Answer Inference for Crowdsourcing based Scoring

Name: Kaikai Sheng
Student ID: 5107119012

Abstract—**Crowdsourcing is an effective paradigm in human centric computing for addressing problems by utilizing human computation power. While efforts have been made to study the crowdsourcing systems for labeling tasks such as classification, those for scoring tasks with continuous and correlative answers have not been well studied. In this paper, we propose two inference algorithms, MCE (Maximum Correlation Estimate) and WMCE (Weighted Maximum Correlation Estimate), to infer true answers based on answers submitted by workers. When estimating answers, WMCE algorithm assigns diverse weight to submitted answers of workers based on their quality while MCE algorithm assigns identical weight to submitted answers of all workers. For a fixed worker population, we reveal that the increase in task redundancy[1] can improve accuracy of estimated answers but such improvement is limited within a certain level. We further show that WMCE algorithm can reduce the influence of this limitation better than MCE algorithm for the same crowdsourcing system. Simulation results validate our theoretical analysis and show that WMCE algorithm outperforms MCE algorithm in the accuracy of estimated answers.**

I. INTRODUCTION

Scoring task, with the increasing need for the evaluation of quality, has been drawing much attention in recent years. Different from labeling tasks which have multiple independent labels for workers to choose, answers of scoring tasks are continuous number in an interval and have correlation with each other. Some scoring tasks like scoring papers in online examination systems [1] have been studied. However, the scoring task can only be accomplished by humans, thus making it hard for the system to work efficiently. To improve the performance of the system, we need to find a way to accomplish the tasks more quickly and accurately with lower cost. Now with the boom of Internet, the power of crowd on the Internet can be utilized to fulfill such kind of task, which corresponds to the idea of crowdsourcing (a new paradigm for human centric computing [2]). Some crowdsourcing platforms have been built to solve scoring tasks. Alfaro *et al.* build a tool called "CrowdGrader" to let students submit and collaboratively grade solutions to homework assignments [3]. However, none of the prior work focuses on the theoretical analysis of the crowdsourcing based scoring system. So we are interested in designing an allocating method for the tasks as well as making the theoretical analysis of the crowdsourcing based scoring system, which can be utilized in many application scenarios such as the evaluation of articles, paintings, and movies.

There have already been many typical crowdsourcing systems worldwide [4] such as Amazon Mechanical Turk [5], Yelp [6] and Yahoo! Answers [7], where people can publish tasks on the system platform and collect answers from workers. Through such systems, workers can fulfill scoring tasks to get paid and the task providers can collect a large number of answers submitted by workers to estimate the true answer at a low cost. However, the collected answers are unreliable because the workers' performance can be influenced by many factors such as biological and psychological conditions or biased understandings of tasks. In order to get reliable answers from workers, task redundancy and inference algorithms shall be utilized. The estimated answer is then derived from all answers submitted by workers. Nevertheless, such scheme is confronted with the following two challenges: (1) How the task redundancy influences on the accuracy of estimated answers? (2) What inference algorithm shall be utilized to infer task answers with higher accurate level?

In labeling tasks, majority voting [8], which chooses what the majority of workers agree on, is a straightforward and widely-used inference algorithm to estimate answers from multiple workers' responses. Inspired by the majority voting algorithm, based on characteristics of the scoring task, we proposed a fast and straightforward algorithm called MCE (Maximum Correlation Estimate) algorithm to do the answer inference for scoring tasks.

However, both of majority voting and MCE have a vulnerable output, which can be easily influenced by the noisy answers from low quality workers. To cope with that, we need to separate the low quality workers from high quality workers. Now many researchers are focusing on estimating task answers by estimating the reliability of workers and many algorithms have been proved to be effective in the labeling tasks. Karger *et al.* propose a task allocation method based on random regular bipartite graph and use low rank approximation to generate the estimated answer [9]. Moreover, they also develop an efficient crowdsourcing system in minimizing the task assignment redundancy while achieving a desirable reliability. Ghosh *et al.* apply eigenvalue decomposition method to homogeneous-task model [10] and Ho *et al.* generalize the model to be heterogeneous [11]. However, the schemes proposed are mainly for labeling tasks, and are unsuitable to be applied to scoring tasks, because the answers in the latter could be correlative with each other. Inspired by the iterative algorithm proposed by Karger *et al.* [12], we design the WMCE (Weighted Maximum Correlation Estimate) algorithm for scoring tasks, which infers task answers based on not only the worker quality but also the answers' correlations with each other. We consider the average degree of approximation between estimated answers and true answers as the performance metric of the crowdsourcing system.

