A Novel Strategy for Active Task Assignment in Crowd Labeling

Zehong Hu, Jie Zhang
Rolls-Royce@NTU Corporate Lab, School of Computer Science and Engineering
Nanyang Technological University, Singapore
huze0004@e.ntu.edu.sg

Abstract

Active learning strategies are often used in crowd labeling to improve the task assignment. However, these strategies, which evaluates each possible assignment at first and then greedily selects the optimal one, may require prohibitive computation time but still cannot improve the assignment to the utmost. Thus, we develop a novel strategy by firstly deriving an efficient algorithm for assignment evaluation. Then, to overcome the uncertainty of labels, we modulate the scope of the greedy task assignment with the posterior uncertainty and keep the evaluation being optimistic. The experiments on four popular worker models and four MTurk datasets show that our strategy achieves the best performance and highest computation efficiency.

1 Introduction

Crowd labeling, which uses crowdsourcing to generate labels for large-scale datasets, has brought significant benefits in many domains [Snow *et al.*, 2008; Deng *et al.*, 2009; Slivkins and Vaughan, 2014; Hu and Zhang, 2017]. However, since labeling tasks are usually tedious and workers are often non-experts, the resulting labels can be very noisy [Hua *et al.*, 2013; Chen *et al.*, 2015]. As a remedy, many crowd labeling markets follow the round-robin repeated labeling strategy which randomly assigns multiple workers with a same task and all tasks are assigned with a same number of workers [Sheng *et al.*, 2008]. In principle, as long as an enough number of labels are collected, this repeated labeling strategy can generate labels with high accuracy. However, workers should be paid for each label they provide. The cost is nontrivial. In addition, the round-robin strategy overlooks the difference between different tasks and workers, and thus will waste lots of budget on easy tasks and low-quality workers. Thereby, it is very crucial to look into a better way of allocating budget among tasks and workers so that we can use less number of labels to generate high-accuracy labels.

In many previous studies, researchers propose to actively select tasks and workers at each step based on the online inference of true labels and worker models. By doing so, more budget can be allocated to more difficult tasks and more reliable workers, and then the label accuracy gets boosted. In this paper, we call this way of task assignment as active task assignment. The study on active task assignment starts from Sheng *et al.* [2008]. They assume workers to be homogeneous and infer the true labels of tasks using majority voting. Their task assignment relies on the uncertainty sampling strategy which randomly selects a worker to label the most uncertain task at each step [Lewis and Catlett, 1994]. Then, Welinder and Perona [2010] consider heterogeneous workers. They use the variational inference to estimate true labels and worker models. Besides, they add one more step in uncertainty sampling to exclude those must-be-bad workers.

Later on, Simpson and Roberts [2015] and Muhammadi *et al.* [2015] employ another more advanced strategy, expected error reduction [Settles, 2012]. Compared with uncertainty sampling which purely relies on the current inference results, the expected error reduction evaluates the benefits of all possible next-step assignments. At each step, the most beneficial assignment is selected. In principle, this prediction-based strategy can achieve higher accuracy. However, to compute the benefits, it needs to run the inference algorithm for all possible future labels. For a market with hundreds of tasks and workers, the computation cost of this strategy can easily become prohibitive. To alleviate the computation cost, Zheng *et al.* [2015] force the estimates of worker models to be fixed and approximately update the estimates of true labels with the Bayes' theorem. Nevertheless, this approximation may bring large errors since it neglects the correlation between the estimates of true labels and worker models.

Both uncertainty sampling and expected error reduction come from the active learning studies which assume there is one fully reliable annotator [Settles, 2012]. However, crowd labeling only has many possibly unreliable workers. Considering the uncertainty of labels, Raykar and Agrawal [2014] employ MDP to model the labeling process and solve the MDP by using the classic ϵ-greedy algorithm. Then, Chen *et al.* [2015] propose a optimistic greedy strategy to solves the MDP which replace the expected values with the upper confidence bounds. Their studies show that this simple greedy strategy can outperform various MDP solving algorithms because the state and action spaces are too large in crowd labeling. However, in crowd labeling, both the estimates of true labels and worker models have a level of uncertainties, and even worse, these uncertainties are mutually reinforced. An upper confidence bound cannot cover all these uncertainties.

In this paper, we propose a novel active task assignment strategy which solves the low efficiency of assignment evaluation and improves the label accuracy by taking all the uncertainties into consideration. More specifically, to boost the computation efficiency, we firstly derive the first-order approximation equations of the inference algorithm developed for crowd labeling. After overcoming the overshoot problem, we efficiently solve the approximation equations with Newton's method. Then, based on the expected error reduction strategy, to suppress the mutual reinforcement between the uncertainties of true labels and worker models, we modulate the scope of task assignment based on the posterior measurement of uncertainties. Meanwhile, for the multidimensional distribution of labels, we employ the conditional value-at-risk to optimistically evaluate the benefits of each assignment. We conduct extensive experiments based on four popular worker models and four MTurk datasets. The empirical results show that our strategy not only requires the least labels for high label accuracy but also achieves the highest computation efficiency among all existing prediction-based strategies.

2 Problem Formulation

In the push market of crowd labeling, there is one data requester who employs M workers to label N tasks. Tasks may belong to different classes numbered from 1 to K. $y_i \in \{1, \dots, K\}$ denotes the true label of task i, and vector $\boldsymbol{y} = [y_i]_N$ represents the true labels of all tasks. At each time step t, the data requester needs firstly decide which task to be assigned to which worker and then collects the label from the worker. We denote the task assignment at step t as the task-worker pair $\langle i^t, j^t \rangle$. Besides, we write the labels collected up to step t as a matrix $S^t = [s_{ij}]_{N \times M}$, where $s_{ij} \in \{0, 1, \dots, K\}$ and 0 means task i has not been labeled by worker j. After obtaining the label, the data requester updates the label matrix S^t at first and then the true label estimates $\tilde{y}_i^t$ and worker models based on the label matrix S^t. So, the accuracy $A(t)$ of our strategy is calculated as

$$A(t) = \frac{1}{N} \sum\nolimits_{i=1}^{N} 1(\tilde{y}_i^t = y_i). \tag{1}$$

Besides, in this paper, we adopt a common assumption for crowd labeling that acquiring a label incurs a fixed unit cost [Lin *et al.*, 2016]. In this case, the objective of our task assignment strategy is to maximize the growth rate of accuracy $A(t)$ as the number of labels t. By doing so, we can minimize the labels needed for achieving high label accuracy and thus save money for the data requester.

