Incentivizing High Quality Crowdsourcing Information using Bayesian Inference and Reinforcement Learning

Anonymous Authors[1]

Abstract

This document provides a basic paper template and submission guidelines. Abstracts must be a single paragraph, ideally between 4–6 sentences long. Gross violations will trigger corrections at the camera-ready phase.

1. Introduction

1.1. Motivation

Peer prediction mechanisms have two fatal drawbacks:

- Existing peer prediction mechanisms only care about incentive compatibility (IC) which only poses requirements to the expected incentives to workers. They achieve IC via comparing the reports between the targeted and selected reference agents. In this way, they only use a tiny part of the information behind all collected labels. Besides, they never analyze the stochastic property of incentives and the variation of incentives among different types of agents.

- Existing peer prediction mechanisms simplify workers' responses to the incentive mechanism by assuming that workers are all fully rational and only follow the utility-maximizing strategy. However, there is strong evidence showing that human workers are not always fully rational, and they may deviate from equilibrium strategies. Thus, these peer prediction mechanisms which is fancy in theory may yet fail in practice.

1.2. Contribution

We have two core contributions in this paper:

- We propose a novel one-shot peer prediction mechanism based on Bayesian inference. Since existing Bayesian inference algorithms (e.g. EM estimator and variational inference) for crowdsourcing are biased in principle, we derive the explicit posterior distribution of the true labels and employ Gibbs sampling for inference. The most challenging problem of our mechanism is to prove the incentive compatibility of our mechanism which has never been explored in the literature. Besides, we also empirically show the advantages of our mechanism on the stability and fairness of incentives over existing ones.

- We design the first reinforcement peer prediction framework which sequentially interacts with workers. It dynamically adjusts the scaling level of our peer prediction mechanism to maximize the utility of the data requester. To avoid assuming a decision-making model for workers, we use the data-driven Gaussian process to represent the scaling level adjustment policy, and online updates our policy according to workers' responses. We theoretically prove the incentive compatibility of our framework and empirically show its advantages on improving the utility if the data requester over one-shot mechanisms.

2. Related Work

3. Learning-Based Peer Prediction

3.1. Formulation and settings

Suppose there is one data requester who assigns M tasks with answer space $\{1, 2\}$ to $N \geq 3$ candidate workers at each step $t = 1, 2 \dots T$. We denote the tasks and workers by $\mathcal{T}^t = \{1, 2, \dots, M\}$ and $\mathcal{C} = \{1, 2, \dots, N\}$, respectively. The label $L_i^t(j)$ generated by worker $i \in \mathcal{C}$ for task $j \in \mathcal{T}^t$ comes from a distribution that depends both on the ground-truth label $L^t(j)$ and worker i's effort level e_i^t and reporting strategy r_i^t. Suppose there are two effort levels, High ($e_i^t = 1$) and Low ($e_i^t = 0$), that a worker can potentially choose from. Meanwhile, worker i can decide either to truthfully report his observation $r_i^t = 1$ or to revert the answer $r_i^t = 0$. Note that worker i's effort level and reporting strategy may be a mixed of the above pure actions. Thus, we use $e_i^t \in [0, 1]$ and $r_i^t \in [0, 1]$ to denote worker i's probability of exerting high efforts and being truthful, respectively. Using the above notations, we can define worker i's score (i.e. the probability of providing the correct label) as

$$
\begin{aligned}
p_i^t =& r_i^t e_i^t p_{i,H} + r_i^t (1 - e_i^t) p_{i,L} + \\
& (1 - r_i^t) e_i^t (1 - p_{i,H}) + (1 - r_i^t)(1 - e_i^t) p_{i,L}
\end{aligned} \tag{1}
$$

where $p_{i,H}$ and $p_{i,L}$ denote the probability of observing the correct label when worker i exerts high and low efforts, respectively. Following the previous studies on peer prediction (Dasgupta and Ghosh, 2013; Liu and Chen, 2017), we assume that all workers share the same set of $p_{i,H}, p_{i,L}$, denoting by p_H, p_L, and $p_H > p_L \geq 0.5$. Besides, we also assume these crowdsourcing tasks are homogeneous.

3.2. One-Step Incentive Mechanism

Now, we present our mechanism designed for inducing truthful reports and high efforts in one step:

Definition 1. *Denote worker i's score estimated by using the Bayesian inference algorithm developed later in this section as $\tilde{p}_i^t$. Then, at step t, our mechanism computes worker i's rewards for her reports on M tasks as:*

$$R^t(i) = a \cdot (\tilde{p}_i^t - 0.5) + b \tag{2}$$

where $a > 0$ and $b \geq 0$ are the scaling level and the guaranteed base payment, respectively.