[1] In this paper, the term redundancy indicates the number of workers allocated to a task.

In this paper, we investigate on inference algorithms of task answers and the relationship between task redundancy and accuracy of estimated answers in the crowdsourcing system based scoring. Specifically, we have the following two-fold contributions.

- We propose two inference algorithms, MCE and WMCE. WMCE algorithm which assigns diverse weights to workers' submitted answers based on their quality can outperform the MCE algorithm which assigns identical weights to the submitted answers.
- We reveal that the increase in task redundancy can improve the accuracy of estimated answers but such improvement is limited within a certain level which is determined by the worker quality and the inference algorithm.

II. SYSTEM MODEL AND PROBLEM FORMULATION

A. System model

In the crowdsourcing based scoring system, tasks are assigned to workers by a server. Workers give their evaluated score[2] for the assigned tasks and submit their scores to the server. After that the server utilizes a certain inference algorithm to estimate the task answer from all submitted answers. The design goal of such system is to let the estimated answer be as close to the true answer as possible.

Suppose that there are m scoring tasks whose scoring range is in a continuous interval $[a, b]$. These tasks can compose a task set denoted by $T = \{t_i\}_{i\in[m]}$[3], where the element $t_i \in [a, b]$ represents the true answer of task i. True answers of tasks are drawn from a distribution denoted by $\mathcal{D}$. These m tasks are assigned to n workers which compose a worker set $W = \{w_j\}_{j\in[n]}$. If the true answer of a task is y, the probability density of worker j with submitted answer x is denoted by the probability density function $f_j(x|y)$, where $\int_a^b f_j(x|y)\, dx = 1$ for any $y \in [a, b]$ and $j \in [n]$. The probability density functions are drawn from a distribution denoted by $\mathcal{Q}$.

B. Problem formulation

For scoring tasks with an continuous score interval in $[a, b]$, we define a correlation function to describe the proximity (correlation) of two scores are with each other in as the equation (1) shows. For the simplicity of mathematical analysis, we use the square form $(\cdot)^2$ instead of the absolute value sign form $|\cdot|$ to profile the correlation.

$$R(x, y) \triangleq \begin{cases} 1 - c(x-y)^2 & if\, |x-y| < \frac{1}{\sqrt{c}}\ , \\ 0 & otherwise\ , \end{cases} \tag{1}$$

where c is the constant. Intuitively, if the difference between x and y, the value of the correlation function is large, which indicates that they have high correlation with each other. Apparently, the true score is the most correlative one with itself. If the difference between two scores is large enough, the value of the correlation function will be 0 since they are not correlative with each other.

We use a regular random bipartite graph $G = (T \cup W, E)$ to model the task allocation. The edge set E represents the task assignment, where $(t_i, w_j) \in E$ if task t_i is assigned to worker w_j. The degree of nodes in T is l, which indicates that each task is randomly assigned to l different tasks. The degree of nodes in W is r, which means that each task is randomly assigned to r different workers. The parameters m, n, l, r are subject to the equation $ml = nr$ according to the property of bipartite graph. In this allocation scheme, all tasks are allocated simultaneously and answers are collected from workers. These answers compose the answer matrix $A = [A_{ij}]^{m\times n}$, where the matrix entity $A_{ij} \in \{[a, b] \cup null\}$. $A_{ij} = null$ if task t_i is not assigned to worker w_j.

In order to describe the performance or accuracy of the system, we define the *average correlation* between estimated answers and true answers.

$$\bar{R} \triangleq \frac{1}{m}\sum_{i=1}^{m} R\left(\hat{t}_i, t_i\right)\ , \tag{2}$$

where t_i is the true answer of task i and $\hat{t}_i$ is the estimated answer of task i. To illustrate the accuracy of the system more obviously when $\bar{R}$ is close to 1, we define correlation error E_c, which is very similar to the term *bit error rate* in communication principle

$$E_c \triangleq 1 - \bar{R}\ . \tag{3}$$

To clarify the analysis in the following sections, we also give definitions of some derivative parameters. We define $\tilde{q}_{j|y}$ as

$$\tilde{q}_{j|y} \triangleq \int_a^b f_j(x|y)\, R(x, y)\, dx, \tag{4}$$

which can be interpreted as the mathematical expectation of correlation between the the task true answer y and the answer of worker j. However, in this definition, $\tilde{q}_{j|y}$ of workers who give the answers randomly is still larger than 0. Hence, we need to add bias to the correlation function to let $\tilde{q}_{j|y} = 0$ if worker j gives answers randomly. The *biased correlation function* $\tilde{R}(x, y)$ is defined as