To achieve this objective, we firstly needs to decide how to model workers' labeling behaviors. In the existing studies, the following two worker models are widely-adopted:

- One-coin model [Welinder and Perona, 2010; Chen *et al.*, 2015] uses a single number $p_j \in [0, 1]$ to denote the probability that worker j correctly labels a task.
- K-coin model [Simpson and Roberts, 2015; Zheng *et al.*, 2017] models worker j with the confusion matrix $C_j = [c_{jkg}]_{K \times K}$, where $c_{jkg} \in [0, 1]$ denotes the probability that worker j labels a task in class k as class g.

Actually, the one-coin model is equivalent to requiring the confusion matrix to satisfy $c_{jkk} = c_{jgg} = p_j$ and $c_{jks} = c_{jgt} = 1 - p_j$ for any $k \neq s$ and $g \neq t$. To uniformly represent the above two models, we denote these constraints by the $K \times K$ basis matrix B_{wl} of which each element is 0 or 1. If $B_{wl}(k, g) = B_{wl}(s, t) = 1$, then $c_{jkg} = c_{jst}$ must hold for all workers. For the one-coin model, there are two basis matrices satisfying $B_{11} = \boldsymbol{I}_{K \times K}$ and $B_{12} = \mathbf{1}_{K \times K} - B_{11}$, where $\boldsymbol{I}$ is the identity and all-ones matrices, respectively. For the confusion matrix, there are $K \times K$ basis matrices satisfying $B_{wl}(w, l) = 1$ for $w, l \in [K]$ and $B_{wl}(k, g) = 0$ for any $k \neq w$ or $g \neq l$. Thus, we can uniformly represent workers by the confusion matrix satisfying:

$$c_{jkg} = \sum\nolimits_{w} \sum\nolimits_{l} \frac{\theta_{jwl}}{||B_{wl}||} B_{wl}(k, g) \tag{2}$$

where $\theta_{j11} = p_j$ in the one-coin model and $\theta_{jwl} = c_{jwl}$ when the K-coin model is used. Then, similar as [Chen *et al.*, 2015], we can assume the probability vector $\boldsymbol{\theta}_{jw} = [\theta_{jwl}]$ to follow the Dirichlet prior $\mathrm{Dir}(\boldsymbol{\alpha}_w^0)$, where $\boldsymbol{\alpha}_w^0 = [\alpha_{wl}^0]$. By using the mean-field variational inference [Liu *et al.*, 2012], we can compute the posterior distribution at step t as

$$\alpha_{jwl}^t = \sum\nolimits_{i=1}^{N} \sum\nolimits_{k=1}^{K} \sum\nolimits_{g=1}^{K} \delta_{ijg}^t \xi_{kgwl} q_{ik}^t + \alpha_{wl}^0 \tag{3}$$

$$\begin{aligned} \log q_{ik}^t = & \sum\nolimits_{j=1}^{M} \sum\nolimits_{g=1}^{K} \sum\nolimits_{w=1}^{W} \sum\nolimits_{l=1}^{L} \delta_{ijg}^t \xi_{kgwl} \Big[\psi(\alpha_{jwl}^t) \\ & - \psi\big(\sum\nolimits_{l=1}^{L} \alpha_{jwl}^t\big) - \log(||B_{wl}||) \Big] + \varepsilon_i \end{aligned} \tag{4}$$

where $q_{ik} = p(y_i = k)$ denotes the posterior distribution of true labels.$\delta_{ijg}^t = 1(s_{ij}^t = g)$ and $\xi_{y_i gwl} = B_{wl}(y_i, g)$. $\psi(\cdot)$ represents the digamma function and ε_i denotes the normalization constant used to keep $\sum_k q_{ik} = 1$. According to the previous studies [Liu *et al.*, 2012; Chen *et al.*, 2015], we should be optimistic—i.e. believing workers have higher probability to be correct. Here, we denote the priors corresponding to the correct- and wrong-label-reporting components by $\alpha_{w^+l^+}^0$ as $\alpha_{w^-l^-}^0$, respectively. Mathematically, there exists at least one k that ensures $B_{w^+l^+}(k, k) = 1$. To be optimistic, we can set $\alpha_{w^-l^-}^0 = 1$ while keeping $\alpha_{w^+l^+}^0 > 1$. On the other hand, $\alpha_{w^+l^+}^0 \ll N$ is needed so that the dominance of the first item in the right-hand side of Equation 3 can be ensured. Our empirical investigation shows that the value changes of $\alpha_{w^+l^+}^0$ in the range discussed above almost have no effects on the inference results. Thus, we keep $\alpha_{w^+l^+}^0 = 4$ in this paper, which is the same as the setting of [Chen *et al.*, 2015] on the one-coin model.

To summarize, we formally present the task assignment process in Algorithm 1. At each step t, we decide the task assignment, namely the task-worker pair $\langle i_t, j_t \rangle$, based on the current posterior distributions at first. After collecting a new label, we update the label matrix and the posterior distributions by iterating Equations 6 and 7 until convergence. When the required T labels are collected, we decide the true labels by using the classic maximum a posteriori probability rule (line 5). Then, the only remaining question is how to use the posterior distributions to guide the task assignment, and we will present our strategy, uncertainty modulated optimistic assignment (UMOA), in the next section.