There have been many inference algorithms developed in the literature of crowdsourcing (Zheng *et al.*, 2017). Among them, the most popular one is the EM estimator (Dawid and Skene, 1979). Another widely-adopted algorithm is the variational inference estimator (Liu *et al.*, 2012) which is similar to the EM estimator (Tzikas *et al.*, 2008). However, these estimators are irredeemably biased since the EM estimator may converge to the local optimum. On the other hand, sampling-based Bayesian inference algorithms, for example Gibbs sampling, are computationally very expensive, even though they use the explicit posterior distribution and can avoid the inference bias. Especially, workers' scores are continuous variables, which will significantly slow down the convergence speed. Therefore, to the best of my knowledge, sampling-based Bayesian inference is never used for crowdsourcing where the number of workers and tasks is usually very large. In this section, to reduce the inference bias and meanwhile avoid overly large computation costs, we firstly assume Dirichlet priors for those continuous variables in our system and derive a joint posterior distribution which only contains the discrete variables. Then, we use Gibbs sampling to sample the obtained posterior distribution and estimate workers' scores based on those samples.

Specially, the joint distribution of the collected labels $\mathcal{L} = [L_i(j)]$ and true labels $\boldsymbol{L} = [L(1), \ldots, L(M)]$ satisfies[1]

$$\begin{aligned} &P(\mathcal{L}, \boldsymbol{L}|\boldsymbol{p}, \boldsymbol{\tau}) = \\ &\quad \prod_{j=1}^{M}\prod_{k=1}^{K}\left\{\tau_k \prod_{i=1}^{N} p_i^{\delta_{ijk}}(1-p_i)^{\delta_{ij(3-k)}}\right\}^{\xi_{jk}} \end{aligned} \tag{3}$$

[1] In this section, we only focus on developing the one-step mechanism and thus omit the superscript t of all variables.

where $\boldsymbol{p} = [p_i]_N$ and $\boldsymbol{\tau} = [\tau_1, \tau_2]$. τ_1 and τ_2 denote the distribution of answer 1 and 2 among all tasks, respectively. Besides, $\delta_{ijk} = \mathbb{1}(L_i(j) = k)$ and $\xi_{jk} = \mathbb{1}(L^t(j) = k)$. Referring to the literature on variational inference (Liu *et al.*, 2012), we assume a Dirichlet prior for both p_i and $\boldsymbol{\tau}$ as

$$[p_i, 1-p_i] \sim \text{Dir}(\alpha_1, \alpha_2)\ ,\ \boldsymbol{\tau} \sim \text{Dir}(\beta_1, \beta_2). \tag{4}$$

where $\text{Dir}(\cdot)$ denotes the Dirichlet distribution. Then, we can derive the joint distribution of $\mathcal{L}$, $\boldsymbol{L}$, $\boldsymbol{p}$ and $\boldsymbol{\tau}$ as

$$\begin{aligned} &P(\mathcal{L}, \boldsymbol{L}, \boldsymbol{p}, \boldsymbol{\tau}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = P(\mathcal{L}, \boldsymbol{L}|\boldsymbol{p}, \boldsymbol{\tau}) \cdot P(\boldsymbol{p}, \boldsymbol{\tau}|\boldsymbol{\alpha}, \boldsymbol{\beta}) \\ &= \frac{1}{B(\boldsymbol{\beta})}\prod_{k=1}^{K}\tau_k^{\hat{\beta}_k - 1} \cdot \prod_{i=1}^{N}\frac{1}{B(\boldsymbol{\alpha})}p_i^{\hat{\alpha}_{i1}-1}(1-p_i)^{\hat{\alpha}_{i2}-1} \end{aligned} \tag{5}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2]$, $\boldsymbol{\beta} = [\beta_1, \beta_2]$ and

$$\begin{aligned} \hat{\alpha}_{i1}^t &= \sum\nolimits_{j=1}^{M}\sum\nolimits_{k=1}^{K}\delta_{ijk}^t \xi_{jk}^t + \alpha_1 \\ \hat{\alpha}_{i2}^t &= \sum\nolimits_{j=1}^{M}\sum\nolimits_{k=1}^{K}\delta_{ij(3-k)}^t \xi_{jk}^t + \alpha_2 \\ \hat{\beta}_k^t &= \sum\nolimits_{j=1}^{M}\xi_{jk}^t + \beta_k. \end{aligned} \tag{6}$$

$B(\cdot)$ denotes the beta function which satisfies

$$B(x, y) = \frac{(x-1)!(y-1)!}{(x+y-1)!}. \tag{7}$$

Furthermore, we can conduct marginalization via integrating Equation 5 over all possible values of $\boldsymbol{p}$ and $\boldsymbol{\tau}$ as

$$\begin{aligned} P(\mathcal{L}, \boldsymbol{L}|\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \int_{\boldsymbol{p}, \boldsymbol{\tau}} P(\mathcal{L}, \boldsymbol{L}, \boldsymbol{p}, \boldsymbol{\tau}|\boldsymbol{\alpha}, \boldsymbol{\beta})\mathrm{d}\boldsymbol{p}\mathrm{d}\boldsymbol{\tau} \\ &= \frac{B(\hat{\boldsymbol{\beta}})}{B(\boldsymbol{\beta})} \cdot \prod_{i=1}^{N}\frac{B(\hat{\boldsymbol{\alpha}}_i)}{B(\boldsymbol{\alpha})} \end{aligned} \tag{8}$$

