Obtaining High-Quality Label by Distinguishing between Easy and Hard Items in Crowdsourcing

Wei Wang, Xiang-Yu Guo, Shao-Yuan Li, Yuan Jiang, and Zhi-Hua Zhou
National Key Laboratory for Novel Software Technology
Nanjing University, Nanjing 210023, China
{wangw,guoxy,lisy,jiangy,zhouzh}@lamda.nju.edu.cn

Abstract

Crowdsourcing systems make it possible to hire voluntary workers to label large-scale data by offering them small monetary payments. Usually, the taskmaster requires to collect high-quality labels, while the quality of labels obtained from the crowd may not satisfy this requirement. In this paper, we study the problem of obtaining high-quality labels from the crowd and present an approach of learning the difficulty of items in crowdsourcing, in which we construct a small training set of items with estimated difficulty and then learn a model to predict the difficulty of future items. With the predicted difficulty, we can distinguish between *easy* and *hard* items to obtain high-quality labels. For *easy* items, the quality of their labels inferred from the crowd could be high enough to satisfy the requirement; while for *hard* items, the crowd could not provide high-quality labels, it is better to choose a more knowledgable crowd or employ specialized workers to label them. The experimental results demonstrate that the proposed approach by learning to distinguish between *easy* and *hard* items can significantly improve the label quality.

1 Introduction

In recent years, unlabeled data can often be obtained abundantly and cheaply, e.g., one can write a crawler in a few lines of code and automatically download hundreds of thousands of images from the internet. Generally, these unlabeled data cannot be directly used in machine learning, since learning algorithms need data with labels to train a model for making predictions, e.g., the complex deep neural network requires large amounts of labeled data in the training process. Providing labels for large amounts of unlabeled data has always been a challenge because labeling the data is expensive and time-consuming. An effective and efficient paradigm for label collection is crowdsourcing [Howe, 2006; 2008], e.g., in the famous crowdsourcing system Amazon Mechanical Turk (AMT), the taskmaster submits the task that can be completed by voluntary workers in exchange for small monetary payments. This task is usually called Human Intelligence Task (HIT), which computers are currently unable to perform while humans can complete easily, e.g., annotating images of trees versus images of non-trees.

Crowdsourcing makes it possible for many independent and relatively inexpensive workers to provide labels for large-scale unlabeled data together. These workers usually come from a large society and each of them is presented with multiple items of the crowdsourcing task. Each worker has to answer the question about the item presented to her/him and provides a label based on her/his own knowledge. Crowdsourcing can collect large-scale labels with small monetary payments in a short period of time, unfortunately, these labels collected via crowdsourcing are typically highly erroneous [Kazai *et al.*, 2011; Vuurens *et al.*, 2011] due to the fact that most workers in the crowd are non-experts. In machine learning, label quality is crucial to the performance of learning algorithm and higher-quality labels will bring better predictions. How to obtain high-quality labels via crowdsourcing is an important issue to address.

For improving the label quality, the common wisdom is to add redundancy into the labels, i.e., each item is presented with multiple workers. In this way, multiple labels are collected for each item and the final solution is determined by aggregating these crowded labels [Sheng *et al.*, 2008; Snow *et al.*, 2008; Sorokin and Forsyth, 2008]. In the past few years, researchers have developed many methods for inferring good labels from the crowd, most of which build probabilistic models for the crowdsourcing process and derive the labels using algorithms based on Expectation Maximization (EM) and other inference tools. Raykar et al. [2009; 2010] used a two-coin model to measure the sensitivity and specificity of each worker and iteratively estimated the two terms by an EM algorithm. Whitehill et al. [2009; 2010] formulated a probabilistic model of worker quality and applied an EM algorithm to infer the most probable label. Raykar and Yu [2012] developed an empirical Bayesian algorithm based on EM to iteratively estimate the ground-truth label and eliminate the spammers. Liu et al. [2012] transformed the crowdsourcing problem into a variational inference problem in graphical models and inferred the labels with variational inference tools including Belief Propagation and Mean Field. Tian and Zhu [2015] inferred the ground-truth label based on the max-margin principle by maximizing the margin between the aggregated score of potential true label and other alternatives. The minimax entropy principle has al-