$$\tilde{R}(x, y) \triangleq R(x, y) - \frac{\int_a^b R(x, y)\, dy}{b-a}, \tag{5}$$

which satisfies $\int_a^b \tilde{R}(x, y)\, dx = 0$. Then the quality of worker j when the true answer of task is y is defined as

$$q_{j|y} \triangleq \int_a^b f_j(x|y)\, \tilde{R}(x, y)\, dx\ . \tag{6}$$

Based on $q_{j|y}$, the quality of worker j is defined as[4]

$$q_j \triangleq \frac{1}{|\partial j|}\sum_{i\in\partial j} q_{j|t_i}\ . \tag{7}$$

[2] Throughout this paper, we will use "score" and "answer" interchangeably

[3] Throughout this paper, we use notation $[N]$ to denote the set $\{1, 2, \ldots, N\}$.

[4] Throughout the paper, we use ∂i to denote the worker set which is assigned to task i and use ∂j to denote the tasks set which is allocated to worker j.

Intuitively, if q_j is large, it indicates that the worker j is reliable. If he/she gives answers randomly, $q_j = 0$.

When the true answer of the task is y, we define $b_{j|y}$, which is the bias of the answer of work j, as

$$b_{j|y} \triangleq \int_a^b (x-y) f_j\left(x|y\right) dx \ . \tag{8}$$

Similarly, we define $v_{j|y}$, which is the variance of the answer of work j, as

$$v_{j|y} \triangleq \int_a^b \left[x - \left(y + b_{j|y}\right)\right]^2 f_j\left(x|y\right) dx \ . \tag{9}$$

III. INFERENCE ALGORITHM

In this section, we introduce two inference algorithms, MCE and WMCE. Different from inference algorithms in crowdsourcing based labeling system like the iterative algorithm proposed in [12], the following two inference algorithms can utilize the correlation between workers' answers and true answers even though they are not exactly equal to each other.

A. MCE Algorithm

In labeling tasks, majority voting, which chooses what the majority of workers agree on, is a straightforward approach to estimate answers from multiple workers' responses. In majority voting, the voting weight of each worker is identical. For scoring tasks, we propose the MCE algorithm as following by integrating the characteristics of maximum likelihood estimation and majority voting.

Algorithm 1 MCE Algorithm

```
Input:
    The answer matrix $A = [A_{ij}]^{m \times n}$
    The correlation parameter function $\tilde{R}(x, y)$
Output:
    The estimated answer vector $\hat{t} = \{\hat{t}_i\}_{i \in [m]}$
1: for $i = 1, 2, ..., m$ do
2:     $\hat{t}_i = \arg\max_{x \in [a,b]} \sum_{j \in \partial i} \tilde{R}(x, A_{ij})$;
3: end for
4: return $\hat{t}$;
```

As is shown in line 2, based on the maximum likelihood estimation, MCE algorithm chooses the task estimated answer x which maximizes the sum of correlation function's value $\tilde{R}(x, y_j)$, where y_j is the answer submitted by worker j. Similar to majority voting, this algorithm gives identical weight to each allocated worker's answers when choosing the estimated answers. Different from majority voting, this algorithm can utilize the correlation between answers, which indicates that this algorithm can estimate answers more accurately than majority voting for scoring tasks.

B. WMCE Algorithm

When quality of workers is divergent, MCE algorithm is error-prone since it gives identical weight to each worker's answer. In order to estimate task answers accurately, we can use workers' quality as their weight while to estimate workers' quality answers precisely, we shall use true answers of tasks. However, neither workers' quality nor task answer is prior known. To resolve such a challenge, we propose the WMCE algorithm which can estimate both the worker quality and task answers by iterations at the same time. At each iteration, the algorithm will update new estimated answer vector $\hat{t}^{(h)} = \left[\hat{t}_i^{(h)}\right]_{i \in [m]}$ and new estimated quality vector $\hat{q}^{(h)} = \left[\hat{q}_j^{(h)}\right]_{j \in [n]}$, where the superscript (h) denotes the value of variables in the h_{th} iteration.

