker and task location distribution. As we are studying algorithms which depend on location, this dataset enables us to draw some reasonable conclusions about their relative performances.

In addition to location, each worker has an innate reliability parameter,

[4] `https://archive.org/details/201309_foursquare_dataset_umn`

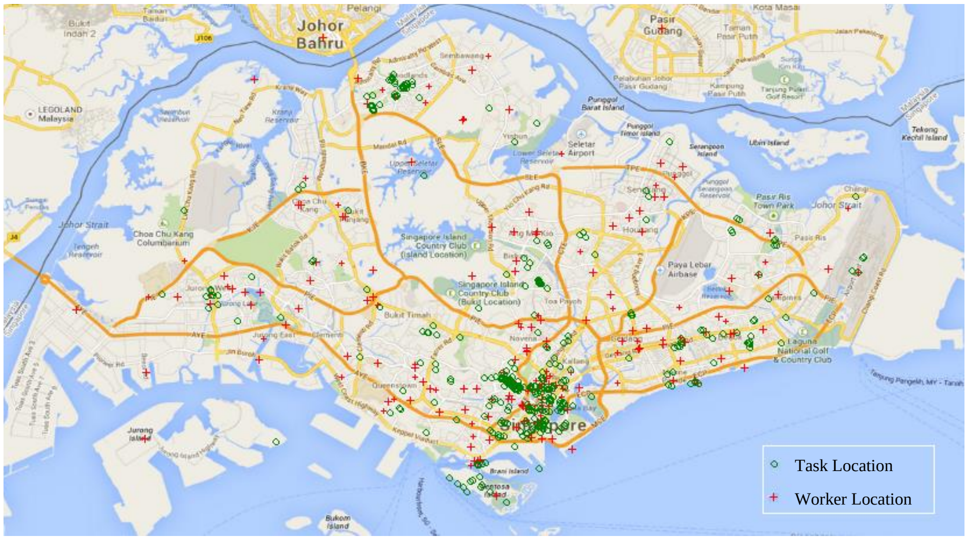


Figure 1: The geo-spatial distribution of workers and tasks in Singapore based on data from *Foursquare*.

h_j, that affects the quality of the results produced by it. At the start of each simulation, the h_j values of the workers are randomly selected following a uniform distribution in the range $[0, 1]$. In effect, it represents the probability of a worker producing a result of acceptable quality if the worker is at the task location. If the distance between the worker and the task is not zero, h_j is discounted by multiplying it with $\delta(l_j, l^{\tau_i})$. Each worker also has a likelihood of participation value $Pr(j)$ which represents the probability of the worker accepting a task request. We set $Pr(j)$ to be directly proportional to the workers' proximity to the task location based on [27].

During the experiment, each of the spatial tasks are allocated to the workers one by one. The ground truth value for each task, $Tr^{\tau_i} \in \{0, 1\}$, is randomly selected from either 0 or 1 with equal probability. Two state-of-the-art approaches most related to Budget-TASC are selected as benchmark approaches. They are:

1. *CrowdBudget* [29]: In the experiments, we set $n_k = \lfloor \frac{B^{\tau_i}}{p_H^{\tau_i}} \rfloor$ for all τ_i assuming preference is given to workers with high reputation standing. Although CrowdBudget is not designed for spatial crowdsourcing, by comparing with it, we provide a perspective on how such approaches perform under spatial crowdsourcing conditions.
2. *GeoTruCrowd* [8]: As this approach was designed for volunteer-based spatial crowdsourcing without involving reward to the workers, it did not specify how budget limitations should be considered. Therefore, in our implementation, the algorithm is stopped as soon as the remaining budget is not enough to employ additional workers or no more eligible workers can be found.

Two parameters are varied in the experiments to simulate different scenarios. They are the task radius, R^{τ_i}, which represents the spatial region of each task, and the total budget, B^{τ_i}.