Algorithm 1: Active Task Assignment

```
Input: the number of labels $T$, the basis matrices $B_{wl}$
Output: the inferred true labels $\boldsymbol{y}^T$
for $t = 0$ to $T - 1$ do
    $\langle i_t, j_t \rangle \leftarrow$ UMOA($q^t_{ik}, \alpha^t_{jwl}, S^t, \mathcal{B}$)
    $S^{t+1} \leftarrow$ Get the label and update the label matrix $S^t$
    $\langle q^{t+1}_{ik}, \alpha^{t+1}_{jwl} \rangle \leftarrow$ Iterate Eq. 3 and 4 until convergence
$y^T_i \leftarrow \arg\max_k q^T_{ik}$   (Maximum a Posterior Probability)
```

3 Task Assignment Strategy

Since the real true labels are unknown, the accuracy $A(t)$ defined in Equation 1 cannot be computed. Instead, we calculate the expectation of label accuracy based on the estimated posterior distribution of true labels as

$$\mathbb{E}A(t) = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}\left[1(y^t_i = y_i)\right] = \frac{1}{N}\sum_{i=1}^{N}\max_{k\in[K]} q^t_{ik}. \quad (5)$$

Suppose worker z provides label λ for task x at step t. Then, the expected accuracy increment or error reduction brought by this new label can be calculated as

$$I(t, x, z, \lambda) = \frac{1}{N}\sum\nolimits_{i=1}^{N}\left(\max\nolimits_k q^{t+1}_{ik} - \max\nolimits_k q^t_{ik}\right). \quad (6)$$

The classic expected error reduction strategy greedily selects the assignment $\langle x, z\rangle$ that maximizes $\mathbb{E}_\lambda I(t, x, z, \lambda)$, where the distributions of λ is computed based on q^t_{ik} and α^t_{jwl}. Nevertheless, when applying this simple greedy strategy, we observe the following two drawbacks.

- **Efficiency Bottleneck**: To find the optimal task assignment, we need to traverse all possible pairs of tasks and workers. In other words, we need to compute $O(MN)$ times of $\mathbb{E}_\lambda I(t, x, z, \lambda)$. For every λ, we need to solve Equations 3 and 4 to compute q^{t+1}_{ik} in Equation 6. Solving Equations 3 and 4 requires to repeatedly go through all tasks and workers, and the time complexity is around $O(MN)$. Thereby, computing all the predictions requires $O(M^2N^2)$ time costs, which significantly lowers the computation efficiency because there are usually hundreds of tasks and workers in a market.
- **Uncertainty Reinforcement**: Since we are uncertain about the true labels, we cannot accurately infer worker models. The uncertainty of worker models will in turn cause us to be more uncertain about the true labels. This reinforcement process amplifies the uncertainties of the inferred true labels and worker models. Thereby, if we purely rely on Equation 6 to guide the task assignment, we may make severe mistakes. For example, after obtaining the first label, the inference algorithm will trust the label because of the optimistic priors. This inferred true label will increase our confidence about the worker's ability to provide correct labels. Then, we will choose this "trusted" worker to label a new task because new tasks have the minimal $\max_k q^t_{ik}$, and this worker becomes more "trusted". Finally, we use this "trusted" worker to label all tasks at first and then greedily select workers who can provide the same label as this worker. This way of task assignment is obviously not reasonable.

In this section, we first overcome the efficiency bottleneck by developing an efficient algorithm to predict q^{t+1}_{ik}. Its time complexity is independent of M and N. Then, we suppress the uncertainty reinforcement via introducing the posterior uncertainty measurements and the conditional value-at-risk to guide our assignment. After these modifications, we formally write our task assignment strategy as Equation 16.

3.1 Efficient Prediction

In Equations 3 and 4, the newly obtained label $s_{xz} = \lambda$ only causes the change that $\delta^t_{xz\lambda} = 0 \rightarrow \delta^{t+1}_{xz\lambda} = 1$. When iteratively solving Equations 3 and 4, since we use the estimates at step t as the starting values of step $t+1$, this change firstly affects the variational variables of task x and worker z. Through the second round of iteration, the variational variables of the workers who have labeled task x and the tasks that have been labeled by worker z will then be updated. This process repeats, and finally the variational variables of all workers and tasks will be updated. To achieve efficient prediction, we focus on the direct effects of s_{xz} on task x and worker z and neglect its high-order effects on other tasks and workers. In other words, when predicting the next-step values of variational variables, we assume $q^{t+1}_{ik} \approx q^t_{ik}$ for $i \neq x$ and $\alpha^{t+1}_{jwl} \approx \alpha^t_{jwl}$ for $j \neq z$. Despite this assumption, solving Equations 3 and 4 for prediction still needs to repeatedly go through all tasks and workers. To overcome this efficiency bottleneck, we construct a new set of equations to directly solve the differences of q_{xk} and α_{zwl} between step t and $t+1$. Firstly, we define the differences as

$$\phi^t_{xk} = q^{t+1}_{xk} - q^t_{xk}, \quad \varphi^t_{zwl} = \alpha^{t+1}_{zwl} - \alpha^t_{zwl}. \quad (7)$$

Then, substituting Equation 3 into Equation 7, we can have

$$\varphi^t_{zwl} = \sum\nolimits_{k=1}^{K}\xi_{k\lambda wl}(\phi^t_{xk} + q^t_{xk}). \quad (8)$$

