Ability Grouping of Crowd Workers via Reward Discrimination

Yuko Sakurai[1], **Tenda Okimoto**[2], **Masaaki Oka**[3], **Masato Shinoda**[4], **and Makoto Yokoo**[3]
1: Kyushu University and JST PRESTO, Fukuoka, 819-0395, Japan,
2: Transdisciplinary Research Integration Center, Tokyo, 101-8430, Japan,
3: Kyushu University, Fukuoka, 819-0395, Japan
4: Nara Women's University, Nara, 630-8506, Japan

Abstract

We develop a mechanism for setting discriminated reward prices in order to group crowd workers according to their abilities. Generally, a worker has a certain level of confidence in the correctness of her answers, and asking about it is useful for estimating the probability of correctness. However, we need to overcome two main obstacles to utilize confidence for inferring correct answers. One is that a worker is not always well-calibrated. Since she is sometimes over/underconfident, her confidence does not always coincide with the probability of correctness. The other is that she does not always truthfully report her confidence. Thus, we design an indirect mechanism that enables a worker to declare her confidence by choosing a desirable reward plan from the set of plans that correspond to different confidence intervals. Our mechanism ensures that choosing a plan including true confidence maximizes the worker's expected utility. We also propose a method that composes a set of plans that can achieve requester-specified accuracy in estimating the correct answer using a small number of workers. We show our experimental results using Amazon Mechanical Turk.

Introduction

One of the most notable services recently introduced to the Web is crowdsourcing such as Amazon Mechanical Turk (AMT). It is based on the idea of the *wisdom of crowds* and solves a problem by combining the forces of many people (Ho and Vaughan 2012; Law and Ahn 2011; Shaw, Horton, and Chen 2011; Snow et al. 2008). Using crowdsourcing services, a requester can ask many workers around the world to do her task at a relatively low cost. Crowdsourcing is also gathering attentions from computer science researchers as a platform for *Human Computation*, which solves problems that can only be solved by a computer. It utilizes human intelligence as functions in computer programs.

An advantage of crowdsourcing is a large work force available at relatively low cost, but the quality of the results is sometimes problematic. In image classification, for example, workers label sample images that are used as training data in machine learning. Although the cost of labels by workers is lower than by experts, the possibility of errors in the former is generally higher than in the latter.

One straightforward way to infer accurate labels in crowdsourcing is to ask multiple workers to label the same data and accept the majority vote given by the workers. This corresponds to treating the quality of the labels given by different workers equally and simply considering the labels that receive the largest number of votes as the true ones. In crowdsourcing, however, since workers' abilities are not even, treating all labels given by different workers equally is not always a good way to infer true labels.

We consider a quality control mechanism where a requester asks workers not only for labels to the data but also for their confidence about them. Confidence is a worker's private information about the probability of correctness of her answer. For example, if we adopt a mechanism that pays more reward to more confident workers, they will obviously be tempted to over-declare their confidence. Therefore, we have to develop a mechanism that is robust against strategic manipulations by workers regarding their confidence.

Mechanism design studies the designing of a game's rules/ protocols so that agents have an incentive to truthfully declare their preferences, and designer can select socially desirable outcomes. Traditionally, mechanism design has been investigated in the areas of microeconomics and game theory (Nisan et al. 2007). Recently, along with the popularization of network environments, such study is attracting much attention from computer scientists. In particular, much work in mechanism design has produced mechanisms for the Internet auctions, prediction markets and so on. Furthermore, studies on designs for quality control mechanisms for crowdsourced tasks have been advanced by AI and multi-agent system researchers (Bacon et al. 2012; Cavallo and Jain 2012; Lin, Mausam, and Weld 2012; Witkowski and Parkes 2012a; 2012b)

Strictly proper scoring rules have been proposed for eliciting truthful subjective beliefs/prediction of agents (Gneiting and Raftery 2007; Matheson and Winkler 1976; Savage 1971). Recently, studies related to proper scoring rules have been advanced by AI and multi-agent system researchers (Boutilier 2012; Robu et al. 2012; Rose, Rogers, and Gerding 2012; Witkowski and Parkes 2012a; 2012b). If we assume each worker believes that her confidence is well-calibrated and the requester will find a correct answer, applying a direct mechanism results in the identical mechanism handled by the proper scoring rule.

However, our preliminary experiments show that it is troublesome for workers to directly report their numeric confidence and some crowd workers are not well-calibrated. Also, in psychometrics, even if examinees are asked to numerically give their confidence, many reported it as if it was binary (0% or 100%) (Kato and Zhang 2010). How precisely a worker estimates the confidence (probability of correctness) depends on her intrinsic ability; she usually cannot control it. That is, she cannot know whether she is overconfident, underconfident, or well-calibrated. Therefore, we need to design a mechanism that is tolerant against over/underestimation. Furthermore, our mechanism must be robust to the strategic behavior of workers who report their confidence. How a worker reports her confidence is under her control, because she can strategically over- or under-declare it.

Thus, we focus on the difficulty for a worker to estimate her confidence and propose an *indirect mechanism* that enables a worker to declare her confidence by choosing a desirable reward plan from the set of plans that correspond to different confidence intervals. By showing several reward plans, a requester can entice workers to consider their confidence at required levels of detail. Our mechanism ensures that a worker's expected utility is maximized if she faithfully selects a reward plan that corresponds to her confidence.

We also develop a method with which a requester can make a set of reward plans based on prior knowledge about the relationship between confidence and the actual probability of correctness. Such knowledge can be obtained from the requester's previous experience or can be adopted from more general observations in cognitive psychology such as that humans tend to be overconfident with difficult problems and underconfident with easy problems. Our method enables a requester to determine the number of workers per task and a set of reward plans that can achieve the specified accuracy.