Algorithm 2 WMCE Algorithm

```
Input:
    The answer matrix $A = [A_{ij}]^{m \times n}$
    The maximum iteration number $h_{max}$
    The biased correlation function $\tilde{R}(x, y)$
Output:
    The estimated answer vector $\hat{t}$
Initialize the estimated qualtiy vector $\hat{q}^{(1)} = \mathbf{1}^n$, where
    $\mathbf{1}^n$ denotes the all-ones vector in n-dimensional.
for $h = 1, 2, ..., h_{max}$ do
    for $i = 1, 2, ..., m$ do
        $\hat{t}_i^{(h)} = \arg\max_{x \in [a,b]} \sum_{j \in \partial i} \hat{q}_j^{(h)} \tilde{R}\left(x, A_{ij}\right)$;
    end for
    for $j = 1, 2, ..., n$ do
        $\hat{q}_j^{(h+1)} = \frac{1}{r} \sum_{i \in \partial j} \tilde{R}(\hat{t}_i^{(h)}, A_{ij})$;
    end for
end for
The estimated answer vector $\hat{t} = \hat{t}^{(h_{max})}$
return $\hat{t}$;
```

During each iteration, the algorithm updates the new estimated answer x which maximums the sum of correlation function's value $\tilde{R}(x, y_j)$ based on the previous estimated quality of worker j, where y_j is the answer submitted by worker j. After that, it updates the estimated workers' quality based on the previous estimated answer.

IV. THEORETICAL ANALYSIS

In this section, we show the main theoretical results. For the simplicity of mathematical analysis, we assume that the difference between worker's answer and true answer is not large. This assumption can be formulated as: $\forall y \in [a, b]$ and $\forall j \in [n]$, $f_j\left(x|y\right) = 0$ when $|x - y| > \frac{1}{\sqrt{c}}$. Given worker quality, answer bias and answer variance, we obtain the mathematical expectation of correlation error E_c for two inference algorithms. Based on the mathematical expectation of E_c, we show that reducing the correlation error by increasing task redundancy is limited within a certain level.

Theorem 1. *For $\forall y \in [a, b]$ and $\forall j \in [n]$, if $f_j\left(x|y\right) = 0$ when $|x - y| > \frac{1}{\sqrt{c}}$, the mathematical expectation of correlation* *error E_c can be expressed as*[5]

$$\mathbb{E}\left[\mathbf{E}_c\right] = \frac{1}{m}\sum_{i\in[m]} \mathbb{E}\left[\mathbf{E}_{c,i}\right], \tag{10}$$

where $\mathbb{E}\left[\mathbf{E}_{c,i}\right]$ is the mathematical expectation of correlation error for i_{th} task. $\mathbb{E}\left[E_c\right]$ can be expressed as

$$\mathbb{E}\left[E_c\right] = X + Y \ , \tag{11}$$

where

$$\begin{cases} X = c\Big(\sum\limits_{j\in\partial i} \tilde{w}_{l,j} b_{j|t_i}\Big)^2 \\ Y = c\sum\limits_{j\in\partial i} \tilde{w}_{l,j}^2 v_{j|t_i} \end{cases} . \tag{12}$$

The normalized worker weight satisfies $\sum\limits_{j\in\partial i}\tilde{w}_{l,j} = 1$, where $\tilde{w}_{l,j} = \frac{1}{l}$ for $\forall j \in \partial i$ in MCE algorithm while $\tilde{w}_{l,j} = \frac{q_j}{\sum\limits_{j\in\partial i} q_j}$ for $\forall j \in \partial i$ in WMCE algorithm.

Proof. It is obvious that we can derive $\mathbb{E}\left[\mathbf{E}_c\right]$ when given $\mathbb{E}\left[E_c\right]$. So we will mainly prove how to obtain $\mathbb{E}\left[E_c\right]$ in the following part.

Since $f_j\left(x|y\right) = 0$ when $|x-y| > \frac{1}{\sqrt{c}}$, when WMCE algorithm finally converges, we can obtain the estimated answer which maximizes the polynomial in line 4 by calculating the first-order derivative of this polynomial. The estimated answer can be expressed as

$$\hat{\mathbf{t}}_i = \sum_{j\in\partial i} \tilde{w}_{l,j}\mathbf{A}_{ij} \ , \tag{13}$$

where $\tilde{w}_{l,j} = \frac{q_j}{\sum\limits_{j\in\partial i} q_j}$. For MCE algorithm, to obtain $\hat{\mathbf{t}}_i$, we need to let $\tilde{w}_{l,j} = \frac{1}{l}$.