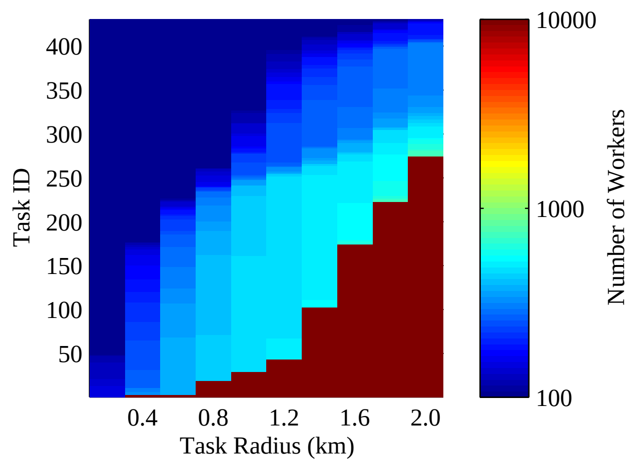


Figure 2: The number of workers "close" to each task in the dataset under different task radius settings.

1. R^{τ_i}: In each experiment, we assume that all spatial tasks share the same R^{τ_i} and B^{τ_i} values. The value for R^{τ_i} is varied from 0.1km to 2.0km in 0.1km increments (i.e., 20 different settings). The number of workers "close" to each task (i.e., within the spatial region of the task) under different task radius (R^{τ_i}) settings are shown in Figure 2. It can be observed that up until a task radius of 1.6km, the majority of the tasks have small to medium crowds (i.e., less than 500 candidate workers) within their respective spatial regions.
2. B^{τ_i}: The value of B^{τ_i} is varied from $5p_M^{\tau_i}$ to $100p_M^{\tau_i}$ in $5p_M^{\tau_i}$ increments (i.e., 20 different settings).

Therefore, a total of $20 \times 20 = 400$ different experiment settings are studied. In all the experiments, the values of $p_H^{\tau_i}$ and $p_M^{\tau_i}$ are set at \$2 and \$1, respectively. We divide the reputation value range from 0 to 1 into three levels by setting the values of Th_{HM} and Th_{ML} to 0.75 and 0.5, respectively. D_c is set to 30km based on the approximate width of Singapore.

After the location and reliability values of a population of 13,919 workers have been generated at the beginning of each experiment, two additional copies were cloned so that the three approaches can be run in parallel for their performances to be compared. To have a fair comparison in all experiments, we use the majority voting rule for aggregating the results returned by the workers involved in each task. The three approaches use equation (2) to assess the workers' performance in each round of crowdsourcing, and adopt the method in [37] to calculate their reputation.

In the experiments, we measure the performance of each approach using the following metrics:

1. *Average Error Rate* (e): This metric measures how effective an approach is in finding credible workers for a spatial task while staying within the budget limitation. The average error rate e is the ratio of incorrect aggregate results (i.e. the aggregate results provided by the workers differ from the ground truth) to the total number of spatial tasks in an experiment (N_τ):

$$e = \frac{1}{N_\tau} \sum_{k=1}^{N_\tau} 1_{[O^{\tau_k} \neq Tr^{\tau_k}]}. \tag{11}$$

2. *Average Budget Utilization Rate* ($B^{(-)}$): This metric is computed as the average ratio of the total amount of money actually spent for each spatial task b^{τ_i}, to the budget for that task during an experiment:

$$B^{(-)} = \frac{1}{N_\tau} \sum_{k=1}^{N_\tau} \frac{b^{\tau_k}}{B^{\tau_k}}. \tag{12}$$

3. *Average Distance Travelled* (D): This metric is computed as the average distance travelled by the workers per spatial task:

$$D = \frac{1}{N_\tau} \sum_{k=1}^{N_\tau} \sum_{j=1}^{n_k} \frac{d(l_j(t), l^{\tau_k})}{n_k} \tag{13}$$

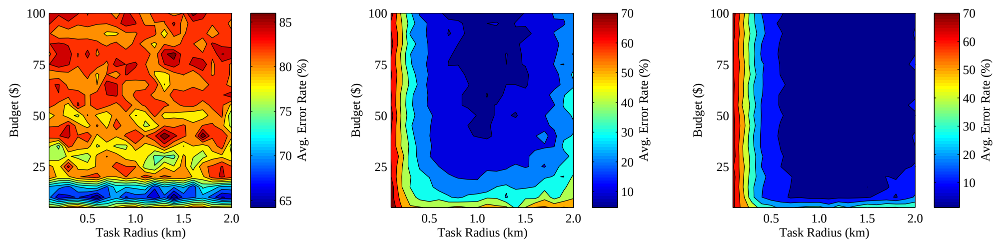
   where n_k is the actual number of workers involved in a spatial task τ_k.