Finally, we evaluated our proposed mechanism by posting tasks on AMT. Our results show that workers can be classified into two groups with different abilities by referring to the reward plans they choose. Furthermore, our method with two reward plans determines a higher probability of correct answers than direct mechanism and majority voting, even when a requester's prior knowledge about workers is not accurate.

Preliminaries

For simplicity, we consider a task for which the answer is given as a binary label $\{0, 1\}$ such as a yes/no decision problem or image labeling. Let $l \in \{0, 1\}$ denote the true label (answer) for the task. The set of workers is denoted as N. The requester specifies the number of workers denoted as n to solve the problem when he posts a task in crowdsourcing. The label given by worker $i \in N$ is denoted as $l_i \in \{0, 1\}$. We define the accuracy of worker i as follows.

Definition 1 (Correctness) *The accuracy of worker i, that is, the probability that worker i correctly assigns the label, is defined as $a_i = P(l_i = l)$.*

Next, we define the confidence estimated by worker i.

Definition 2 (Confidence) *Confidence x_i stands for worker i's subjective probability of her answer's correctness. If the worker is well-calibrated, x_i is identical to a_i. If the worker is overconfident, $x_i > a_i$ holds. If the worker is underconfident, $x_i < a_i$ holds.*

Also, we assume that a worker declares her confidence as y_i. y_i is not always equal to x_i.

We assume that $x_i \in [0.5, 1]$ holds, since we focus on a task with binary labels. $x_i = 0.5$ means that the worker randomly decides her label. On the other hand, $x_i = 1$ means that she has absolutely no doubt about her label. Furthermore, since each worker executes a task at different times all over the world, we assume that confidences $x_1, x_2, \ldots, x_n$ are independent and identically distributed.

Definition 3 (Reward) *When the requester posts a task in crowdsourcing, he sets two reward functions f and g. The worker's reward is $f(y_i)$ if the requester concludes that her label l_i is true, and it is $g(y_i)$ if her label l_i is false.*

For any confidence score of y_i, it is natural to assume that $f(y_i) \geq g(y_i)$. Without assuming this condition, a worker may have an incentive to declare the opposite label.

Based on this definition, we assume that each worker declares label l_i that she considers true. Next we define the expected utility of worker i.

Definition 4 (Expected Utility) *When worker i with her true confidence x_i declares y_i, her expected utility is defined as*

$$u(x_i, y_i) = x_i f(y_i) + (1 - x_i) g(y_i).$$

We assume that each worker believes that the requester will make a correct decision. In crowdsourcing, it is difficult for each worker to know other workers' abilities as a common knowledge, since the workers are gathered via the network and do not know each other. Also, when the number of workers is reasonably large, so single worker has a decisive power to reverse the decision of the requester. Thus, we assume that a worker thinks the requester will judge her answer correct with probability x_i.

Reward discrimination: m-plan mechanism

A requester infers the true label (answer) by aggregating workers' labels (answers) to the data by considering the different abilities of workers to produce true answers. We propose an indirect mechanism in which a worker selects her desirable reward plan among multiple reward plans instead of reporting her confidence. We call our mechanism the m-plan mechanism. It can categorize the reliabilities of workers' labels by making workers select a reward plan. Also, it gives a worker an incentive to report the most preferred plan which indicates an interval including her true confidence.

The procedure of m-plan mechanisms is as follows: (1) After solving a labeling problem, the requester shows a set of reward plans to a worker and asks her to select the most preferred reward plan as well as a label for the data. (2) A worker reports her choice for the reward plan and the label. (3) The requester infers the true label from labels by multiple workers and pays each worker based on her reward plan and her label.

Method of determining rewards

We explain a method for determining the rewards for each plan. First, we assume that the requester divides the range of confidence scores into m intervals. Let $\mathbf{s} = (s_0, \ldots, s_{m-1}, s_m)$ be the list of threshold confidences, where $0.5 = s_0 \leq s_1 \leq \ldots \leq s_{m-1} \leq s_m = 1$. Plan j means the interval $(s_{j-1}, s_j]$[1]. The case of $m = 1$ corresponds to majority voting, because the requester cannot classify the workers by their confidences. On the other hand, when we set m to $+\infty$ and let intervals be small enough, it becomes equivalent to the direct revelation mechanism.

Plan j has two different reward amounts (α_j, β_j): α_j when a reported label is correct and β_j when it is incorrect. Reward functions f and g consist of $\alpha_1, \ldots, \alpha_m$ and $\beta_1, \ldots, \beta_m$. Thus, we define

$$\begin{aligned} f(y_i) &= \sum_{1 \leq j \leq m} \alpha_j I_{(s_{j-1}, s_j]}(y_i), \\ g(y_i) &= \sum_{1 \leq j \leq m} \beta_j I_{(s_{j-1}, s_j]}(y_i) \end{aligned}$$

where $I_{(a,b]}(y_i)$ means that $I_{(a,b]}(y_i) = 1$ if $y_i \in (a, b]$ and $I_{(a,b]}(y_i) = 0$ if $y_i \notin (a, b]$.

Mechanism design studies are to design rules/ protocols so that agents have an incentive to truthfully declare their preferences. Here, we give a definition of a sincere report for workers in our indirect mechanism with a set of m reward plans.

Definition 5 (Sincere report) *For a set of m reward plans, a sincere (or straightforward) report of player i, whose true confidence is x_i, is to choose plan j^*, which corresponds to the confidence interval of $(s_{j^*-1}, s_{j^*}]$ which includes her true confidence x_i.*

We show how to construct the reward plans to derive sincere reports from the workers.