Hence, $\mathbb{E}\left[\mathbf{E}_{c,y}\right]$ can be derived as[6]

$$\begin{aligned} \mathbb{E}\Big[\mathbf{E}_{c,y}\Big] &= \mathbb{E}\left[1 - R\left(\hat{\mathbf{t}}_i, t_i\right)\right] = c\mathbb{E}\Big[\left(\hat{\mathbf{t}}_i - t_i\right)^2\Big] \\ &= c\mathbb{E}^2\left[\left(\hat{\mathbf{t}}_i - t_i\right)\right] + c\mathbb{D}\left[\left(\hat{\mathbf{t}}_i - t_i\right)\right] \ . \end{aligned} \tag{14}$$

To lighten the formula, let $X = c\mathbb{E}^2\left[\left(\hat{\mathbf{t}}_i - t_i\right)\right]$, $Y = c\mathbb{D}\left[\left(\hat{\mathbf{t}}_i - t_i\right)\right]$.

For X, substituting (13) into X, we have

$$\begin{aligned} X &= c\mathbb{E}^2\Big[\left(\hat{\mathbf{t}}_i - t_i\right)\Big] = c\mathbb{E}^2\Big[\big(-t_i + \sum_{j\in\partial i}\tilde{w}_{l,j}\mathbf{A}_{ij}\big)\Big] \\ &= c\mathbb{E}^2\Big[\big(\sum_{j\in\partial i}\tilde{w}_{l,j}\left(\mathbf{A}_{ij} - t_i\right)\big)\Big] \\ &= c\Big(\sum_{j\in\partial i}\tilde{w}_{l,j}\mathbb{E}\left[\mathbf{A}_{ij} - t_i\right]\Big)^2 = c\Big(\sum_{j\in\partial i}\tilde{w}_{l,j} b_{j|t_i}\Big)^2 \ . \end{aligned} \tag{15}$$

For Y, substituting (13) into Y, we have

$$\begin{aligned} Y &= c\mathbb{D}\Big[\left(\hat{\mathbf{t}}_i - t_i\right)\Big] = c\mathbb{D}\Big[\big(-t_i + \sum_{j\in\partial i}\tilde{w}_{l,j}\mathbf{A}_{ij}\big)\Big] \\ &= c\mathbb{D}\Big[\big(\sum_{j\in\partial i}\tilde{w}_{l,j}\mathbf{A}_{ij}\big)\Big] = c\sum_{j\in\partial i}\mathbb{D}\Big[\tilde{w}_{l,j}\mathbf{A}_{ij}\Big] \\ &= c\sum_{j\in\partial i}\tilde{w}_{l,j}^2\mathbb{D}\Big[\mathbf{A}_{ij}\Big] = c\sum_{j\in\partial i}\tilde{w}_{l,j}^2 v_{j|t_i} \ . \end{aligned} \tag{16}$$

Hence, $\mathbb{E}\left[E_c\right]$ can be expressed as

$$\begin{aligned} \mathbb{E}\left[\mathbf{E}_{c,i}\right] &= X + Y \\ &= c\big(\sum_{j\in\partial i}\tilde{w}_{l,j} b_{j|t_i}\big)^2 + c\sum_{j\in\partial i}\tilde{w}_{l,j}^2 v_{j|t_i} \ . \end{aligned} \tag{17}$$

This finishes the proof of Theorem 1. □

Based on Theorem 1, we show that the mathematical expectation of E_c is above the system bias which is determined by the answer bias and inference algorithm.

Corollary 1. ***(System Bias)*** *If $f_j\left(x|y\right) = 0$ when $|x-y| > \frac{1}{\sqrt{c}}$, for fixed worker population, $\mathbb{E}\left[\mathbf{E}_c\right]$ can be reduced by increasing the task redundancy but it has to satisfy the following inequality*

$$\mathbb{E}\left[\mathbf{E}_c\right] > \frac{c}{m}\sum_{i\in[m]}\Big(\sum_{j\in[n]}\tilde{w}_{n,j} b_{j|t_i}\Big)^2 \ , \tag{18}$$

where $\tilde{w}_{n,j} = \frac{1}{n}$ for $\forall j \in [n]$ in MCE algorithm while $\tilde{w}_{n,j} = \frac{q_j}{\sum\limits_{j\in[n]} q_j}$ for $\forall j \in [n]$ in WMCE algorithm. The right term in (18) *is defined as the system bias.*

Proof. Substituting (11) and (12) into (10), we have

$$\begin{aligned} \mathbb{E}\Big[\mathbf{E}_c\Big] &= \frac{1}{m}\sum_{i\in[m]}\mathbb{E}\Big[\mathbf{E}_{c,i}\Big] \\ &= \frac{1}{m}\sum_{i\in[m]} c\Big(\sum_{j\in\partial i}\tilde{w}_{l,j} b_{j|t_i}\Big)^2 + \frac{1}{m}\sum_{i\in[m]}\Big(c\sum_{j\in\partial i}\tilde{w}_{l,j}^2 v_{j|t_i}\Big) \end{aligned} \tag{19}$$