so been introduced into crowdsourcing. Zhou et al. [2012; 2014] proposed a minimax entropy method to infer the ground-truth label by minimizing the entropy of the probabilistic distribution. Wauthier and Jordan [2011] proposed a Bayesian framework named the Bayesian Bias Mitigation to unify the process of inferring the labels and learning from the inferred labels. Wang and Zhou [2016] presented a PAC bound on the number of labels collected via crowdsourcing for learning a good model.

Some other works focused on how to choose which items are assigned to each worker. Yan et al. [2011] employed a probabilistic model to select the worker from which to query the label. Karger et al. [2011; 2014] proposed an item assignment algorithm based on a random regular bipartite graph. Ho and Vaughan [2012] developed an algorithm for assigning heterogeneous items to workers with different qualities based on the online primal-dual technique [Buchbinder and Naor, 2005]. Later, Ho and Vaughan [2013] utilized items with true labels to estimate workers' performance and proposed a provably near-optimal assignment algorithm for heterogeneous items. Liu et al. [2013] also exploited items with known answers to evaluate workers' performance for item assignment. Chen et al. [2013; 2015] formulated a finite horizon Markov Decision Process in a Bayesian setting and characterized the optimality using dynamic programming. Raykar and Agrawal [2014] modeled the item assignment problem as a Markov decision process and defined an Bayesian decision theoretic utility function to jointly consider the cost for acquiring additional labels and the possible accuracy improvement. There are also some interesting works on developing payment mechanisms to encourage workers to provide high-quality labels for items [Shah *et al.*, 2015; Shah and Zhou, 2015; 2016]. In this situation, the task is posted together with the labeling guidelines and payment instructions, and it is demanded that the worker label each item according to her/his own belief of the answer being correct. They also assumed that the threshold of this belief is known to the payment mechanism designer and that the worker's payment is based on her/his performance on the gold standard items (a set of items whose true labels are known). However, it is difficult to get this threshold in real-world applications and the gold standard items will increase the labeling cost.

1.1 Our Focus and Contribution

In crowdsourcing, thousands of workers have internet access to the posted task. These voluntary workers have different abilities, since they may come from different regions, receive different educations and have different knowledge. Usually, the items of the crowdsourcing task also have different difficulties for different workers. For *easy* items, most workers could provide correct labels with high probability, aggregating several labels from the crowd could infer the high-quality labels; while for *hard* items which need specific domain knowledge for correct labeling, only a portion of the crowd could provide correct labels with high probability, aggregating several labels from the crowd without knowing who these knowledgable workers are may not infer the high-quality labels. An optimistic idea is to identify these knowledgable workers and let them provide labels to *hard* items. However, it is difficult to identify the knowledgable workers without prior knowledge about the crowd. Furthermore, in many crowdsourcing systems, the voluntary workers are anonymous and transient, the taskmaster cannot assign particular items to an identified worker, nor could she/he expect some worker to show up again in the future.

This motivates us to propose a novel crowdsourcing approach, in which we learn to distinguish between *easy* and *hard* items for the crowd. For *easy* items, it is feasible to derive high-quality labels from the current crowd; while for *hard* items, it is difficult to derive high-quality labels from the current crowd without prior knowledge. Choosing a more knowledgable crowd or employing specialized workers for these *hard* items will be a good strategy for obtaining high-quality labels. Following this direction, we propose a two-stage efficient algorithm: in the first stage, we let the crowd label a small portion of items to estimate their difficulty; then in the second stage we train a model with this training set to predict the difficulty of future items. Similar items should have similar labels and the worker would provide similar labels to similar items. Thus, similar items should have similar labeling difficulty with respect to the same crowd. This could explain why the process of learning the difficulty is reasonable. The experimental results demonstrate that learning to distinguish between *easy* and *hard* items with the proposed algorithm can significantly improve the label quality.