Definition 6 (Set of m Reward Plans) *The requester sets reward plans as follows to classify the workers into m groups based on their confidences.*

- *For any plan j, the reward for correct labels should be higher than the reward for incorrect labels:*
$$\alpha_j \geq \beta_j.$$
- *The rewards for correct labels increase with respect to j:*
$$\alpha_1 < \alpha_2 < \ldots < \alpha_m.$$
- *The rewards for incorrect labels decrease with respect to j:*
$$\beta_1 > \beta_2 > \ldots > \beta_m.$$
- *The expected utility of plan j is the same as that of plan $j + 1$ at s_j:*
$$\alpha_j s_j + \beta_j(1 - s_j) = \alpha_{j+1} s_j + \beta_{j+1}(1 - s_j).$$

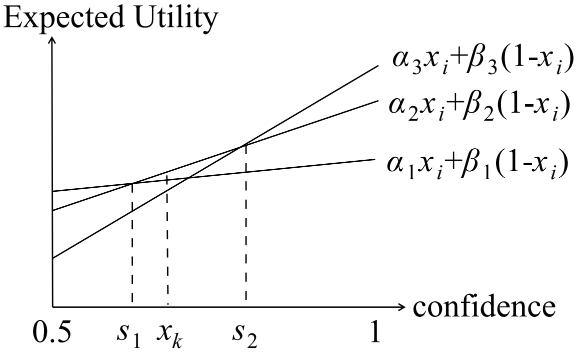


Figure 1: Expected utilities for m-plan mechanism

Next, we show that this method can guarantee that truthful reports are the best strategy for worker expectations.

Theorem 1 *When the requester offers workers a set of m reward plans in Definition 6, a worker can maximize her expected utility by sincere report. To be more specific, if $x_i \in (s_{j^*-1}, s_{j^*}]$,*

$$\forall\, y_i \in (s_{j^*-1}, s_{j^*}],\ u(x_i, y_i) = \max_{y'_i} u(x_i, y'_i).$$

Proof *If we regard $F(x_i, y_i)$ as a function of y_i, $F(x_i, y_i)$ is constant in each interval $y_i \in [s_{j-1}, s_j]$. So it suffices to show that $\max_{y'_i} F(x_i, y'_i) = F(x_i, x_i)$. First, $y_i \in [s_{j^*}, s_{j^*+1}]$ implies $F(x_i, x_i) \geq F(x_i, y_i)$, since we obtain that*

$$\begin{aligned} &F(x_i, x_i) - F(x_i, y_i) \\ &\quad= \ -x_i(\alpha_{j^*+1} - \alpha_{j^*}) + (1 - x_i)(\beta_{j^*+1} - \beta_{j^*}) \\ &\quad\geq \ -s_{j^*}(\alpha_{j^*+1} - \alpha_{j^*}) + (1 - s_{j^*})(\beta_{j^*+1} - \beta_{j^*}) = 0. \end{aligned}$$

Using this inequality repeatedly, we have $F(x_i, x_i) \geq F(x_i, y_i)$ for $j^* < j$ and $y_i \in [s_{j-1}, s_j]$. We see $F(x_i, x_i) \geq F(x_i, y_i)$ in case of $j^* > j$ and $y_i \in [s_{j-1}, s_j]$ in a similar way.

Example 1 *Consider a task with a set of three reward plans. We assume that the confidence of worker k, named x_k, is founded in $(s_1, s_2]$. Then the maximum expected utility is determined by the function of $\alpha_2 x_k + \beta_2(1 - x_k)$. As shown in Fig. 1, worker k can maximize her expected utility by selecting plan 2.*

Relations with proper scoring rule

We explain relationships between proper scoring rules and our mechanism. Proper scoring rules have been proposed for eliciting truthful subjective beliefs or prediction of agents, e.g., weather forecast, prediction market, and so on.

When a requester sets the number of reward plans to infinity, the mechanism converges to a proper scoring rule. Conversely, from reward functions of the direct mechanism,

[1] Although plan 0 indicates the interval $[0.5, s_1]$ in a precise sense, to simplify notation, we denote $(s_{j-1}, s_j]$ as an interval without exception.

we can define a m-plan mechanism by linear approximation. For example, let us consider a quadratic proper rule. When we assume $f(y_i) = 1 - (1 - y_i)^2$ and $g(y_i) = 1 - y_i^2$, we obtain that $f(1/2) = g(1/2) = 3/4$, $f(3/4) = 15/16$, $g(3/4) = 7/16$ and $f(1) = 1, g(1) = 0$. In this case, we can define $(\alpha_1, \beta_1) = (3/4, 3/4)$, $(\alpha_2, \beta_2) = (15/16, 7/16)$, and $(\alpha_3, \beta_3) = (1, 0)$ as a set of reward plans. Here, we can describe three expected utilities as shown in Fig. 1 where s_1 is set to $5/8$, s_2 is set to $7/8$. Thus, a worker can maximize her expected utility by choosing the plan that includes her true confidence. The expected utility becomes a convex function of her confidence.

Actually, a set of reward plans consists of discrete and finite reward plans. Therefore, a requester only has to define a set of reward plans according to the conditions as stated. In more detail, he can flexibly define the values of α_j and β_j and there is no need to use a specific proper scoring rule.

Furthermore, we do not restrict whether reward functions are symmetric ($f(y_i) = g(1 - y_i)$). If reward functions are confined to be symmetric, $f(1/2) = g(1/2)$ must be satisfied which means that a worker who randomly decides her label obtains an identical reward whether the label is correct or not. However, in our preliminary experiments, some workers estimated 50% confidence, but could exceed 50% accuracy as a result. Thus, it is desirable for requesters to utilize the labels from such workers and so give a worker an incentive to make those with 50% confidence report the true label. Thus, we allow asymmetric reward functions to cover that $f(1/2) > g(1/2)$ holds.