Recall that each task is allocated to l different workers randomly, which indicates that for $\forall i \in [m]$, the probability density functions of workers who are assigned to task i is drawn from the same distribution $\mathcal{Q}$ as those of n workers. Hence, the derived parameters q_j, $b_{j|y}$ for $\forall i \in [m]$ and $\forall j \in \partial i$ have the same distributions as those for $\forall j \in [n]$. So (19) can be derived as

$$\begin{aligned} \mathbb{E}\Big[\mathbf{E}_c\Big] &= \frac{c}{m}\sum_{i\in[m]}\Big(\sum_{j\in\partial i}\tilde{w}_{l,j} b_{j|t_i}\Big)^2 + \frac{c}{m}\sum_{i\in[m]}\Big(\sum_{j\in\partial i}\tilde{w}_{l,j}^2 v_{j|t_i}\Big) \\ &= \frac{c}{m}\sum_{i\in[m]}\Big(\sum_{j\in[n]}\tilde{w}_{n,j} b_{j|t_i}\Big)^2 + \frac{c}{m}\sum_{i\in[m]}\Big(\frac{n^2}{l^2}\sum_{j\in\partial i}\tilde{w}_{n,j}^2 v_{j|t_i}\Big) \\ &= \frac{c}{m}\sum_{i\in[m]}\Big(\sum_{j\in[n]}\tilde{w}_{n,j} b_{j|t_i}\Big)^2 + \frac{c}{m}\sum_{i\in[m]}\Big(\frac{n}{l}\sum_{j\in[n]}\tilde{w}_{n,j}^2 v_{j|t_i}\Big) \\ &= \frac{c}{m}\sum_{i\in[m]}\Big(\sum_{j\in[n]}\tilde{w}_{n,j} b_{j|t_i}\Big)^2 + \frac{cn}{ml}\sum_{i\in[m]}\Big(\sum_{j\in[n]}\tilde{w}_{n,j}^2 v_{j|t_i}\Big) \ . \end{aligned} \tag{20}$$

[5] $\mathbb{E}\left[\cdot\right]$ is a notation of mathematical expectation. Throughout this paper, we use boldface characters to denote random variables.

[6] $\mathbb{D}\left[\cdot\right]$ is a notation of variance.

Hence we have

$$\mathbb{E}\left[\mathbf{E}_c\right] > \frac{c}{m}\sum_{i\in[m]}\left(\sum_{j\in[n]}\tilde{w}_{n,j}b_{j|t_i}\right)^2 . \tag{21}$$

This finishes the proof of Corollary 1. □

In a fixed worker population, for the right term of equation (20), $\tilde{w}^2_{n,j}v_{j|t_i}$ can scale as $\Theta\left(\frac{1}{n^2}\right)$ for any $j\in[n]$, so $\sum\limits_{j\in[n]}\tilde{w}^2_{n,j}v_{j|t_i}$ can scale as $\Theta\left(\frac{1}{n}\right)$. Under the the constraint condition $l\le n$, when l increases up to the scale $\Theta\left(n\right)$, the right term of equation (20) can scale as $\Theta\left(\frac{1}{n}\right)$. In this case, the system bias dominantly determines $\mathbb{E}\left[\mathbf{E}_c\right]$ or the accuracy of estimated answers and the increase in task redundancy almost cannot improve the the accuracy of estimated answers.

Considering the complexity of expression for system bias in (18), we will show some intuitive understandings about why WMCE algorithm outperforms MCE algorithm in the accuracy of estimated answers. Since high quality workers usually have low answer bias, for the term $\tilde{w}_{n,j}b_{j|t_i}$, larger answer bias $b_{j|t_i}$ matches with smaller answer weight $\tilde{w}_{n,j}$ for WMCE algorithm, while larger answer bias $b_{j|t_i}$ always matches with the same answer weight $\frac{1}{n}$ for MCE algorithm. Hence, compared to WMCE algorithm, MCE algorithm will amplify the answer bias of workers which results in larger system bias. So WMCE algorithm outperforms MCE algorithm in the accuracy of estimated answers.