Since $\sum_k q^{t+1}_{xk} \leq 1$, $0 \leq \varphi^t_{zwl} \leq 1$. Meanwhile, in Equation 3, α^t_{jwl} will become far larger than 1 as the number of labels increases. Thus, for the relatively small φ^t_{zwl}, we can know $\psi(\alpha^{t+1}_{zwl}) - \psi(\alpha^t_{zwl}) \approx \zeta(\alpha^t_{zwl})\varphi^t_{zwl}$, where $\zeta(\cdot)$ denotes the trigamma function and satisfies $\zeta(x) = \sum_{i=0}^{\infty}(x+i)^{-2}$ [Mező, 2013]. Then, substituting Equations 4 and 8 into Equation 7, we can have

$$\log\left(1 + \frac{\phi^t_{xk}}{q^t_{xk}}\right) - \sum\nolimits_{g=1}^{K} T_{kg}\cdot\phi^t_{xg} + \tilde{\varepsilon}_x \approx H(k) \quad (9)$$

where $T_{kg} = \sum_{w=1}^{W}\sum_{l=1}^{L}\tau_{zwl}\cdot\xi_{g\lambda wl}$ and

$$\tau_{zwl} = \xi_{k\lambda wl}\cdot\zeta(\alpha^t_{zwl}) - \sum\nolimits_{q=1}^{K}\xi_{k\lambda wq}\cdot\zeta(\sum\nolimits_{l=1}^{L}\alpha^t_{zwl})$$

$$H(k) = \sum\nolimits_{w=1}^{W}\sum\nolimits_{l=1}^{L}\sum\nolimits_{g=1}^{K}\tau_{zwl}\cdot\xi_{g\lambda wl}\cdot q^t_{xg} +$$

$$\sum_{w=1}^{W}\sum_{l=1}^{L}\xi_{k\lambda wl}\left[\psi(\alpha^t_{zwl}) - \psi(\sum_{l=1}^{L}\alpha^t_{zwl}) - \log||B_{wl}||\right].$$

The relaxation variable $\tilde{\varepsilon}_x$ ensures the probability sum of all label values to always be 1, namely

$$\sum\nolimits_{k=1}^{K} q_{xk}^{t+1} - \sum\nolimits_{k=1}^{K} q_{xk}^{t} = \sum\nolimits_{k=1}^{K} \phi_{xk}^{t} = 0. \tag{10}$$

In this way, we convert the prediction of the variational variables as solving $K+1$ logarithmic equations.

The most efficient method to solve Equations 9 and 10 is Newton's method [Burden and Faires, 2004]. However, the overshoot of Newton's method may cause the essential condition $\phi_{xk}^{t} \geq -q_{xk}^{t}$ in Equation 9 to be violated. Thus, we eliminate the non-negative condition of the logarithmic function via defining a new variable as $\sigma_{xk}^{t} = \log\left(1 + \phi_{xk}^{t}/q_{xk}^{t}\right)$. Then, we can rewrite Equations 9 and 10 as $\boldsymbol{F} = \boldsymbol{0}$, where the vector $\boldsymbol{F} = [F(k)]_{K+1}$. For $k = 1, \ldots, K$,

$$\begin{aligned} F(k) &= \sigma_{xk}^{t} - \sum\nolimits_{g=1}^{K} T_{kg} \cdot q_{xg}^{t} \cdot (e^{\sigma_{xg}^{t}} - 1) + \tilde{\varepsilon}_x - H(k) \\ F(K+1) &= \sum\nolimits_{k=1}^{K} q_{xk}^{t} \cdot (e^{\sigma_{xk}^{t}} - 1). \end{aligned} \tag{11}$$

Applying Newton's method to solve the $K+1$ exponential equations $\boldsymbol{F} = \boldsymbol{0}$, we can efficiently solve σ_{xk}^{t} and then compute the estimates of the expected accuracy increment as

$$\tilde{I}(t, x, z, \lambda) = (\max\nolimits_k q_{xk}^{t} \cdot e^{\sigma_{xk}^{t}} - \max\nolimits_k q_{xk}^{t})/N. \tag{12}$$

3.2 Task Assignment

To suppress the reinforcement between the uncertainties of true labels and worker models, we need to set an upper bound for the posterior uncertainty of the inferred true labels. When the number of labels is very small, the uncertainty upper bound should be very low to prevent the misjudgment of true labels and the corresponding worker models. If the upper bound is not reached, we will force the next-step task assignment to stay at the same task to ensure accurate estimates of true labels and worker models. We call this period as the exploration stage. When the number of labels is large enough, few wrong labels cannot cause us to make mistakes in the inference of worker models. In this stage, to boost the label accuracy to the utmost, we should be able to freely select the task assignment that can bring the largest expected accuracy increment, which requires a high uncertainty upper bound. Thus, we introduce the logistic function and formally write the uncertainty upper bound of the inferred true labels as

$$1 - \max\nolimits_{k \in [K]} q_{i_{t-1}k}^{t} \leq \frac{K-1}{K} \cdot \frac{1}{1 + \exp\left[-a(t-b)\right]} \tag{13}$$

where the left-hand side denotes the posterior uncertainty of task i_{t-1}. On the right-hand side, the coefficient $\frac{K-1}{K}$ is used because the minimal value of $\max_{k \in [K]} q_{i_{t-1}k}^{t}$ can be theoretically proven to be $1/K$. Besides, the uncertainty parameters, a and b, adjust the length of our exploration stage, respectively. We will empirically decide the values of these two parameters in the next section. Based on this upper bound, we can define the scope of task assignment at step t as

$$Q(t) = \begin{cases} \{\langle i, j\rangle | i = i_{t-1}, s_{ij}^{t} = 0\} & \text{Cond.1 \& 2} \\ \{\langle i, j\rangle | s_{ij}^{t} = 0\} & \text{Otherwise} \end{cases} \tag{14}$$

where condition 1 denotes the violation of Equation 13. Condition 2 denotes that the number of workers who have labeled task i_{t-1} is smaller than M. In other words, there are still some workers for task i_{t-1} to choose. Besides, at step $t = 0$, we use the convention that $i_{-1} = 1$ for the first task.