The rest of this paper is organized as follows. After introducing some preliminaries in Section 2, we present our approach in Section 3 and discuss how to choose the parameter in Section 4. Finally, we conduct experiments in Section 5 and make a conclusion in Section 6.

2 Preliminaries

In this paper, we consider the task involving binary-choice items. The task has a set of m items $\{x_1, \ldots, x_m\}$ over $\mathcal{X}$, each item corresponds to a binary-choice example x_i with an unobserved true label $y_i \in \{0, 1\}$, $1 \leq i \leq m$ (e.g., annotating whether an image contains trees or not). These items will be labeled by a crowd, where $\mathcal{W}$ denotes the set of all workers in the crowd. To improve label quality and reliability, an example x_i is generally presented to N workers denoted by $\{w_1, \ldots, w_N\}$, $w_j \in \mathcal{W}$, $1 \leq j \leq N$. For any item $x_i \in \mathcal{X}$, a worker w_j provides a label $y_i^j = w_j(x_i) \in \{0, 1\}$ on x_i and a final label is then inferred for x_i based on the multiple labels.

The item may have different difficulties for different workers and the difficulty depends on many factors, e.g., the time it takes for labeling, the knowledge required for correct labeling, the worker's honesty, the payment mechanism, and the item itself. In fact, the difficulty of an item can be evaluated from the output of the worker, i.e., if worker w_j provides a correct label to item x_i with great probability, it is regarded that x_i is an *easy* item for w_j; if w_j provides an incorrect label to x_i with great probability, it is regarded that x_i is a *hard* item for w_j. The posterior probability $d_{x_i}(w_j) = P(y_i^j \neq y_i | x_i)$ denotes the probability that w_j provides an incorrect label for x_i and can be thought of as the difficulty of x_i for w_j. For any item x, we define the following difficulty of x for the crowd

$\mathcal{W}$:

Definition 1 ***(Difficulty)*** *Let $\mathbb{D}_{\mathcal{W}}$ denote the underlying distribution over the workers in the crowd $\mathcal{W}$, the difficulty of the item (x, y) with respect to the crowd $\mathcal{W}$ is defined as*

$$d_x(\mathcal{W}) = \int_{w \in \mathbb{D}_{\mathcal{W}}} P\big(w(x) \neq y|x\big)p(w)dw.$$

For the sake of convenience, we will write $d_x(\mathcal{W})$ as d_x for the fixed crowd $\mathcal{W}$. In particular, one frequently provides nearly random labels to the items that are too difficult to answer [Shah and Zhou, 2015], and the empirical observations also show that the workers in crowdsourcing systems, as opposed to being adversarial in nature, at worst provide random labels to items [Yuen *et al.*, 2011; Gadiraju *et al.*, 2015]. Although the crowd may contain few adversaries who give wrong answers deliberately, it is reasonable to assume that $d_x \leq 1/2$, i.e., the crowd is not dominated by the adversaries. When d_x is small, x is an *easy* item; while d_x is large, x is an *hard* item. For *hard* items, there may exist some knowledgable workers w_j with small $d_x(w_j)$. However, it is not feasible to identify these knowledgable workers for *hard* items without prior knowledge about the crowd.

3 Our Method

When no prior knowledge about the crowd is known, randomly selecting N workers from the crowd and using the majority voting to infer the final label $\widehat{y}$ for x is a good error-pruning strategy, i.e.,

$$\widehat{y} = \begin{cases} 1 & \text{if } \frac{1}{N}\sum_{j=1}^{N} y^j > \frac{1}{2} \\ \text{random guess} & \text{if } \frac{1}{N}\sum_{j=1}^{N} y^j = \frac{1}{2} \\ 0 & \text{if } \frac{1}{N}\sum_{j=1}^{N} y^j < \frac{1}{2} \end{cases}$$

For the item x, we can give the upper bound on the label quality for majority voting with respect to the difficulty of the item in the following proposition.