V. SIMULATION RESULTS

In this section, we compare the performance of two algorithms in two circumstances where distribution of worker quality is convergent or divergent. We consider the quality distribution is convergent when any worker from the worker population satisfy the assumption in section IV: for $\forall y\in[a,b]$ and $\forall j\in[n]$, $f_j\left(x|y\right)=0$ when $|x-y|>\frac{1}{\sqrt{c}}$ because in this case, the difference between the maximum quality and minimum quality is not large. We show that when worker quality is convergent, the increase of redundancy can improve the performance of both algorithms and such improvement is indeed limited by the theoretical system bias. In this case, the performance of WMCE algorithm is not much better than that of MCE algorithm. When worker quality is divergent, WMCE algorithm can outperform MCE algorithm much better. In simulation, we set the correlation function in $[0,10]$ with $c=0.25$. We create $m=1000$ tasks and $n=1000$ workers, which indicates that $l=r$ for any task redundancy l. For simplicity, the worker population has two types, which are high quality and low quality.

A. Convergent quality distribution

The parameters of high quality worker and low quality worker are set in Table I.

The comparison of the two algorithms is shown in Figure 1. For both algorithms, as task redundancy increases, the correlation error E_c decreases. When task redundancy is a bit large, both algorithms approach their corresponding theoretical system bias and the correlation error E_c is almost unchanged with the increase of task redundancy. In this circumstance, the performance of WMCE algorithm is not much better than that of MCE algorithm.

TABLE I: Parameters of two worker types

	High quality	Low quality
Ratio	50%	50%
Quality	0.3188	0.1095
Bias	0.1	0.3881

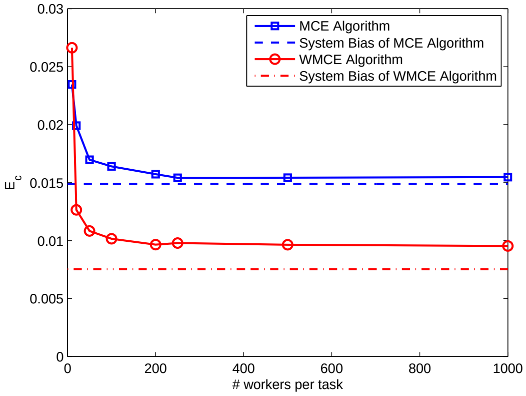

Fig. 1: Comparison of two algorithms

Figure 2 shows the estimated worker quality in descending order for WMCE algorithm when it finally converges. It is obvious that there is a sharp falling edge when the worker index is 500, which indicates that WMCE algorithm can separate high quality workers and low quality workers clearly. This result corresponds to the initial setting, where high quality workers account for 50% of worker population.

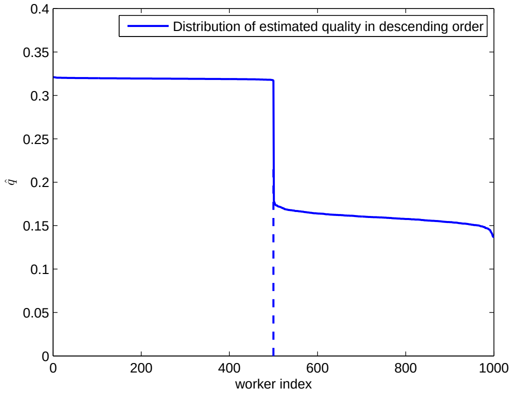

Fig. 2: Estimated quality in descending order

B. Divergent quality distribution

The parameters of high quality worker and low quality worker are set in Table II.

For the divergent quality distribution of worker population, the performance comparison of the two algorithms is shown

TABLE II: Parameters of two worker types

	High quality	Low quality
Ratio	30%	70%
Quality	0.7684	0.0404
Bias	0.1	3

in Figure 3. From Figure 3, we can see that WMCE algorithm outperforms MCE algorithm much better. This is because WMCE algorithm can identify the quality of workers and assign the answer weight based on workers' quality. In contrary, MCE algorithm gives identical weight to every workers, which cannot avoid the influence of noisy answers. So the divergence in quality distribution can make WMCE algorithm select high quality from worker population. This is the reason that WMCE algorithm can outperform MCE much better for the convergent quality distribution than that for the divergent quality distribution. The Figure 4 shows the distribution of estimated worker quality in descending order. From this figure, WMCE algorithm can estimate the worker quality precisely based on their submitted answers.