where $\hat{\boldsymbol{\alpha}}_i = [\alpha_{i1}, \alpha_{i2}]$ and $\hat{\boldsymbol{\beta}} = [\hat{\beta}_1, \hat{\beta}_2]$. Following Bayes' theorem, we can know the posterior distribution satisfies

$$P(\boldsymbol{L}|\mathcal{L}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{P(\mathcal{L}, \boldsymbol{L}|\boldsymbol{\alpha}, \boldsymbol{\beta})}{P(\mathcal{L}|\boldsymbol{\alpha}, \boldsymbol{\beta})} \propto B(\hat{\boldsymbol{\beta}})\prod_{i=1}^{N}B(\hat{\boldsymbol{\alpha}}_i). \tag{9}$$

Note that the previous studies have shown that we should be optimistic about workers' willingness to provide the correct label (Chen *et al.*, 2015), which requires $\alpha_1 > \alpha_2$. In this paper, for the simplicity of theoretical analysis, we set $\alpha_1 = 2$ and $\alpha_2 = 1$. Besides, since we have no knowledge about the distribution of the true labels, we employ the uniform distribution for $\boldsymbol{\tau}$, that is, setting $\beta_1 = \beta_2 = 1$.

To observe the posterior distribution, we resort to the classic Gibbs sampling. Firstly, according to Bayes' theorem, we can know the conditional posterior distribution satisfies

$$P(L(j)|\mathcal{L}, \boldsymbol{L}(\bar{j}), \boldsymbol{\alpha}, \boldsymbol{\beta}) \propto P(\boldsymbol{L}|\mathcal{L}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{10}$$

Algorithm 1 Gibbs sampling for crowdsourcing

Input: the collected labels $\mathcal{L}$, the number of samples W
Output: the sample sequence $\mathcal{S}$
$\mathcal{S} \leftarrow \emptyset$, Initialize $\boldsymbol{L} = [L(j)]_M$ with the uniform distribution
for $s = 1$ **to** W **do**
  for $j = 1$ **to** M **do**
    Set $L(j) = 1$ and compute $x_1 = B(\hat{\boldsymbol{\beta}}) \prod_{i=1}^{N} B(\hat{\boldsymbol{\alpha}}_i)$
    Set $L(j) = 2$ and compute $x_2 = B(\hat{\boldsymbol{\beta}}) \prod_{i=1}^{N} B(\hat{\boldsymbol{\alpha}}_i)$
    $L(j) \leftarrow$ Sample $\{1, 2\}$ with $P(1) = x_1/(x_1 + x_2)$
  end for
  Append $\boldsymbol{L}$ to the sample sequence $\mathcal{S}$
end for

where $\boldsymbol{L}(\bar{j})$ denotes the true labels of all tasks expect for task j. Then, we can generate the samples of the posterior distribution $P(\boldsymbol{L}|\mathcal{L}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ by using Algorithm 1. In each round of sampling, Algorithm 1 traverses all tasks by increasing j from 1 to M and always update the true label vector $\boldsymbol{L}$ via replacing $L(j)$ with the newly obtained sample (line 3-6). Here, we write the s-th sample as $\boldsymbol{L}^{(s)}$. Since Gibbs sampling requires the burn-in process, we need to discard the first b samples in the obtained sample sequence $\mathcal{S}$ and can only use the latter $W - b$ samples. Thus, we can estimate worker i's score p_i as

$$\tilde{p}_i = \frac{\sum_{s=b+1}^{W} \left[\alpha_1 + \sum_{j=1}^{M} \mathbb{1}(L^{(s)}(j) = L_i(j))\right]}{(W - b) \cdot (\alpha_1 + \alpha_2 + M)} \tag{11}$$

and the distribution of true labels $\boldsymbol{\tau}$ as

$$\tilde{\tau}_k = \frac{\sum_{s=b+1}^{W} \left[\beta_1 + \sum_{j=1}^{M} \mathbb{1}(L^{(s)}(j) = k)\right]}{(W - b) \cdot (\beta_1 + \beta_2 + M)}. \tag{12}$$

According to Equation 3, we can calculate the distribution of the true label of task j as

$$P(L(j) = k) = \tau_k \prod_{i=1}^{N} p_i^{\delta_{ijk}} (1 - p_i)^{\delta_{ij(3-k)}} \tag{13}$$

which will be used for the estimation of accuracy in the following section. Besides, both W and b should be large values, and in this paper, we set $W = 1000$ and $b = 100$.