Judgment rule

The advantage of utilizing the elicited confidence interval to determine true labels is that a requester can effectively achieve a higher probability of correctness using a smaller number of workers than majority voting. Most requesters repeatedly post their tasks in crowdsourcing. Therefore, we suppose that a requester has prior knowledge about the abilities of the workers in the population, e.g., the relationship between the confidence estimated by workers and the actual probability of correctness. We develop an efficient judgment rule with which a requester can determine correct answers by utilizing his prior probability information.

First, we define the requester's prior probability information about possible workers.

Definition 7 (Prior information of requester) *A requester has the following information about the worker population before posting a task.*

- $\nu(x_i)$*: density function of worker's confidence* x_i.
$$\int_{0.5}^{1} \nu(x_i) dx_i = 1$$
is satisfied.
- h_j*: probability that a worker selects plan* j *on condition that the worker is truthful. Formally, we define* h_j *as* $P(s_{j-1} \leq x_i < s_j)$*, that is,*
$$h_j = \int_{s_{j-1}}^{s_j} \nu(x_i) dx_i.$$
- $E(x_i)$ *: conditional expectation of answer accuracy for given confidence* $x = x_i$*, that is,*
$$P(a_i \mid x = x_i).$$

The requester can properly classify the abilities of workers and achieve the higher required accuracy, due to the increase in prior information about workers. However, practically, the amount of information a requester can learn about workers is limited. From this point of view, it is reasonable that requesters can learn such probability information about workers as defined above. They can calculate the average accuracy for each plan based on this prior information.

Definition 8 (Average accuracy) *The average accuracy for plan* j *is given by*

$$c_j = (\int_{s_{j-1}}^{s_j} E(x_i)\nu(x_i)dx_i) \times h_j^{-1}.$$

Next we introduce the judgment rule. In aggregating the workers' labels, $w_{1,j}$ denotes the number of labels 1 in plan j and $w_{0,j}$ denotes the number of labels 0 in plan j. We denote the set of the number of each label reported from the workers by

$$W = \{\mathbf{w} = \begin{pmatrix} w_{1,1}, \ldots, w_{1,m} \\ w_{0,1}, \ldots, w_{0,m} \end{pmatrix} \mid \sum w_{i,j} = n\}.$$

If the true label $l = 1$, the conditional probability of $\mathbf{w}$ is

$$\begin{aligned} P(\mathbf{w} \mid l = 1) &= n!(\prod_{k \in \{0,1\}, 1 \leq j \leq m} w_{k,j}!)^{-1} \\ &\times \prod_{1 \leq j \leq m} (h_j c_j)^{w_{1,j}} (h_j(1 - c_j))^{w_{0,j}} \\ &\equiv d_{\mathbf{w}}. \end{aligned}$$

We define a judgment rule for determining the correct answer.

Definition 9 (Judgment rule) *If* $P(\mathbf{w} \mid l = 1) > P(\mathbf{w} \mid l = 0)$*, the requester judges the answer to be* 1*. If* $P(\mathbf{w} \mid l = 1) = P(\mathbf{w} \mid l = 0)$*, the requester judges the answer to be* 1 *or* 0 *randomly. This judgment rule is equivalent to*

$$\sum_{1 \leq j \leq m} (w_{1,j} - w_{0,j}) \log \frac{c_j}{1 - c_j} > 0 \quad (= 0).$$

The requester decides the answer by weighted voting, where a vote from a worker who selects plan j is counted with $\log(c_j/(1 - c_j))$. This is based on a maximum likelihood estimation method.

Method of constructing m-plans

In practice, requesters want to achieve specific accuracy from the aggregated labels of workers. We refer to ζ as the expected accuracy required by requesters. We propose a method to determine the required number of workers and reward plans to achieve required accuracy ζ, when a requester applies our judgment rule. Without loss of generality, hereafter we assume the true label $l = 1$.

Definition 10 *We define the expected accuracy of the requester's judgment as*

$$d(n,m,\mathbf{s}) = \sum_{\mathbf{w}\in W'} d_{\mathbf{w}} + \frac{1}{2}\sum_{\mathbf{w}\in W''} d_{\mathbf{w}},$$

where $W' = \{\mathbf{w} \in W \mid P(\mathbf{w} \mid l = 1) > P(\mathbf{w} \mid l = 0)\}$ *and* $W'' = \{\mathbf{w} \in W \mid P(\mathbf{w} \mid l = 1) = P(\mathbf{w} \mid l = 0)\}$*.*

We provide a lower bound of the expected accuracy.

Theorem 2 *When a requester offers a set of m reward plans to n workers, he obtains a lower bound of the required accuracy: for any* $0 < \theta < 1$*,*

$$\begin{aligned} & d(n,m,\mathbf{s}) \\ \geq\ & 1 - \Big(\sum_{1\le j\le m} h_j (c_j^{\theta}(1-c_j)^{1-\theta} + c_j^{1-\theta}(1-c_j)^{\theta})\Big)^n. \end{aligned}$$

If we select $\theta = 1/2$*, we have*

$$d(n,m,\mathbf{s}) \geq 1 - \Big(2\sum_{1\le j\le m} h_j\sqrt{c_j(1-c_j)}\Big)^n.$$