The task requesters' perspective would like to keep e as low as possible. They also prefer keeping $B^{(-)}$ low. On the other hand, the workers would like to keep the travel distance low so as to incur less cost for completing spatial tasks. Therefore, the lower the values for these metrics, the better the performance of an approach.

5.2. Results

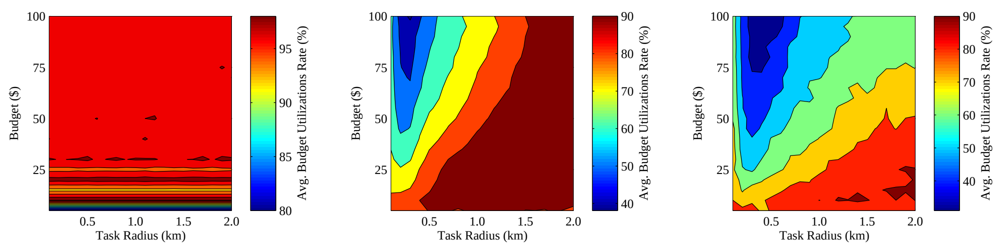
In this section, we present the results achieved by each approach under different experimental settings. The results in terms of average error rates are depicted in Figure 3. The horizontal axis represents different settings of task radius, while the vertical axis represents the different settings of budget size. The colour code represents different e values. The warmer the colour, the larger the average error rate. It can be observed that the average error rates of CrowdBudget do not correlate strongly with either task radius or budget size. This is not surprising as CrowdBudget does not use spatial information for task allocation. When the budget size is small or the task radius is small, the average error rates for both GeoTruCrowd and Budget-TASC are high as the number of eligible workers is small. As budget size and task radius increases, the average error rate of Budget-TASC decreases significantly to below 10% and remains in that region. GeoTruCrowd shows similar improvements initially. However, as task radius increases to 1.7km, the median number of candidate workers within the region increases from around 500 to over 10,000 (Figure 2). At this point, more workers with *medium* reputation standing have been included into the task allocation plan computed by GeoTruCrowd. The average error rate of GeoTruCrowd increases significantly to around 30%.

Figure 4 depicts the average budget utilization rates of the three approaches. As CrowdBudget requires pre-determination of the size of the crowd and it searches for workers around the whole city instead of within the task regions, its budget utilization rates are high and remain relatively stable under different experiment settings. This is because in city-wide searches, it is usually possible to find enough workers with high reputation to fully utilize the budget. For GeoTruCrowd and Budget-TASC, the average budget utilization rates increase with increasing task radius and decrease with increasing budget size. Among the three approaches, Budget-TASC achieves the lowest average budget utilization rate.



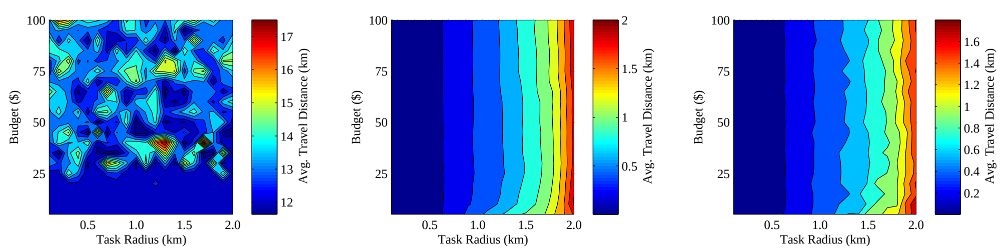
(a) CrowdBudget (b) GeoTruCrowd (c) Budget-TASC

Figure 3: Experiment Results – average estimation error rate (e).



(a) CrowdBudget (b) GeoTruCrowd (c) Budget-TASC

Figure 4: Experiment Results – average budget utilization rate ($B^{(-)}$).



(a) CrowdBudget (b) GeoTruCrowd (c) Budget-TASC

Figure 5: Experiment Results – average distance travelled (D).