In addition, the obtained label λ from worker z can be any value in $\{1, \ldots, K\}$. Considering the uncertainty of online inference, we use the upper confidence bound to replace the expectation in the expected error reduction strategy. For the K-dimensional distribution over K possible values, the upper confidence of the expected accuracy increment, $\tilde{G}(t, x, z)$, can be computed as the conditional value-at-risk which satisfies [Rockafellar and Uryasev, 2002]:

$$\begin{aligned} &\tilde{G}(t, x, z) = \max\nolimits_{q_\lambda \geq 0, \lambda \in [K]} \tilde{I}(t, x, z, \lambda) \cdot q_\lambda \\ &\qquad \text{s.t.} \quad q_\lambda \leq p(s_{xz}^{t+1} = \lambda)/\gamma(t), \quad \sum\nolimits_{\lambda} q_\lambda = 1 \end{aligned} \tag{15}$$

where $p(s_{xz}^{t+1} = \lambda) = \sum_k q_{xk}^{t} c_{jk\lambda}^{t}$ denotes the probability that the obtained label is λ. Besides, $\gamma(t)$ denotes the risk level, and $1 - \gamma$ equals the required confidence level. We will empirically decide the $\gamma(t)$ function in the next section. Lastly, our task assignment strategy, uncertainty modulated optimistic assignment (UMOA), decides the task assignment at step t by the following equation:

$$\langle i_t, j_t \rangle = \arg\max\nolimits_{\langle x, z\rangle \in Q(t)} \tilde{G}(t, x, z). \tag{16}$$

4 Experimentation

In this section, we firstly compare the computation efficiency and label accuracy of different task assignment strategies by employing the four worker models and four Mechanical Turk datasets. Then, we explain how to set the risk level function $\gamma(t)$ and the uncertainty parameters a and b. Note that, since our strategy is very robust to parameter values, we actually use the same parameter settings in all the experiments.

4.1 Computation Efficiency Comparison

Table 1 compares the average one-step time costs of our strategy and the following benchmarking strategies:

- Round-Robin strategy [Sheng *et al.*, 2008] randomly assigns the same number of workers for all tasks;
- Uncertainty sampling [Welinder and Perona, 2010] selects the most uncertain task at each step and excludes must-be-bad workers whose correction rate and the variance are lower than 0.55 and 0.05^2 respectively;
- Expected error reduction [Muhammadi *et al.*, 2015] greedily selects the task assignment that can bring the maximum one-step expected accuracy increment;
- Optimistic KG [Chen *et al.*, 2015] calculates the the optimistic knowledge gradient (KG) as the maximum expected accuracy increment of all possible next-step labels and selects the one with the highest optimistic KG;
- Bayesian approximation [Zheng *et al.*, 2015] fixes the learned worker models, uses Bayes' theorem to predict the one-step expected accuracy increment, and selects the assignment with the maximal increment at each step.

In Table 1, the brackets (N, M) denote that the numbers of tasks and workers are N and M, respectively. Our experiment

Table 1: Computation time of different strategies

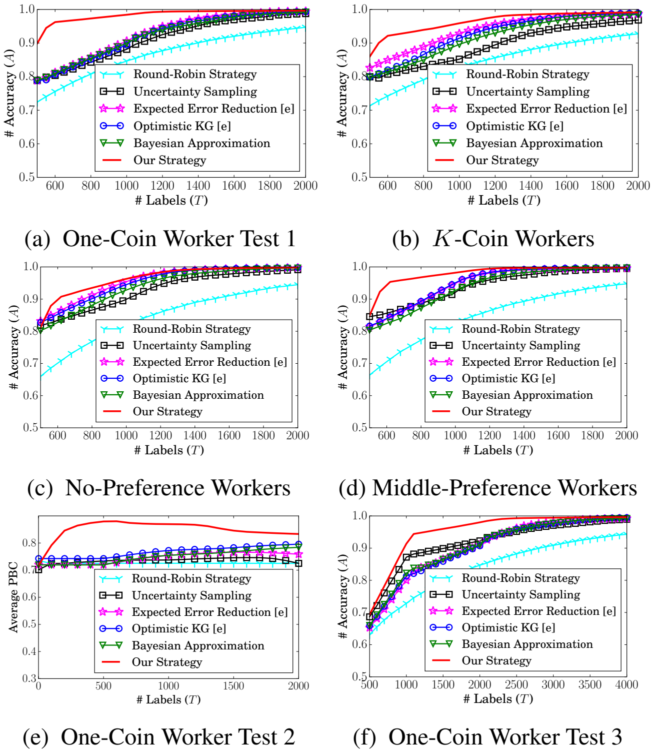
Task Assignment Strategies	**One-Step Time Cost (ms)**		
	$(20, 10)$	$(60, 10)$	$(60, 20)$
Round-Robin Strategy	0.7	0.8	0.8
Uncertainty Sampling	0.9	1.2	0.9
Expected Error Reduction	**24**	**247**	**431**
Optimistic KG	**21**	**180**	**436**
Bayesian Approximation	**3.1**	**8.3**	**16**
Our Strategy	**1.5**	**3.2**	**5.6**

settings here are the same as that in Figure 1a, which will be detailed later. The time cost is estimated via running 100 rounds of experiments on Xeon CPU E5-1650 and collecting $M \cdot N$ labels in each round of experiments. From Table 1, we can conclude that our strategy is the most efficient among all prediction-based task assignment strategies (i.e. the last ones). The time complexity of our strategy, which uses the efficient prediction algorithm, is lower than $O(NM)$. By contrast, the time complexity of the expected error reduction and optimistic KG strategies is around $O(N^2M)$. In our following experiments with hundreds of tasks and thousands of labels to collect, the computation time of these two strategies is prohibitive. Thus, we also use our efficient prediction algorithm to improve the efficiency of these two strategies and mark them with [e] for distinction in the following sections. Besides, the computation efficiency of the Bayesian approximation strategy is acceptable because it approximately predicts the one-step expected accuracy increment by fixing workers' models. However, this approximation neglects the correlation between the estimates of worker models and true labels, which will degrade the label accuracy. We will empirically show this point in the next two sections.