Proof *Let z_i be the weighted vote of worker i. Then its probability distribution is* $P(z_i = \log\frac{c_j}{1-c_j}) = h_j c_j$ *and* $P(z_i = \log\frac{1-c_j}{c_j}) = h_j(1-c_j)$ *for each j. We regard $d(n,m,\mathbf{s})$ as the probability that the sum of z_i becomes positive. By considering the expectation of* $\exp(-\theta(z_1 + z_2 + \cdots + z_n))$*, we can get*

$$\begin{aligned} & P(z_1 + \ldots + z_n \le 0) \\ \le\ & \Big(\sum_{1\le j\le m} h_j (c_j^{\theta}(1-c_j)^{1-\theta} + c_j^{1-\theta}(1-c_j)^{\theta})\Big)^n \end{aligned}$$

by exponential Chebyshev's inequality. □

Let c represent the average accuracy of the workers: $c = \int_{0.5}^{1} \nu(x_i)E(x_i)dx_i$. We assume $c > 0.5$, since we assume that not all of the workers randomly decide their labels. For $m = 1$, Th. 2 induces $d(n,1,\cdot) \geq 1 - (2\sqrt{c(1-c)})^n$. This inequality implies $\lim_{n\to\infty} d(n,1,\cdot) = 1$ if $c > 0.5$.

Next, we determine the minimal number of workers so that the required accuracy achieves ζ. We also show that this theorem guarantees the existence of the minimal value.

Theorem 3 *We assume that a requester sets his required accuracy to* ζ (< 1)*. The minimal number of workers such that* $d(n,m,\mathbf{s}) \geq \zeta$ *is obtained by* $n^* = \min\{n \mid d_m(n) \geq \zeta\}$*, where the definition of $d_m(n)$ is given in the following proof.*

Proof *We define the upper bound of $d(n,m,\mathbf{s})$ for fixed n, m:*

$$d_p(n,m) = \sup_{\mathbf{s}} d(n,m,\mathbf{s}).$$

Since this $d_p(n,m)$ is increasing with respect to n and m, the limit

$$d_m(n) = \sup_m d_p(n,m) = \lim_{m\to\infty} d_p(n,m)$$

exists. Therefore, $\lim_{n\to\infty} d_m(n) = 1$ *holds and then it is guaranteed that* $d_m(n) \geq \zeta$ *is satisfied for sufficiently large n. As a result, we can obtain* $n^* = \min\{n \mid d_m(n) \geq \zeta\}$*.* □

We define non-deficit property which is important for a requester in practice.

Definition 11 (Non-deficit) *The mechanism satisfies non-deficit property, if the expected total payments of the requester b_e are not greater than his budget b:* $b - b_e \geq 0$*.*

Here, guaranteeing non-deficit in expectation means that the mechanism satisfies non-deficit on average if a requester executes identical tasks multiple times. Furthermore, in mechanism design literatures, it is common to estimate non-deficit in expectation when a mechanism designer has prior probability information.

We show how to determine proper reward plans under budget constraints. The upper bound of the number of workers is denoted as n_a, and the upper bound of the number of plans is denoted as m_a. We estimate cost b_e of m-plan mechanism by calculating the total amount of expected rewards:

$$\begin{aligned} b_e &= \sum_{\mathbf{w}\in W'} \{d_{\mathbf{w}} \sum_{1\le j\le m} (w_{1,j}\alpha_j + w_{0,j}\beta_j)\} \\ &+ \sum_{\mathbf{w}\in W''} \{d_{\mathbf{w}} \sum_{1\le j\le m} (w_{1,j} + w_{0,j})\frac{\alpha_j + \beta_j}{2}\} \\ &+ \sum_{\mathbf{w}'\in W\setminus(W'\cup W'')} \{d_{\mathbf{w}} \sum_{1\le j\le m} (w_{1,j}\beta_j + w_{0,j}\alpha_j)\}. \end{aligned}$$

Definition 12 (Method of constructing m plans) *When a requester defines the parameters for the constraints, she can construct appropriate reward plans:*
Implementability: *a requester needs to confirm that it is possible to satisfy the required accuracy under his restrictions. The required number of workers is calculated by* $n_r = \inf\{n | d_m(n) > \zeta\}$*. If* $n_r > n_a$*, the requester needs to decrease the value ζ.*
Searching for a triplet $(n,m,\mathbf{s})$**:** *By increasing the number of workers n from n_r to n_a and the number of plans m from 1 to m_a, the requester searches for a triplet $(n,m,\mathbf{s})$ that satisfies* $D_p(n,m,\mathbf{s}) > \zeta$*.*
Checking non deficit: *Check whether expected cost b_e is smaller than his budget b. If* $b_e > b$*, then lower the rewards.*

In a real setting, the assumption of $m \ll n$ is reasonable, since most tasks posted in crowdsourcing are micro tasks with micro cost. The increase in m increases the requester's total cost. When $m \ll n$ holds, the computational cost of executing this method is $O(n^{2m})$. Furthermore, given a requester's budget, we can compute the a set of reward plans which minimizes the number of workers needed in order to guarantee a required accuracy.

Example 2 *A requester estimates that* $\nu(x_i) = 2$ *for* $x_i \in [0.5, 1]$*, i.e., x_i is uniformly selected. Also, he estimates* $E(x_i) = 0.8x_i + 0.1$*, i.e., each worker tends to be slightly overconfident. Now, the requester sets the maximum number of plans* $m_a = 5$*, required accuracy* $\zeta = 0.9$*, and budget* $b = 30$*. He wants to determine the smallest number of reward plans and the minimum number of workers to achieve his required accuracy under a budget constraint.*

First, we calculate $\lim_{m\to\infty} d_p(n,m)$ *and get* $n_r = 7$ *since* $d_m(7) = 0.911 > 0.9$*. Next, from* $m = 1$ *to* $m_a = 5$*,* *we calculate whether* $\mathbf{s}$ *exists such that* $d(7, m, \mathbf{s}) \geq 0.9$. *We find* $\mathbf{s} = (s_0, s_1, s_2) = (0.5, 0.75, 1)$, *which satisfies* $d(7, 2, \mathbf{s}) = 0.902 > 0.9$. *As a result, the requester sets* $n = 7$ *and* $m = 2$. *We can divide the workers into two groups by threshold confidence* 0.75 *with* $(\alpha_1, \beta_1) = (4, 3)$ *and* $(\alpha_2, \beta_2) = (5, 0)$. *The expected total cost becomes* $b_e = 27.03$. *Since* $27.03 < 30$ *holds, the requester determines his appropriate reward plans.*

The m-plan mechanism may decrease accuracy more than the direct revelation mechanism. Thus, we can diminish the loss of accuracy by increasing the number of plans.