Figure 5 depicts the average distance travelled by workers under the three approaches. Since CrowdBudget does not consider the distance between workers and tasks and only looks for the most reputable workers, the average distance the selected workers need to travel is generally high under different experiment settings. As CrowdBudget searches among all known workers, its performance in terms of this metric is not affected by either the budget size or the task radius. The jaggedness in Figure 5(a) is due to the changing relative spatial locations of tasks and selected workers in each experiment since workers' honesty values are regenerated for different experiment settings. With the help of task radius limits, GeoTruCrowd and Budget-TASC can both achieve low average travel distances under the experiment settings studied. The average travel distances of these two approaches increase with increasing task radius.

5.3. Discussions

By taking the average of the e, $B^{(-)}$, and D values over all experimental settings, we obtain the overall performance of the three approaches in terms of their $\overline{e}$, $\overline{B^{(-)}}$, and $\overline{D}$ values as shown in Table 2. In terms of the overall average error rate $\overline{e}$, the proposed Budget-TASC approach achieves a 45.0% reduction compared to GeoTruCrowd, and a 81.0% reduction compared to CrowdBudget. At the same time, Budget-TASC utilizes 17.1% less budget than GeoTruCrowd, and 28.8% less budget than CrowdBudget. In terms of overall average distance travelled per worker, $\overline{D}$, Budget-TASC achieves comparable performance with GeoTruCrowd. These two approaches significantly outperform CrowdBudget in this aspect.

Table 2: Summary of performance

Approach	$\overline{e}$	$\overline{B^{(-)}}$	$\overline{D}$
CrowdBudget	81.1%	96.5%	13.9km
GeoTruCrowd	28.0%	84.8%	0.60km
Budget-TASC	15.4%	67.6%	0.58km

6. Conclusions and Future Work

Existing spatial crowdsourcing platforms mostly adopt a passive model of task distribution. After a spatial task is posted by a task requester, the platform simply advertises this task publicly to all registered workers and waits for workers to respond. In this paper, we formulated the problem of spatial crowdsourcing task allocation with budget constraints as a Multiple Choice Knapsack Problem, and proposed, Budget-TASC, a greedy algorithm that efficiently solves the spatial crowdsourcing MCKP in $O(N^2)$ time by leveraging on the heuristics related to workers' locations and reputation. Budget-TASC eliminates the need for task requesters to pre-specify the number of workers required for a task, thereby simplifying the user experience. Furthermore, it is able to support spatial crowdsourcing systems to take workers' past performance into account when determining the sizes of their rewards. Such a reward scheme can serve as an incentive for workers to try to provide high quality results consistently over time.

We have shown that Budget-TASC is able to find one of the available optimal solutions in $O(N^2)$ for a given task and a set of workers. Through extensive simulations based on real-world data from *Foursquare*, we demonstrated that the proposed approach outperforms state-of-the-art approaches by achieving over 45% reduction in the average error rate, and results in more than 15% of savings on the budget. Our work forms a basis for real-world spatial crowdsourcing systems to handle paid tasks.

Budget-TASC can be implemented as a automatic decision support mechanism in a mobile crowdsourcing system that helps requesters to allocate spatial tasks automatically. With information about worker performance and location, Budget-TASC computes task allocation plans that not only stays within the specified budget, but also improves the task requesters chances of obtaining high quality results subject to the physical limitations of the task locations and worker pool.

The worker reward model used in this paper supports two levels of non-zero reward for workers with different reputations. In some applications, more fine-grained reputation levels as a basis for determining worker rewards may be needed to provide strong incentives for workers to acquire and maintain good reputation standing. The extension of our results to such cases is a topic for future investigation.

Acknowledgements

This research is supported, in part, by the National Research Foundation Singapore under its Interactive Digital Media (IDM) Strategic Research Programme; and the Lee Kuan Yew Post-Doctoral Fellowship Grant. The authors would like to express their gratitude towards the editor and the reviewers for their help in improving the paper.

References

[1] A. Doan, R. Ramakrishnan, A. Y. Halevy, Crowdsourcing systems on the world-wide web, Communications of the ACM 54 (4) (2011) 86–96.