3.3. Reinforcement Incentive Mechanism

need to discuss with Yitao

$$v^t(j) = \left|\log \frac{P(L(j) = 1)}{P(L(j) = 2)}\right|, v^t = \frac{1}{M} \sum_{j=1}^{M} v^t(j) \tag{14}$$

$$F(A^t) = F_v(v^t) \tag{15}$$

4. Game-Theoretic Analysis

4.1. One-Step mechanism analysis

To conduct the game-theoretical analysis on our mechanism, we firstly study the error upper bound of our Bayesian inference algorithm. For the posterior label vector $\boldsymbol{L}$ generated based on the posterior distribution $P(\boldsymbol{L}|\mathcal{L}, \boldsymbol{\alpha}, \boldsymbol{\beta})$, we introduce n and m to denote the number of correct and wrong labels, respectively. Then, we can have

Proposition 1. *Suppose the number of workers whose real score is lower and higher than* 0.5 *are* $N_{>0.5}$ *and* $N_{<0.5}$*, respectively. Then, when* $M \gg 1$*, the expectation*[2] *of the number of wrong labels satisfies*

$$\mathbb{E}_{\mathcal{L},\boldsymbol{L}}\left[\frac{m}{M}\right] \lesssim \frac{(x + e^{\delta})(1 + x)^{M-1}}{(1 + e^{\delta})[1 - e^{-c(y)M}](1 + y)^{M}}$$

where $x = \left(4^{N+1} \prod_{i=0}^{N} \hat{p}_i^2\right)^{-1}$, $y = \left(\prod_{i=0}^{N} \hat{\lambda}_i\right)^{-1}$ *and*

$$\delta = [N_{<0.5} - N_{>0.5}] \log(M) + \phi(\boldsymbol{p})$$

$$\phi(\boldsymbol{p}) = \sum_{i=1}^{N} (-1)^{1(p_i > 0.5)} \varepsilon(\hat{p}_i)\ ,\ c(y) = \frac{(1 - y)^2}{2(1 + y)^2}.$$

Proof. Among the n tasks of which the posterior label is correct, we introduce x_0 and y_0 to denote the number of tasks of which the real true label is 1 and 2, respectively. Among these n tasks, we further use x_i and y_i to denote the number of tasks of which the label provided by worker i is correct and wrong, respectively. Among the m tasks of which the posterior label is wrong, we introduce w_0 and z_0 to denote the number of tasks of which the real true label is 1 and 2, respectively. Among these m tasks, we also introduce w_i and z_i to denote the number of tasks of which the label provided by worker i is correct and wrong, respectively. Thus, we can have $x_i + y_i = n$ and $w_i + z_i = m$. To compute the expectation of m, we need to analyze the probability density function of m. According to Equation 9, we can know the probability density function $P(m)$ satisfies

$$P(m) = \frac{C_M^m}{Z} H(m, p_0; M, 0) \prod_{i=1}^{N} H(m, p_i; M, 1) \tag{16}$$

where Z is the normalization constant and

$$H(m, p; M, t) = \sum_{x=0}^{n} \sum_{w=0}^{m} 2^{M+1} C_n^x C_m^w \times \\ p^{x+w} (1 - p)^{y+z} B(x + z + 1 + t, y + w + 1). \tag{17}$$

Note that, when deriving Equation 17, we utilize the independence between different workers. Besides, due to the

[2] The collected labels can be regarded as random variables generated based on the distribution of true labels and the real scores of workers. Different labels will cause our Bayesian inference to output different estimations. Here, we use the subscript of $\mathbb{E}$ to denote the random variable and follow this rule in this paper.

normalization constant Z, the factor 2^{M+1} will not affect the value of $P(m)$. Applying Lemma 2 in the supplementary file for the H function, we can know

$$H(m,p;M,t) = 2^{M+1}\times \int_0^1 [(2p-1)x+1-p]^{M-m}[(1-2p)x+p]^m x^t \mathrm{d}x. \tag{18}$$

Then, $P(m) = Z^{-1}C_M^m Y(m)$, where

$$Y(m) = e^{\log H(m,p_0;M,0)+\sum_{i=1}^N \log H(m,p_i;M,1)} \tag{19}$$

and $Z = \sum_{m=1}^M C_M^m Y(m)$. The theoretical analysis on the H function is the core of our proof. However, for the clarity of our analysis, we put it in Section 4.2. Here, we directly use the obtained lower and upper bounds of the H function (Propositions 7 and 8) and can have

$$\begin{cases} e^{C-\underline{K}m} \lesssim Y(m) \lesssim e^{C-\overline{K}m} & 2m \le M \\ e^{C+\delta-\underline{K}n} \lesssim Y(m) \lesssim e^{C+\delta-\overline{K}n} & 2m > M \end{cases} \tag{20}$$

where $C = H(0,p_0;M,0) + \sum_{i=1}^N H(0,p_i;M,1)$ and

$$\underline{K} = \sum\nolimits_{i=0}^N \log\hat{\lambda}_i \ , \ \overline{K} = 2\sum\nolimits_{i=0}^N \log(2\hat{p}_i) \ .$$