Evaluation

We experimentally evaluated the performance of our proposed method on AMT. We show the two experimental results: (1) how well a crowdsourced worker determines her confidence in the correctness of her answers. (2) how effectively the m-plan mechanism categorizes worker abilities and obtains higher accuracy in estimating true labels compared to majority voting.

In AMT, as Human Intelligence Tasks (HITs), we posted tasks that are considered difficult for computers to solve without human assistances. We set two acceptance criteria for workers: (1) HITs approval rate for all requester HITs exceeds 80% and (2) number of approved HITs exceeds 50 HITs, since a requester can limit his HITs to workers who meet specific criteria on AMT.

Preliminary Experiments

In our preliminary experiments, we evaluate the relationship between the confidence that a worker estimates and her actual accuracy. We executed two kinds of tasks on AMT: name disambiguation and image labeling. In a task of name disambiguation, a worker views two web pages and guesses whether the name Alexander Macomb appearing on both pages refers to the same person. Alexander Macomb who commanded the army at the Battle of Plattsburg is a famous historical personage. In a task of image labeling, a worker views an image of a bird and selects its correct name.

In our HITs, a worker solved 10 problems and declared her confidence in her estimate of the percentage of correct answers. We proposed two kinds of monetary incentives to workers:

- Reward for reporting correct answers: If a worker's answer is correct, she gets 1 cent. Otherwise, she gets 0 cents.
- Reward for estimating and reporting true confidence: If the estimation of a worker is correct, she receives 3 cents. Otherwise, she gets 0 cents.

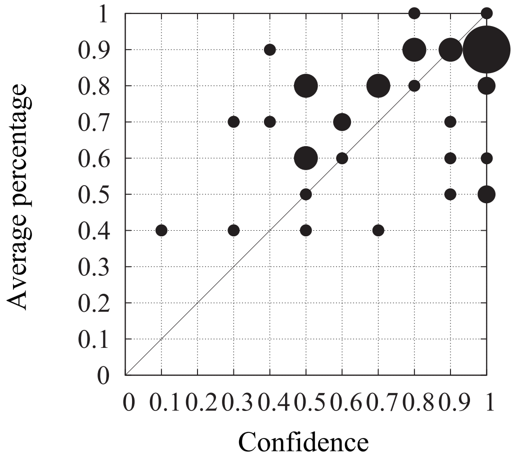


Figure 2: Results of preliminary experiment in Name disambiguation

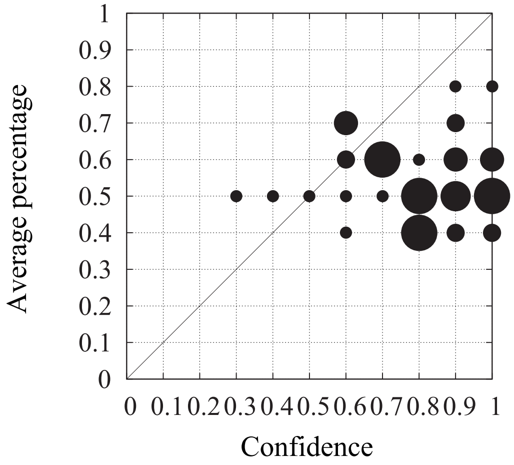


Figure 3: Results of preliminary experiment in Image labeling

Figures 2 and 3 present results for the 50 workers. The size of point indicates the number of workers. In image labeling, overconfident workers were 60%, well-calibrated workers were only 14%, and underconfident workers were 20%. 66% workers reported their confidence within 10% of the actual percentage of correct answers. On the other hand, in name disambiguation, overconfident workers were 84%, well-calibrated workers were only 6%, and underconfident workers were 10%. Most workers are over/underconfident, since this task is complicated and difficult for workers. However, for 28% of workers, the difference between her reported confidences and accuracy was within 10% and also for 46%, the difference was within 20%.

Furthermore, in another kind of HITs for image labeling, we asked a worker to declare a numeric confidence score on each answer. We provided an identical reward for all workers and did not give any monetary incentive to workers to declare her confidence truthfully, since this incentive mechanism is identical to the proper scoring rule. We gathered 100 workers and a worker answered 10 problems. After solving each problem, a worker entered any numeric confidence score from 0% to 100%. The following table shows the relationships between average confidence and average accuracy. 3 workers did not answer her confidence scores. We found that about half the workers always answered 100% or the score close to 100%.

Table 1: Relationships between average confidence and average accuray

Confidence	$30\% \sim$	$40\% \sim$	$50\% \sim$	$60\% \sim$	$70\% \sim$	$80\% \sim$	$90\% \sim$	100%
Accuracy	80%	65%	66%	68.5%	68.5%	68.3%	83.3%	94.1%
♯ workers	1	2	7	9	12	17	27	22

Our results suggest that the precision of estimating a worker's confidence depends on task difficulty. These results indicate that it is more appropriate to roughly ask workers about their confidence rather than to ask them for precise values of it.