[2] P. Michelucci, J. L. Dickinson, The power of crowds, Science 351 (6268) (2016) 32–33.

[3] P. G. Ipeirotis, Analyzing the Amazon Mechanical Turk marketplace, XRDS: Crossroads, The ACM Magazine for Students 17 (2) (2010) 16–21.

[4] M. Venanzi, A. Rogers, N. R. Jennings, Crowdsourcing spatial phenomena using trust-based heteroskedastic gaussian processes, in: Proceedings of the 1st AAAI Conference on Human Computation and Crowdsourcing (HCOMP-13), 2013, pp. 182–189.

[5] G. Pickard, W. Pan, I. Rahwan, M. Cebrian, R. Crane, A. Madan, A. Pentland, Time-critical social mobilization, Science 334 (6055) (2011) 509–512.

[6] A. Rutherford, M. Cebrian, S. Dsouza, E. Moro, A. Pentland, I. Rahwan, Limits of social mobilization, Proceedings of the National Academy of Sciences of the United States of America (PNAS) 110 (16) (2013) 6281–6286.

[7] F. Alt, A. S. Shirazi, A. Schmidt, U. Kramer, Z. 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), 2010, pp. 13–22.

[8] L. Kazemi, C. Shahabi, L. Chen, GeoTruCrowd: Trustworthy query answering with spatial crowdsourcing, in: Proceedings of the 21th International Conference on Advances in Geographic Information Systems (ACM SIGSPATIAL'13), 2013, pp. 304–313.

[9] A. Jøsang, R. Ismail, C. Boyd, A survey of trust and reputation systems for online service provision, Decision Support Systems 43 (2) (2007) 618–644.

[10] H. Yu, Z. Shen, C. Miao, C. Leung, D. Niyato, A survey of trust and reputation management systems in wireless communications, Proceedings of the IEEE 98 (10) (2010) 1755–1772.

[11] Z. Shen, H. Yu, C. Miao, J. Weng, Trust-based web-service selection in virtual communities, Journal for Web Intelligence and Agent Systems (WIAS) 9 (3) (2011) 227–238.

[12] H. Yu, S. Liu, A. C. Kot, C. Miao, C. Leung, Dynamic witness selection for trustworthy distributed cooperative sensing in cognitive radio networks, in: Proceedings of the 13th IEEE International Conference on Communication Technology (ICCT'11), 2011, pp. 1–6.

[13] H. Yu, Z. Shen, C. Leung, C. Miao, V. R. Lesser, A survey of multi-agent trust management systems, IEEE Access 1 (1) (2013) 35–50.

[14] P. Sinha, A. A. Zoltners, The multiple-choice knapsack problem, Operations Research 27 (3) (1979) 503–515.

[15] A. Kittur, J. V. Nickerson, M. Bernstein, E. Gerber, A. Shaw, J. Zimmerman, M. Lease, J. Horton, The future of crowd work, in: Proceed-

ings of the 2013 Conference on Computer Supported Cooperative Work (CSCW'13), 2013, pp. 1301–1318.

[16] Y. Sakurai, T. Okimoto, M. Oka, M. Shinoda, M. Yokoo, Ability grouping of crowd workers via reward discrimination, in: Proceedings of the 1st AAAI Conference on Human Computation and Crowdsourcing (HCOMP'13), 2013, pp. 147–155.

[17] Y. Liu, J. Zhang, H. Yu, C. Miao, Reputation-aware continuous double auction, in: Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI-14), 2014, pp. 3126–3127.

[18] H. Yu, Z. Shen, C. Miao, B. An, Challenges and opportunities for trust management in crowdsourcing, in: Proceeding of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'12), 2012, pp. 486–493.

[19] H. Yu, Z. Shen, C. Miao, B. An, A reputation-aware decision-making approach for improving the efficiency of crowdsourcing systems, in: Proceedings of the 12th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'13), 2013, pp. 1315–1316.

[20] H. Yu, C. Miao, B. An, C. Leung, V. R. Lesser, A reputation management model for resource constrained trustee agents, in: Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13), 2013, pp. 418–424.