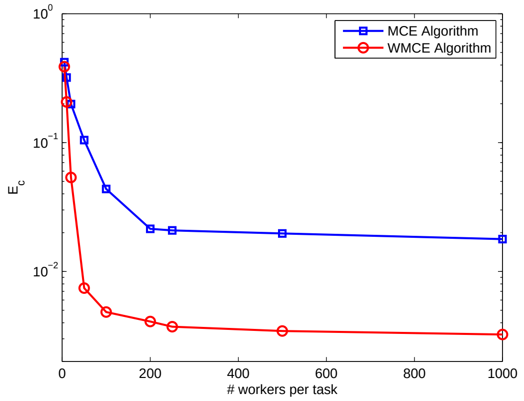

Fig. 3: Estimated quality in descending order

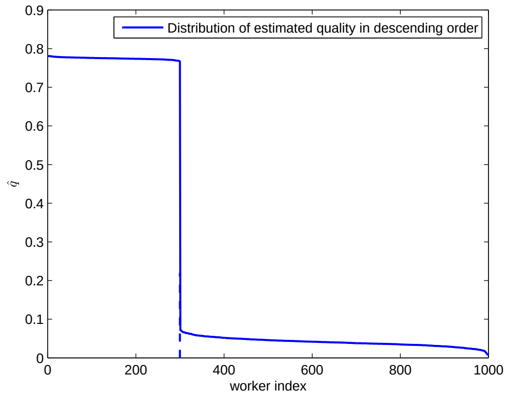

Fig. 4: Estimated quality in descending order

VI. CONCLUSION

In this paper, we have investigated on the crowdsourcing system for scoring tasks whose answers are continuous and correlative with each other. We have considered the question of inference algorithms and the relationship between task redundancy and correlation error of estimated answer. We have proposed two inference algorithms, MCE and WMCE. MCE algorithm assigns identical answer weight to all workers while WMCE algorithm assigns diverse answer weight to workers based on their quality. We have proved that increase of redundancy can decrease correlation error but the correlation error cannot be smaller than the system bias for both algorithms. We have remarked that the WMCE algorithm has smaller system bias than MCE algorithm, which indicates that WMCE algorithm outperforms MCE algorithm in the accuracy of estimated answers. Simulation results have validated out theoretical analysis and shown that WMCE algorithm can outperform MCE algorithm, especially when the worker quality distribution is divergent.

REFERENCES

[1] P. Guo, Z. Hai-yan, and D. Yi-wen, "The design and implementation of online scoring system," in *IT in Medicine & Education, 2009. ITIME'09. IEEE International Symposium on*, vol. 1. IEEE, 2009, pp. 606–610.

[2] R. Kling and S. L. Star, "Human-centered systems in the perspective of organizational and social informatics," *ACM SIGCAS Computers and Society*, vol. 28, no. 1, pp. 22–29, 1998.

[3] L. de Alfaro and M. Shavlovsky, "Crowdgrader: Crowdsourcing the evaluation of homework assignments," *arXiv preprint arXiv:1308.5273*, 2013.

[4] A. Doan, R. Ramakrishnan, and A. Y. Halevy, "Crowdsourcing systems on the world-wide web," *Communications of the ACM*, vol. 54, no. 4, pp. 86–96, 2011.

[5] A. Kittur, E. H. Chi, and B. Suh, "Crowdsourcing user studies with mechanical turk," in *Proc. 26th Annual SIGCHI conf. on human factors in computing systems*, 2008.

[6] J. Aasman, "Social network analysis and geotemporal reasoning in a web 3.0 world," *Int'l Conf. on Computational Science and Engineering*, vol. 4, pp. 546–548, 2009.

[7] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman, "Knowledge sharing and yahoo answers: everyone knows something," in *Proc. 17th int'l conf. on World Wide Web*, 2008.

[8] N. Littlestone and M. K. Warmuth, "The weighted majority algorithm," *Information and computation*, vol. 108, no. 2, pp. 212–261, 1994.

[9] D. R. Karger, S. Oh, and D. Shah, "Efficient crowdsourcing for multi-class labeling," in *Proc. ACM SIGMETRICS*, 2013, pp. 81–92.

[10] A. Ghosh, S. Kale, and P. McAfee, "Who moderates the moderators?: crowdsourcing abuse detection in user-generated content," in *Proc. ACM EC*, 2011, pp. 167–176.

[11] C.-J. Ho, S. Jabbari, and J. W. Vaughan, "Adaptive task assignment for crowdsourced classification," in *Proc. ICML*, 2013, pp. 534–542.

[12] D. R. Karger, S. Oh, and D. Shah, "Iterative learning for reliable crowdsourcing systems." in *NIPS*, 2011, pp. 1953–1961.