Experiments for reward plans

We executed two kinds of tasks to evaluate the performance of our proposed reward plans: name disambiguation and image labeling.

Name disambiguation First, in name disabiguation, we executed experiments on AMT using 5 problems whose average percentages of correct answers exceeded 50% in the preliminary experiment. Based on the preliminary experiment, we calculated that $\nu(x_i) = 2.68x_i - 0.01$ and $E(x_i) = 0.6x_i + 0.24$ for name disambiguation. For name disambiguation, $d(5, 2, (0.5, 0.99, 1)) = 0.65$ is the best required accuracy. Although this expected accuracy might be low, we can improve the accuracy by increasing the number of workers and the number of plans. Since we wanted to show how well our mechanism could perform e ven when the expected accuracy is relatively low, we evaluated our mechanism in the simplest settings where the number of workers was 5 and the number of plans was 2.

We evaluated our mechanism for three types of sets of reward plans. For sets 1, 2, and 3, the threshold confidence was set to $1/2$, $2/3$, and $3/4$, respectively. As shown in Table 2, we determined each pair of rewards as the pair of minimal prices that satisfy the threshold condition. For Set 1, this setting means that choosing Plan 2 is the best strategy when a worker has any confidence score in $(0.5, 1]$, since the expected utility of Plan 2 exceeds Plan 1. If a worker's confidence score is 0.5, her expected utility becomes 2 for both plans. Thus, there exist possibility that a worker with confidence score 0.5 selects Plan 1. Strictly speaking, Set 1 cannot divide agents' abilities into 2 groups.

We aggregated the results by five workers for all HITs. Table 3 shows the percentages which plan workers selected. For both Sets 2 and 3, 68% workers selected Plan 1. Table 4 shows how many seconds it took to be completed. Interestingly, workers in Set 3 spent longer time than workers in Sets 1 and 2.

Table 6 shows the average percentages of correct answers for each plan. The reason why setting five workers is to verify the performance of our mechanism for a small number of workers. Our mechanism separates a set of workers into two groups when the threshold is set to $2/3$ which is close to theoretical appropriate threshold confidence. In Set 2, the result of Plan 2 reached 100% accuracy. First reason is that our mechanism worked well and the second reason might be that the reward for incorrect answer was set to 0. As shown in Table 4, the workers in Plan 2 of Set 2 spent the longest time. This result implies that the workers made a deliberate decision. In Set 3, the result of Plan 2 was not so good. We suppose that some workers with over-confidence selected Plan 2.

Table 2: Reward plans for a HIT on AMT

	Plan 1	Plan 2
Set 1 (1/2)	$(2, 2)$	$(3, 1)$
Set 2 (2/3)	$(2, 2)$	$(3, 0)$
Set 3 (3/4)	$(4, 4)$	$(5, 1)$

Table 3: Percentage of workers for each plan

	Plan 1	Plan 2
Set 1 (1/2)	40%	60%
Set 2 (2/3)	68%	32%
Set 3 (3/4)	68%	32%

Table 4: Average work time (second)

	Plan 1	Plan 2	Total
Set 1 (1/2)	44.2	49.4	47.32
Set 2 (2/3)	44.29	102	62.76
Set 3 (3/4)	124.94	51.13	101.32

Table 5: Average percentages of correct answer

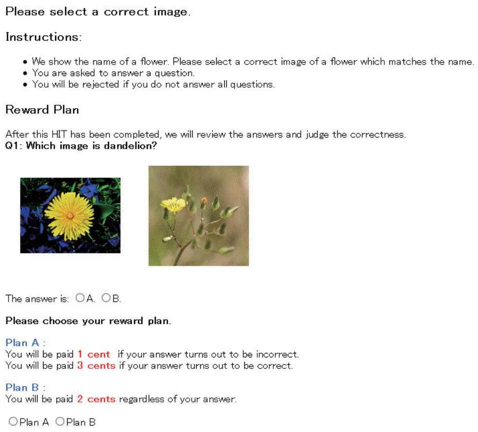
	Name	
	Plan 1	Plan 2
Set 1 (1/2)	60%	53%
Set 2 (2/3)	41%	100%
Set 3 (3/4)	65%	50%

Table 6: Accuracy of judgment

	Name		
	WL	PE	MV
Set 1 (1/2)	60%	60%	60%
Set 2 (2/3)	100%	80%	80%
Set 3 (3/4)	80%	80%	80%

Furthermore, we compared the obtained accuracy of our judgment rule with that of majority voting.@ In the field of machine-learning, several useful techniques to control qualities of crowdsourced tasks have been proposed and more elaborated machine-learning based methods for label aggregation exist (Dai, Mausam, and Weld 2010; Whitehill et al. 2009). However, to estimate the ability of workers, they require each worker to do many tasks. On the other hand, our proposal is to design a quality-control mechanism that can work even when a requester asks workers to perform a single task. Majority voting can be applied to a single task. Thus, we used it as a baseline.

The accuracy means the ratio of correct answers to five problems. WL indicates the accuracy when we assume that $\nu(x_i) = 2$ (uniform distribution) and $E(x_i) = x_i$ (well-calibrated workers) for our proposed judgment rule, which is



Please select a correct image.

Instructions:

- We show the name of a flower. Please select a correct image of a flower which matches the name.
- You are asked to answer a question.
- You will be rejected if you do not answer all questions.

Reward Plan

After this HIT has been completed, we will review the answers and judge the correctness.
Q1: Which image is dandelion?

The answer is: ○A. ○B.

Please choose your reward plan.

Plan A :
You will be paid 1 cent if your answer turns out to be incorrect.
You will be paid 3 cents if your answer turns out to be correct.

Plan B :
You will be paid 2 cents regardless of your answer.