[21] H. Yu, C. Miao, B. An, Z. Shen, C. Leung, Reputation-aware task allocation for human trustees, in: Proceedings of the 13th International Con-

ference on Autonomous Agents and Multiagent Systems (AAMAS'14), 2014, pp. 357–364.

[22] H. Yu, X. Yu, S. F. Lim, J. Lin, Z. Shen, C. Miao, A multi-agent game for studying human decision-making, in: Proceedings of the 13th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS'14), 2014, pp. 1661–1662.

[23] H. Yu, C. Miao, Z. Shen, C. Leung, Y. Chen, Q. Yang, Efficient task sub-delegation for crowdsourcing, in: Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI-15), 2015, pp. 1305–1311.

[24] Z. Pan, H. Yu, C. Miao, C. Leung, Efficient collaborative crowdsourcing, in: Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI-16), 2016, pp. 4248–4249.

[25] E. Kamar, S. Hacker, E. Horvitz, Combining human and machine intelligence in large-scale crowdsourcing, in: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'12), 2012, pp. 467–474.

[26] Y. Bachrach, T. Graepel, G. Kasneci, M. Kosinski, J. Van Gael, Crowd IQ: Aggregating opinions to boost performance, in: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'12), 2012, pp. 535–542.

[27] L. Jaimes, I. Vergara-Laurens, M. Labrador, A location-based incentive mechanism for participatory sensing systems with budget constraints,

in: Proceedings of the 2012 IEEE International Conference on Pervasive Computing and Communications (PerCom'12), 2012, pp. 103–108.

[28] D. Zhao, X.-Y. Li, H. Ma, How to crowdsource tasks truthfully without sacrificing utility: Online incentive mechanisms with budget constraint, in: Proceedings of the 33rd Annual IEEE International Conference on Computer Communications (INFOCOM'14), 2014, pp. 1213–1221.

[29] L. Tran-Thanh, M. Venanzi, A. Rogers, N. R. Jennings, Efficient budget allocation with accuracy guarantees for crowdsourcing classification tasks, in: Proceedings of the 12th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS'13), 2013, pp. 901–908.

[30] L. Kazemi, C. Shahabi, GeoCrowd: Enabling query answering with spatial crowdsourcing, in: Proceedings of the 20th International Conference on Advances in Geographic Information Systems (ACM SIGSPATIAL'12), 2012, pp. 189–198.

[31] C. Baraniuk, Quake AI plans disaster rescue, New Scientist 228 (3041) (2015) 22.

[32] B. Khosravifar, J. Bentahar, M. Gomrokchi, R. Alam, Crm: An efficient trust and reputation model for agent computing, Knowledge-Based Systems (KBS) 30 (2012) 1–16.

[33] H. Fang, G. Guo, J. Zhang, Multi-faceted trust and distrust prediction for recommender systems, Decision Support Systems 71 (2015) 37–47.

[34] O. A. Wahab, J. Bentahar, H. Otrok, A. Mourad, A survey on trust and reputation models for web services: Single, composite, and communities, Decision Support Systems 74 (2015) 121–134.

[35] H. Yu, Z. Shen, B. An, An adaptive witness selection method for reputation-based trust models, in: Proceedings of the 15th International Conference on Principles and Practice of Multi-agent Systems (PRIMA'12), 2012, pp. 184–198.

[36] H. Yu, Z. Shen, C. Miao, B. An, C. Leung, Filtering trust opinions through reinforcement learning, Decision Support Systems 66 (2014) 102–113.

[37] A. Jøsang, R. Ismail, The beta reputation system, in: Proceedings of 15th Bled Electronic Commerce Conference, 2002, pp. 41–55.

[38] S. Liu, H. Yu, C. Miao, A. C. Kot, A fuzzy logic based reputation model against unfair ratings, in: Proceedings of the 12th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'13), 2013, pp. 821–828.

[39] R. Ireneus Scibor-Marchocki, Spherical trigonometry, Elementary-Geometry Trigonometry web page, 1997.

[40] M. Sarwat, J. Levandoski, A. Eldawy, M. Mokbel, LARS*: An efficient and scalable location-aware recommender system, IEEE Transactions on Knowledge and Data Engineering (TKDE) 26 (6) (2014) 1384–1399.