4.2 Accuracy Comparison on Synthetic Data

To show the advantage of our strategy on accuracy growth rate, we conduct experiments on the one- and K-coin models. Besides, to increase the difficulty of experiments, we also introduce the following two worker models:

- No-Preference Model [Welinder and Perona, 2010] assumes that workers have no preference over any label value but workers' labeling behaviors depend on the true labels of tasks. In other words, workers will choose the two possible wrong labels with the same probability, which is equivalent to requiring $C_j(1,2) = C_j(1,3)$, $C_j(2,1) = (2,3)$ and $C_j(3,1) = C_j(3,2)$.
- Middle-Preference Model [Rodway *et al.*, 2012] considers the psychological observation that if there are three options, human workers usually have special preference for the middle one. In this case, we can assume that workers have the same labeling behavior for the other two options, which requires $C_j(1,1) = C_j(3,3)$, $C_j(1,2) = C_j(3,2)$ and $C_j(1,3) = C_j(3,1)$.



(a) One-Coin Worker Test 1

(b) K-Coin Workers

(c) No-Preference Workers

(d) Middle-Preference Workers

(e) One-Coin Worker Test 2

(f) One-Coin Worker Test 3

Figure 1: Experiments on four popular worker models

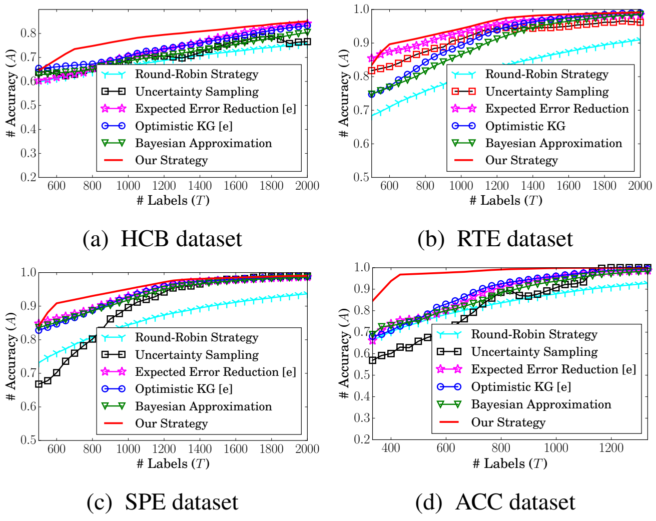
We present all results in Figure 1. Here, we set the number of experiments as 200 so that the differences between different strategies can be distinguished by t-test. For example, the p-values of the t-tests corresponding to $T = 1,000$ in Figure 1b between our strategy and benchmarking strategies are all below 1×10^{-10}. In our experiments, we firstly generate the true labels of tasks via the uniform distribution. Meanwhile, we generate worker models $\boldsymbol{\theta}_{jw}$ by using the Dirichlet distribution $\text{Dir}(\boldsymbol{\alpha}_w)$, where $\alpha_{ww} = 4$ and $\alpha_{wl} = 1$ if $l \neq w$. Then, we compute workers' confusion matrices using Equation 2 and generate labels based on the categorical distribution.

In Figures 1a-d, we set the numbers of tasks and workers as 500 and 10, respectively. Considering the fact that we should collect at least 500 labels for 500 tasks, we start the accuracy comparison from #Labels$=$ 500. Actually, the accuracy of all strategies equals $1/K$ when #Labels$=0$. From these four figures, we can conclude that our task assignment strategy can reach high accuracy (e.g. 0.95) with much less number of labels than all the benchmarking strategies. Meanwhile, Expected Error Reduction [e] and Optimistic KG [e], which combines the classic strategies with our efficient prediction algorithm, also perform better than the other three strategies. On the other hand, the advantage of our strategy on the single-coin model is larger than that on the other three models. The reason for this phenomenon is that, for example, the K rows of the confusion matrix corresponding to the K-coin model are independently generated. Since the true label is also unknown, it becomes difficult to judge which worker is better in this case (e.g. $c_{111} > c_{211}$ while $c_{122} < c_{222}$).

To further validate the advantage of our strategy, we conduct the other two sets of experiments based on Figure 1a. Firstly, in the one-coin model, θ_{j11} denotes worker j's probability of being correct (PBC). In Figure 1e, we set $\theta_{j11} = j * 0.05 + 0.45$ and compute the average PBC of employed workers up to step n as $\sum_{t=1}^{n} \theta_{j_t 11}/n$. The results show that our strategy has the highest average PBC, which means that our strategy successfully identifies and employs higher-quality workers. This observation explains the rationale behind the advantage of our strategy. Secondly, in Figure 1f, we increase the number of tasks to 1000, which requires two times of labels to be collected. Compared with expected error reduction, our strategy reduces #labels to reach 0.95 by 50% in both Figures 1a and f, which reveals the good robustness of our strategy to the scale of collected labels. In fact, as the number of required labels T increases, the ratio between the length of the exploration stage in our strategy and T will decrease, which is helpful to enhance our advantage over benchmarking strategies. However, running 200 experiments is quite time-consuming. Thus, we keep the number of tasks as 500 in our following experiments on MTurk datasets.