Then, the expectation of m can be calculated as

$$\mathbb{E}[m] \lesssim \frac{\sum_{m=0}^M me^{-\overline{K}m} + \sum_{m=0}^M me^{\delta-\overline{K}n}}{\sum_{m=0}^k e^{-\underline{K}m} + \sum_{m=k+1}^M e^{\delta-\underline{K}n}} \tag{21}$$

where $k = \lfloor M/2 \rfloor$. By using Lemmas 4 and 5 in the supplementary file, we can know the upper bound of the numerator in Equation 21 is $M(x+e^\delta)(1+x)^{M-1}$. On the other hand, by using Lemma 8 in the supplementary file, we can obtain the lower bound of the denominator in Equation 21 is $(1+e^\delta)[1-e^{-c(y)M}](1+y)^M$. In this case, we can conclude Proposition 1 by combining these two bounds. □

Then, focusing on game-theoretical analysis, we can have

Proposition 2. *When $M\to\infty$ and $(2p_H)^{2(N-1)} > M$, reporting truthfully and exerting high efforts is a Nash equilibrium for all workers in our mechanism.*

Proof. In this proof, we continue to use the notations defined in Proposition 1. According to Equation 11, we can know that $\tilde{p}_i \approx \mathbb{E}_{\boldsymbol{L}}(x_i+z_i)/M$ when $M\to\infty$. Meanwhile, $p_i \approx \mathbb{E}_{\boldsymbol{L}}(x_i+w_i)/M$. Thus, we can have

$$\mathbb{E}_{\mathcal{L}}|\tilde{p}_i - p_i| \approx \mathbb{E}_{\mathcal{L},\boldsymbol{L}}\frac{|w_i - z_i|}{M} \lesssim \mathbb{E}_{\mathcal{L},\boldsymbol{L}}\left[\frac{m}{M}\right]. \tag{22}$$

For worker i, suppose all other workers report truthfully and exert high efforts. Then, in Proposition 1, $N_{<0.5} - N_{>0.5} \le 2-N \le -1$. When $M\to\infty$, $e^\delta \to 0$ and $e^{-M}\to 0$. Thus, $\mathbb{E}_{\mathcal{L},\boldsymbol{L}}[m/M] \lesssim x(1+x)^{M-1}$. Considering $2\hat{p}\ge 1$, we can know that $x \ge (2p_H)^{2(N-1)}$. Then, when $M\to\infty$ and $(2p_H)^{2(N-1)} > M$, we can have $(1+x)^{M-1}\to e$ and $x\to 0$. In this case, $\mathbb{E}_{\mathcal{L}}|\tilde{p}_i - p_i| \to 0$. Besides, we know that p_i reaches the maximum only when worker i report truthfully and exert high efforts. Thereby, in our mechanism, workers can only maximize their rewards by reporting truthfully and exerting high efforts, which concludes Proposition 2. □

4.2. H function analysis

Here, we present our analysis on the H function defined in Equation 18. For the simplicity of notation, we introduce $\bar{p}$ as $\bar{p} = 1-p$. Then, we can have:

Proposition 3. $H(m,0.5;M,t) = 2(t+1)^{-1}$.

Proposition 4. $H(m,p;M,t) = H(n,\bar{p};M,t)$.

Proposition 5. *As a function of m, $H(m,p;M,t)$ is logarithmically convex.*

Proof. Proposition 3 can be proved by integrating $2x^t$ on $[0,1]$. Proposition 4 can be proved by showing that $H(n,\bar{p};M,t)$ has the same expression as $H(m,p;M,t)$. Thus, in the following proof, we focus on Propostion 5. Fixing p, M and t, we denote $\log(H)$ by $f(m)$. Then, we compute the first-order derivative as

$$H(m)f'(m) = 2^{M+1}\int_0^1 \lambda u^n(1-u)^m x^t \mathrm{d}x \tag{23}$$

where $u = (2p-1)x+1-p$ and $\lambda = \log(1-u)-\log(u)$. Furthermore, we can solve the second-order derivative as

$$2^{-2(M+1)}H^2(m)f''(m) = \int_0^1 g^2(x)\mathrm{d}x \int_0^1 h^2(x)\mathrm{d}x - \left(\int_0^1 g(x)h(x)\mathrm{d}x\right)^2 \tag{24}$$

where the functions $g, h : (0,1)\to\mathbb{R}$ are defined by

$$g = \lambda\sqrt{u^n(1-u)^m} \ , \ h = \sqrt{u^n(1-u)^m}. \tag{25}$$

By the Cauchy-Schwarz inequality,

$$\int_0^1 g^2(x)\mathrm{d}x \int_0^1 h^2(x)\mathrm{d}x \ge \left(\int_0^1 g(x)h(x)\mathrm{d}x\right)^2 \tag{26}$$

we can know that $f''(m)\ge 0$ always holds, which concludes that f is convex and H is logarithmically convex. □

For the special case that $t=1$ and $M\gg 1$, we can further derive the following three propositions for $H(m,p;M,1)$:

Proposition 6. *The ratio between two ends satisfies*

$$\log\frac{H(0,p;M,1)}{H(M,p;M,1)} \approx \begin{cases} \log(M)+\epsilon(p) & p>0.5 \\ 0 & p=0.5 \\ -\log(M)-\epsilon(\bar{p}) & p<0.5 \end{cases}$$

where $\epsilon(p) = \log(2p-1)-\log(p)$ and $\epsilon(p)=0$ if $p=0.5$.