○Plan A ○Plan B

Figure 4: Example of HITs for image labeling

considered a standard case. PE indicates the accuracy when we set ν and E for our proposed judgment rule. MV means the result when we apply majority voting. Our method performed the best when the threshold confidence was set to $2/3$ (Set 2). These results indicate that our mechanism can categorize a set of workers into groups based on their abilities and improve the accuracy in a real setting. Due to space limitation, we show only a fraction of our experimental results. The obtained results were not sensitive to particular parameter settings.

Image labeling As another kinds of tasks with binary labels, we put tasks of image labeling for a flower on AMT as shown in Figure 4. This task is easier for workers than name ambiguation. In this HITs, we gathered 11 workers and offered sets of reward plans for them (Table 7). Based on the preliminary experiment, we calculated that $\nu(x_i) = 2.7x_i - 0.03$ and $E(x_i) = 0.6x_i + 0.24$. $f(5, 2, (0.75)) = 0.86$ is the best required accuracy. Tables 8, 9, 10, and 11 show the average results for 5 problems. Our mechanism worked well for this task, especially for Set 2.

Conclusion

In a crowdsourcing service, asking workers about their confidence is useful for improving the quality of task results. However, it is difficult for workers to precisely report their confidence. So, we focused on the difficulty of estimating confidence and developed an indirect mechanism in which it is guaranteed that it is the best strategy for workers to select a desirable reward plan which includes the worker's true confidence among the small number of plans. We also proposed a method for constructing an appropriate set of reward plans under a requester's constraints on budget and required accuracy.

Future works will include generalizing our mechanism to multiple-choice problems. Also, we will evaluate the usefulness of our mechanisms for various AMT tasks.

Acknowledgments

This work was supported by JST PRESTO program.

Table 7: Reward plans for a HIT on AMT

	Plan 1	Plan 2
Set 1 (1/2)	$(2, 2)$	$(3, 1)$
Set 2 (2/3)	$(3, 3)$	$(4, 1)$
Set 3 (3/4)	$(4, 4)$	$(5, 1)$

Table 8: Percentage of workers for each plan

	Plan 1	Plan 2
Set 1 (1/2)	38.2%	61.8%
Set 2 (2/3)	29.1%	70.9%
Set 3 (3/4)	36.4%	63.6%

Table 9: Average work time (second)

	Plan 1	Plan 2	Total
Set 1 (1/2)	21.8	49.2	38.7
Set 2 (2/3)	24.2	27.7	25.1
Set 3 (3/4)	23.6	29.6	25.8

Table 10: Average percentages of correct answer

	Name	
	Plan 1	Plan 2
Set 1 (1/2)	76.2%	88.2%
Set 2 (2/3)	53.8%	87.5%
Set 3 (3/4)	57.1%	75.0%

Table 11: Accuracy of judgment

	Name		
	WL	PE	MV
Set 1 (1/2)	80%	100%	100%
Set 2 (2/3)	80%	100%	80%
Set 3 (3/4)	80%	80%	80%

References

Bacon, D. F.; Chen, Y.; Kash, I.; Parkes, D. C.; Rao, M.; and Sridharan, M. 2012. Predicting your own effort. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, 695–702.

Boutilier, C. 2012. Eliciting forecasts from self-interested experts: Scoring rules for decision makers. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, 737–744.

Cavallo, R., and Jain, S. 2012. Efficient crowdsourcing contests. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, 677–686.

Dai, P.; Mausam; and Weld, D. S. 2010. Decision-theoretic control of crowd-sourced workflows. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI 2010)*.

Gneiting, T., and Raftery, A. E. 2007. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association* 102(477):359–378.

Ho, C.-J., and Vaughan, J. W. 2012. Online task assignment in crowdsourcing markets. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI 2012)*.

Kato, K., and Zhang, Y. 2010. An item response model for probability testing. In *International Meeting of the Psychometric Society*.

Law, E., and Ahn, L. V. 2011. *Human Computation*. Morgan & Claypool Publishers.

Lin, C. H.; Mausam; and Weld, D. S. 2012. Crowdsourcing control: Moving beyond multiple choice. In *Proceedings of the 4th Human Computation Workshop*, 25–26.

Matheson, J. E., and Winkler, R. L. 1976. Scoring rules for continuous probability distributions. *Management Science* 22(10):1087–1096.

Nisan, N.; Roughgarden, T.; Tardos, E.; and Vazirani, V. V. 2007. *Algorithmic Game Theory*. Cambridge University Press.

Robu, V.; Kota, R.; Chalkiadakis, G.; Rogers, A.; and Jennings, N. R. 2012. Cooperative virtual power plant formation using scoring rules. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI 2012)*, 370–376.

Rose, H.; Rogers, A.; and Gerding, E. H. 2012. A scoring rule-based mechanism for aggregate demand prediction in the smart grid. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, 661–668.

Savage, L. J. 1971. Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association* 66(336):783–801.

Shaw, A. D.; Horton, J. J.; and Chen, D. L. 2011. Designing incentives for inexpert human raters. In *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work (CSCW 2011)*, 275–284.

Snow, R.; O'Connor, B.; Jurafsky, D.; and Ng, A. Y. 2008. Cheap and fast - but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2008)*, 254–263.

Whitehill, J.; Ruvolo, P.; Wu, T.; Bergsma, J.; and Movellan, J. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *Advances in Neural Information Processing Systems* 22:2035–2043.

Witkowski, J., and Parkes, D. C. 2012a. Peer prediction without a common prior. In *Proceedings of the 13th ACM Conference on Electronic Commerce (EC 2012)*, 964–981.

Witkowski, J., and Parkes, D. C. 2012b. A robust bayesian truth serum for small populations. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI 2012)*, 1492–1498.