4.3 Accuracy Comparison on MTurk Datasets

In addition to using the synthetic data, we employ four popular MTurk datasets as our testbeds in Figure 2. HCB dataset contains the judgments on the relevance between Web pages and search queries [Buckley *et al.*, 2010]. RTE dataset workers need to check whether a hypothesis sentence can be inferred from the provided sentence [Snow *et al.*, 2008]. SPE dataset consists of the positive or negative labels of movie reviews [Pang and Lee, 2005]. ACC dataset workers classify websites according to their adult contents [Ipeirotis *et al.*, 2010]. A challenge of using these datasets is that they are very sparse and the label corresponding to the required assignment may not exist. Thus, we employ the famous SQUARE library (Version 2.0) to complement those non-existent labels [Sheshadri and Lease, 2013]. It uses the true labels to compute workers' confusion matrices at first and then generates workers' labels accordingly. Since the SQUARE library relies on the K-coin worker model, we also use it for the computation of all the strategies. Besides, to facilitate computation, we select the first 500 tasks and 10 workers in the experiments with the first three datasets. The ACC dataset has only 333 tasks with known true labels that have been labeled by workers. Thus, we set the number of tasks as 333 in Figure 2d. From Figure 2, we can conclude that our strategy always significantly outperforms the round-robin, uncertainty sampling and Bayesian approximation strategies. Noted the t-test also supports our conclusion. For example, in Figure 2b, if $T = 1,000$, the p-values of the t tests between our strategies and other strategies are all below 1×10^{-4}. Besides, the advantage of our strategy over the expected error reduction and optimistic KG strategies, which needs to employ our efficient prediction algorithm to solve the efficiency bottleneck, always exists but may become small in some cases. This is because, if the differences between the two rows of the confusion matrix are very large, it will be difficult to distinguish which is the best.



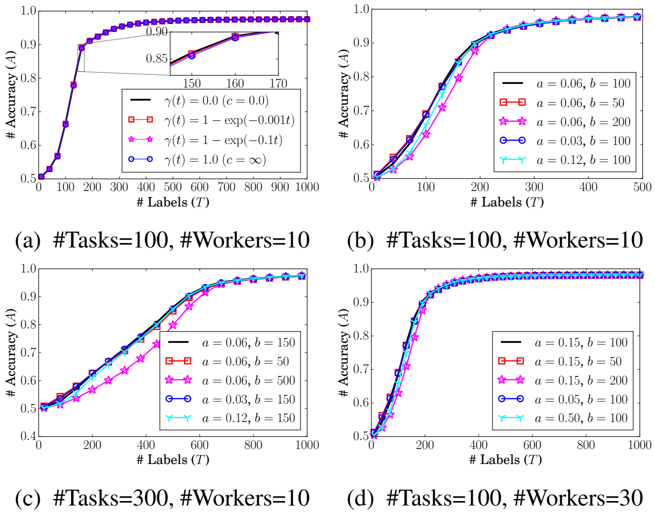
(a) HCB dataset

(b) RTE dataset

(c) SPE dataset

(d) ACC dataset

Figure 2: Experiments on four popular MTurk datasets



(a) #Tasks=100, #Workers=10

(b) #Tasks=100, #Workers=10

(c) #Tasks=300, #Workers=10

(d) #Tasks=100, #Workers=30

Figure 3: Experiments on different parameter settings

4.4 Empirical Study on Parameter Settings

In Figure 3, we compare different settings of the parameters, $\gamma(t)$, a and b, in our strategy. Note that studying the optimal parameter settings should be the first task of our experiments. However, our empirical results show that the performance of our strategy is very robust to different values of parameters. Thus, in all the aforementioned experiments, we directly use the same parameter settings as the first line of Figure 3c without further adjustment. Here, we use the one-coin worker model as our testbed to show the rationale behind our parameter settings. Since parameters are correlated with each other, we actually decide the optimal parameter values by iteratively testing many different combinations on the four models.

Similar to Section 4.2, we randomly generate the true labels and worker models at first and then generate workers' labels based on the categorical distribution. But, instead of using the Dirichlet distribution, we use the the uniform distribution over $[0.5, 1.0]$ to generate p_j in the one-coin work model. In Figure 3a, we employ the function family $\gamma(t) = 1 - \exp(-c \cdot t)$ to test the effects of changing risk level functions $\gamma(t)$ on label accuracy. We calculate label accuracy by running the simulation for $10,000$ rounds so that the t-test to distinguish the slight differences between different risk level functions. The simulation results show that decreasing c from ∞ to 0 can slightly improve the label accuracy. Thus, we use $\gamma(t) = 0$ in our following experiments.

In Figure 3b, we study the effects of the uncertainty parameters a and b by running the simulation for $2,000$ rounds. From the figure, we can see that overly large a and b values must be avoided. They will lead to an overly long exploration stage, leading to the slow growth of label accuracy. On the other hand, overly small a and b will also slightly slow down the growth of label accuracy because of the higher risk of learning wrong worker models. From Figure 3b to 3c and 3d, we increase the numbers of tasks and workers by 2 times, respectively. The simulation results show that the major threat to label accuracy is always the overly large b which will greatly lengthen the exploration stage. From Figure 3c, we can also observe that, except for overly large b, the effects of other value changes in a and b are very tiny. Thus, we use the same parameter settings as the first line of Figure 3c in all the comparison experiments in Sections 4.1 and 4.2.

5 Conclusion and Future Work

In this paper, we propose a novel active tasks assignment strategy for crowed labeling based on the widely-adopted active learning strategy, expected error reduction. To boost the computation efficiency, we develop an approximation algorithm of variational inference to efficiently evaluate the benefits of each possible assignment. To improve the label accuracy, we use the posterior uncertainty to modulate the scope of task assignment and replace the expectation with the conditional value-at-risk for an optimistic evaluation. The experiments on two popular worker models and four MTurk datasets show that our strategy not only requires the least labels for high accuracy but also has better efficiency than existing prediction-based task assignment approaches.

Note that there are many other studies on task assignment in crowd labeling. For example, Lin *et al.* [2016], Kamar *et al.* [2012] and Fan *et al.* [2015] focus on how to improve task assignment by incorporating the classifiers of tasks. In fact, our strategy can be beneficial for them because the benchmarking strategies in this paper are widely-adopted in their studies. It will be one of our future studies to incorporate the contextual information of tasks in our strategy. Ho *et al.* [2013], Parameswaran *et al.* [2014], and Tarable *et al.* [2017] learn worker models using golden tasks of which the true labels are known in advance. The golden tasks can also be used in our strategy to improve the learning of worker models. Besides, we also wish to extend our strategy to the pull market of crowd labeling where tasks can be assigned to a worker only when he requires. In this market, the optimistic accuracy increment $\widetilde{G}$ can be used to not only decide the task assignment but also adjust the waiting time of workers.