Proposition 7. *The lower bound can be calculated as*

$$\log H(m,p) \gtrsim \begin{cases} H(0,p) - \underline{k}m & 2m \le M \\ H(M,p) - \underline{k}n & 2m > M \end{cases}$$

where $\underline{k} = \log\left(\max\left\{p/(\bar{p}+M^{-1}), \bar{p}/(p+M^{-1})\right\}\right)$.

Proposition 8. *The upper bound can be calculated as*

$$\log H(m,p) \lesssim \begin{cases} H(0,p) - \overline{k}m & 2m \le M \\ H(M,p) - \overline{k}n & 2m > M \end{cases}$$

where $n = M - m$ *and* $\overline{k} = 2\log\left(2 \cdot \max\{p, \bar{p}\}\right)$.

Proof. By Proposition 3, $\log H(m, 0.5; M, 1) \equiv 0$, which proves the above three propositions for the case that $p = 0.5$. Considering the symmetry ensured by Proposition 4, we thus focus on the case that $p > 0.5$ in the following proof and transform $H(m,p)$ into the following formulation

$$H(m,p) = \omega(p) \cdot \int_{\bar{p}}^{p} x^n (1-x)^m (x-1+p)\mathrm{d}x \quad (27)$$

where $\omega(p) = 2^{M+1}/(2p-1)^2$. Then, we can solve $H(0,p)$ and $H(M,p)$ as

$$\begin{aligned} H(0,p) &= \omega(p) \int_{\bar{p}}^{p} x^M (x-\bar{p})\mathrm{d}x \\ &= \frac{(2p)^{M+1}}{(2p-1)(M+1)} - O\left(\frac{(2p)^{M+1}}{M^2}\right) \end{aligned} \quad (28)$$

$$\begin{aligned} H(M,p) &= \omega(p) \int_{\bar{p}}^{p} (1-x)^M (x-\bar{p})\mathrm{d}x \\ &= \frac{p(2p)^{M+1}}{(2p-1)^2(M+1)(M+2)} - O\left(\frac{(2\bar{p})^{M+1}}{M+2}\right). \end{aligned} \quad (29)$$

Using the Taylor expansion of function $\log(x)$, we can calculate the ratio in Proposition 6 as

$$\log\frac{H(0,p)}{H(M,p)} = \log(M) + \log\frac{2p-1}{p} + O\left(\frac{1}{M}\right) \quad (30)$$

which concludes Proposition 6 when $M \gg 1$.

Furthermore, we can solve $H(1,p)$ as

$$\begin{aligned} H(1,p) &= \omega(p) \int_{\bar{p}}^{p} x^{M-1}(x-\bar{p})\mathrm{d}x - H(0,p) \\ &= \frac{(2\bar{p}+M^{-1})(2p)^M}{(2p-1)(M+1)} - O\left(\frac{(2p)^{M+1}}{M^2}\right) \end{aligned} \quad (31)$$

The value ratio between $m = 0$ and $m = 1$ then satisfies

$$\log\frac{H(1,p)}{H(0,p)} = \log\frac{p}{\bar{p}+M^{-1}} + O\left(\frac{1}{M}\right). \quad (32)$$

By Rolle's theorem, there exists a $c \in [m, m+1]$ satisfying

$$\log H(1,p) - \log H(0,p) = f'(c) \quad (33)$$

where $f(m) = \log H(m,p)$. Meanwhile, Proposition 5 ensures that $f''(m) \ge 0$ always holds. Thus, we can have

$$\log H(m+1,p) - \log H(m,p) \ge \log\frac{H(1,0)}{H(0,p)} \quad (34)$$

which concludes the first case of Proposition 7. Similarly, we compute the ratio between $m = M-1$ and M as

$$\log\frac{H(M,p)}{H(M-1,p)} = \log\frac{p}{\bar{p}+M^{-1}} + O\left(\frac{1}{M}\right). \quad (35)$$

Meanwhile, Rolle's theorem and Proposition 5 ensure that

$$\log H(m,p) - \log H(m-1,p) \le \log\frac{H(M,0)}{H(M-1,p)} \quad (36)$$

which concludes the second case of Proposition 7.