Acknowledgments

This work was conducted within the Rolls-Royce@NTU Corporate Lab with support from the National Research Foundation (NRF) Singapore under Corp Lab@University Scheme.

References

[Buckley *et al.*, 2010] Chris Buckley, Matthew Lease, and Mark D. Smucker. Overview of the trec 2010 relevance feedback track. In *TREC Notebook*, 2010.

[Burden and Faires, 2004] R. Burden and J. Faires. *Numerical Analysis*. Cengage Learning, 2004.

[Chen *et al.*, 2015] 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.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. of CVPR*, 2009.

[Fan *et al.*, 2015] Ju Fan, Guoliang Li, Beng Chin Ooi, Kian-lee Tan, and Jianhua Feng. icrowd: An adaptive crowdsourcing framework. In *Proc. of SIGMOD*, 2015.

[Ho *et al.*, 2013] Chien-Ju Ho, Shahin Jabbari, and Jennifer Wortman Vaughan. Adaptive task assignment for crowdsourced classification. In *Proc. of ICML*, 2013.

[Hu and Zhang, 2017] Zehong Hu and Jie Zhang. Optimal posted-price mechanism in microtask crowdsourcing. In *Proc. of IJCAI*, 2017.

[Hua *et al.*, 2013] Gang Hua, Chengjiang Long, Ming Yang, and Yan Gao. Collaborative active learning of a kernel machine ensemble for recognition. In *Proc. of ICCV*, 2013.

[Ipeirotis *et al.*, 2010] Panagiotis G Ipeirotis, Foster Provost, and Jing Wang. Quality management on amazon mechanical turk. In *Proc. of the ACM SIGKDD workshop on human computation*, 2010.

[Kamar *et al.*, 2012] Ece Kamar, Severin Hacker, and Eric Horvitz. Combining human and machine intelligence in large-scale crowdsourcing. In *Proc. of AAMAS*, pages 467–474, 2012.

[Lewis and Catlett, 1994] David D Lewis and Jason Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Proc. of ICML*, 1994.

[Lin *et al.*, 2016] Christopher H Lin, M Mausam, and Daniel S Weld. Re-active learning: Active learning with relabeling. In *Proc. of AAAI*, 2016.

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

[Mező, 2013] István Mező. Some infinite sums arising from the weierstrass product theorem. *Applied Mathematics and Computation*, 219(18):9838–9846, 2013.

[Muhammadi *et al.*, 2015] Jafar Muhammadi, Hamid R Rabiee, and Abbas Hosseini. A unified statistical framework for crowd labeling. *Knowledge and Information Systems*, 45(2):271–294, 2015.

[Pang and Lee, 2005] Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proc. of ACL*, 2005.

[Parameswaran *et al.*, 2014] Aditya Parameswaran, Stephen Boyd, Hector Garcia-Molina, Ashish Gupta, Neoklis Polyzotis, and Jennifer Widom. Optimal crowd-powered rating and filtering algorithms. *Proceedings of the VLDB Endowment*, 7(9):685–696, 2014.

[Raykar and Agrawal, 2014] Vikas Raykar and Priyanka Agrawal. Sequential crowdsourced labeling as an epsilon-greedy exploration in a markov decision process. In *Proc. of AISTATS*, 2014.

[Rockafellar and Uryasev, 2002] R Tyrrell Rockafellar and Stanislav Uryasev. Conditional value-at-risk for general loss distributions. *Journal of banking & finance*, 26(7):1443–1471, 2002.

[Rodway *et al.*, 2012] Paul Rodway, Astrid Schepman, and Jordana Lambert. Preferring the one in the middle: Further evidence for the centre-stage effect. *Applied Cognitive Psychology*, 26(2):215–222, 2012.

[Settles, 2012] Burr Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114, 2012.

[Sheng *et al.*, 2008] Victor S Sheng, Foster Provost, and Panagiotis G Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proc. of SIGKDD*, 2008.

[Sheshadri and Lease, 2013] Aashish Sheshadri and Matthew Lease. SQUARE: A Benchmark for Research on Computing Crowd Consensus. In *Proc. of HCOMP*, 2013.

[Simpson and Roberts, 2015] Edwin Simpson and Stephen Roberts. Bayesian methods for intelligent task assignment in crowdsourcing systems. In *Decision Making: Uncertainty, Imperfection, Deliberation and Scalability*, pages 1–32. Springer, 2015.

[Slivkins and Vaughan, 2014] Aleksandrs Slivkins and Jennifer Wortman Vaughan. Online decision making in crowdsourcing markets: Theoretical challenges. *ACM SIGecom Exchanges*, 12(2):4–23, 2014.

[Snow *et al.*, 2008] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proc. of EMNLP*, 2008.

[Tarable *et al.*, 2017] Alberto Tarable, Alessandro Nordio, Emilio Leonardi, and Marco Ajmone Marsan. The importance of worker reputation information in microtask-based crowd work systems. *IEEE Transactions on Parallel and Distributed Systems*, 28(2):558–571, 2017.

[Welinder and Perona, 2010] Peter Welinder and Pietro Perona. Online crowdsourcing: rating annotators and obtaining cost-effective labels. In *CVPR Workshops*, 2010.

[Zheng *et al.*, 2015] Yudian Zheng, Jiannan Wang, Guoliang Li, Reynold Cheng, and Jianhua Feng. Qasca: A quality-aware task assignment system for crowdsourcing applications. In *Proc. of SIGMOD*, 2015.

[Zheng *et al.*, 2017] 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.