Lastly, we focus on the upper bound described by Proposition 8. According to the inequality of arithmetic and geometric means, $x(1-x) \le 2^{-2}$ holds for any $x \in [0,1]$. Thus, when $2m \le M$ (i.e. $n \ge m$), we can have

$$H(m,p) \le 2^{-2m}\omega(p) \cdot \int_{\bar{p}}^{p} x^{n-m}(x-1+p)\mathrm{d}x \quad (37)$$

where the equality only holds when $m = 0$.

$$\int_{\bar{p}}^{p} x^{n-m}(x-1+p)\mathrm{d}x = \frac{(2p-1)p^{\delta}}{\delta} + \frac{\Delta}{\delta(\delta+1)} \quad (38)$$

where $\delta = n - m + 1$ and $\Delta = \bar{p}^{\delta+1} - p^{\delta+1} < 0$. Hence,

$$\log\frac{H(m,p)}{H(0,p)} \le -2m[\log(2p) - \varepsilon(m)] + O\left(\frac{1}{M}\right) \quad (39)$$

where $\varepsilon(m) = -(2m)^{-1}[\log(n-m+1) - \log(M+1)]$. Since $\log(x)$ is a concave function, we can know that

$$\varepsilon(m) \le (M)^{-1}\log(M+1) = O\left(M^{-1}\right) \quad (40)$$

which concludes the first case in Proposition 8. Similarly, for $2m > M$ (i.e. $n < m$), we can have

$$\log\frac{H(m,p)}{H(M,p)} \le -2n[\log(2p) - \hat{\varepsilon}(n)] + O\left(\frac{1}{M}\right) \quad (41)$$

where $\hat{\varepsilon}(n) \le O(M^{-1})$. Thereby, we can conclude the second case of Proposition 8. Note that the case where $p < 0.5$ can be derived by using Proposition 4. $\square$

For the case that $t = 0$ and $M \gg 1$, using the same method as the above proof, we can derive the same lower and upper bounds as Propositions 8 and 7. On the other hand, for $t = 0$, Proposition 6 does not hold and we can have

Proposition 9. $H(m,p;M,0) = H(n,p;M,0)$

Proof. When $t = 0$,

$$H(m,p) = 2^{M+1}(2p-1)^{-1}\int_{\bar{p}}^{p} x^n(1-x)^m\mathrm{d}x. \quad (42)$$

Then, substituting x as $1 - v$ concludes Proposition 9. $\square$

4.3. Reinforcement mechanism analysis

According to Proposition 2, when $\mathbb{E}[m/M]$ approaches 0, $\tilde{p}_i \approx p_i$. In this case, we can have

$$\log \frac{P(L(j)=1)}{P(L(j)=2)} \approx \lambda_0 + \sum\nolimits_{i=1}^{N} (2\delta_{ij1} - 1)\lambda_i \quad (43)$$

where $\lambda_0 = \log(\tau_1/\tau_2)$ and $\lambda_i = \log(p_i/\bar{p}_i)$. For worker i, we assume that all other workers report truthfully and exert high efforts. Suppose the real true label is 1. In order to ensure $\mathbb{E}[m/M]$ to approach 0, the probability ratio in Equation 43 must be positive with almost 1.0 probability. Thus, we can directly discard the absolute operation in Equation 14 and calculate the expected value of task j as

$$\mathbb{E}_1 v(j) \approx \lambda_0 + (N-1)(2p_H - 1)\lambda_H + (2p_i - 1)\lambda_i. \quad (44)$$

Similarly, if the real true label is 2, then

$$\mathbb{E}_2 v(j) \approx -\lambda_0 + (N-1)(2p_H - 1)\lambda_H + (2p_i - 1)\lambda_i. \quad (45)$$

Thus, the average task value v satisfies

$$\begin{aligned} &\mathbb{E}v = \tau_1 \mathbb{E}_1 v(j) + \tau_2 \mathbb{E}_2 v(j) \\ &= (2\tau_1 - 1)\lambda_0 + (N-1)(2p_H - 1)\lambda_H + (2p_i - 1)\lambda_i. \end{aligned} \quad (46)$$

Suppose the true label is 1.

$$x = \log \frac{P(L=1)}{P(L=2)} = g + \sum_{i=1}^{N} f(x_i, y_i, w_i, z_i) \quad (47)$$

where $g = g_1 - g_2$, and

$$g_1 = \log(s_1 + t_2 + 1)\ ,\ g_2 = \log(s_2 + t_1 + 1).$$

Omitting the subscript in f, we can have $f = f_1 - f_2$ with probability p and $f = f_2 - f_1$ with probability $1-p$. Here,

$$f_1 = \log(x + z + 2)\ ,\ f_2 = \log(w + y + 1).$$

Thus, we can have

$$\mathbb{E}g = \mathbb{E}g_1 - \mathbb{E}g_2\ ,\ \mathbb{E}f = (2p-1)(\mathbb{E}f_1 - \mathbb{E}f_2). \quad (48)$$

From the previous proof, we know that $P(m)$ is very small when $m >> 1$. Thus, we mainly focus on the region where m is relatively small. For a given small m,

$$\mathbb{E}_{s_1,t_2} g_1 \approx \mathbb{E}_{t_2} \log(np + t_2 + 1) \quad (49)$$

$$\log(np + t_2 + 1) = \log(np + 1) + \sum_{i=1}^{\infty} (-1)^{i-1} q^i \quad (50)$$

$$q = \frac{t_2}{np+1} \Rightarrow 0 \le q^i \le c^i \cdot \left(\frac{m}{M}\right)^i \le c^i \frac{m}{M} \quad (51)$$

Then,

$$\mathbb{E}g_1 \approx \mathbb{E}_m \log(1 + Mp - mp) \quad (52)$$

$$\log(1 + Mp - mp) \approx \log(1 + Mp) + \sum_{i=1}^{\infty} (-1)^i \left(\frac{m}{M}\right)^i \quad (53)$$

Using the similar way of approximation for the computation of $\mathbb{E}g_2$, $\mathbb{E}f_1$ and $\mathbb{E}f_2$, we can have

$$\begin{aligned} \mathbb{E}g_1 \approx \log(Mp)\ ,\ \mathbb{E}g_2 \approx \log(M(1-p)) \\ \mathbb{E}f_1 \approx \log(Mp)\ ,\ \mathbb{E}f_2 \approx \log(M(1-p)) \end{aligned} \quad (54)$$

Thus, if all workers exert high efforts and report truthfully,

$$\mathbb{E}x \approx \log \lambda_0 + \sum\nolimits_{i=1}^{N} \log \lambda_{i,H} \quad (55)$$

If the true label is 2, then

$$\mathbb{E}x \approx \log \lambda_0 - \sum\nolimits_{i=1}^{N} \log \lambda_{i,H} \quad (56)$$

Thereby,

$$\mathbb{E}|x| \approx (2p_0 - 1)\log \lambda_0 + \sum\nolimits_{i=1}^{N} \log \lambda_{i,H} \quad (57)$$

When, for example, worker 1 deviate from the desired equilibrium strategy, the non-equilibrium state correspond to

$$\mathbb{E}|x'| \approx (2p_0 - 1)\log \lambda_0 + \log \lambda_1 + \sum\nolimits_{i=2}^{N} \log \lambda_{i,H} \quad (58)$$

The minimal value of $\mathbb{E}|x'|$ is reached when worker 1 exert high efforts and report falsely, namely $\log \lambda_i = -\log \lambda_{i,H}$. Thus, the maximal reward increment brought by worker 1's strategy switch is

$$V_1 = F(\mathbb{E}x) - F(\mathbb{E}x') \approx 2\log \lambda_{1,H} \cdot \left.\frac{\mathrm{d}F}{\mathrm{d}x}\right|_{x=x_H} \quad (59)$$

Since p_0 is difficult to estimate, we define the upper bound of the value increment as

$$V = 2 \max_i \log \lambda_{i,H} \cdot \max_{x \in [x_H, \infty)} \frac{\mathrm{d}F}{\mathrm{d}x} \quad (60)$$

where $x_H = \log \lambda_0 + \sum_{i=1}^{N} \log \lambda_{i,H}$. Considering the discounted reward calculation in reinforcement learning, we can know the maximum value difference can be created by the manipulation of any worker is $(1-\rho)^{-1}V$. Meanwhile, if the reinforcement part increases the scaling factor by δ to obtain the reward increment, we need to pay more than

$\sum_{i=1}^{N} M\delta(p_{i,H} - 0.5)$. Thus, if we want to prevent the reinforcement learning module from the adversarial manipulation, the minimal gap δ between to two available scaling factors should satisfy

$$\sum_{i=1}^{N} M\delta(p_{i,H} - 0.5) > V. \tag{61}$$

References

Xi Chen, Qihang Lin, and Dengyong Zhou. Statistical decision making for optimal budget allocation in crowd labeling. *Journal of Machine Learning Research*, 16:1–46, 2015.

Anirban Dasgupta and Arpita Ghosh. Crowdsourced judgement elicitation with endogenous proficiency. In *Proc. of WWW*, 2013.

Alexander Philip Dawid and Allan M Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied statistics*, pages 20–28, 1979.

Yang Liu and Yiling Chen. Sequential peer prediction: Learning to elicit effort using posted prices. In *AAAI*, pages 607–613, 2017.

Qiang Liu, Jian Peng, and Alexander T Ihler. Variational inference for crowdsourcing. In *Proc. of NIPS*, 2012.

Dimitris G Tzikas, Aristidis C Likas, and Nikolaos P Galatsanos. The variational approximation for bayesian inference. *IEEE Signal Processing Magazine*, 25(6):131–146, 2008.

Yudian Zheng, Guoliang Li, Yuanbing Li, Caihua Shan, and Reynold Cheng. Truth inference in crowdsourcing: is the problem solved? *Proc. of the VLDB Endowment*, 10(5):541–552, 2017.