Proposition 1 *For the item (x, y) and the crowd $\mathcal{W}$, let $\widehat{y}$ denote the inferred label of x with majority voting from N workers, the following inequality holds.*

$$P(\widehat{y} \neq y) \leq \exp\Big(-2N(1/2 - d_x)^2\Big). \tag{1}$$

Proof. Proposition 1 can be proved with Hoeffding [1963] bound. □

Proposition 1 states that small d_x will bring high-quality label while large d_x will lead to low-quality label. If the taskmaster requires that a minimum quality of the labels obtained from the crowd should be achieved, the maximum difficulty d_x can be derived according to Equation 1. This maximum difficulty d_x can be thought of as a threshold η, the *easy* items whose d_x is no larger than η could be labeled by the crowd and the label quality of them could satisfy the requirement; while the *hard* items whose d_x is larger than η should not be labeled by the current crowd since the crowd may provide low-quality labels for them with great risk.

Now an important issue arises: how to estimate the difficulty efficiently? A straight-forward way is that we let the crowd label all items and then estimate d_x with these crowded labels. The item $x \in \mathcal{X}$ is labeled by N workers $\{w_1, \ldots, w_N\}$, we give the definition of empirical difficulty $\widehat{d}_x$ of x:

$$\widehat{d}_x = \frac{1}{N}\sum_{j=1}^{N} \mathbb{I}(y^j \neq \widehat{y}). \tag{2}$$

With this empirical difficulty $\widehat{d}_x$, we give the following proposition:

Proposition 2 *For the item (x, y) and any $\epsilon \in (0, 1)$, the following inequality on $\widehat{d}_x$ and d_x holds.*

$$P\Big(\big|\widehat{d}_x - d_x\big| \geq \epsilon\Big) \leq 2\exp\big(-2N\epsilon^2\big). \tag{3}$$

Proof. For the item (x, y), if $\widehat{y} = y$, it is easy to prove that $P(|d_x - \widehat{d}_x| \geq \epsilon) \leq 2\exp(-2N\epsilon^2)$ with Hoeffding [1963] bound; if $\widehat{y} \neq y$, it is easy to prove that $P(|(1 - d_x) - \widehat{d}_x| \geq \epsilon) \leq 2\exp(-2N\epsilon^2)$ with Hoeffding [1963] bound. So we get that $P\big(\big||1/2 - d_x| - |1/2 - \widehat{d}_x|\big| \geq \epsilon\big) \leq 2\exp(-2N\epsilon^2)$. Since the label $\widehat{y}$ is inferred by majority voting, we get that $\widehat{d}_x \leq 1/2$ with Equation 2. Considering that $d_x \leq 1/2$, we have that $P\big(\big|\widehat{d}_x - d_x\big| \geq \epsilon\big) \leq 2\exp(-2N\epsilon^2)$. □

Proposition 2 states that $\widehat{d}_x$ is a good approximation of d_x when we can not reach the true label of the item. After the crowd has labeled all items, we get the estimation $\widehat{d}_x$ of the true difficulty d_x for each item. If $\widehat{d}_x \leq \eta$, it implies that x is an *easy* item; if $\widehat{d}_x > \eta$, it implies that x is a *hard* item. For the requirement of high label quality, we should not accept the inferred labels for *hard* items. In this situation, the crowd has provided labels to these *hard* items, but we could not accept their labels in order to avoid the risk. This wastes too much labeling cost and motivates us to develop an approach for estimating the difficulty before labeling all items.

In order to address this problem we propose a two-stage algorithm. In the first stage, we select a small portion of items $\mathcal{T} \subset \mathcal{X}$ randomly from the whole set $\mathcal{X}$ and let the crowd provide labels to them. With these crowded labels we estimate the empirical difficulty $\widehat{d}_x$ of the item $x \in \mathcal{T}$ and construct a training set $\mathcal{L}$ with the estimated difficulty, i.e., $\mathcal{L} = \big\{(x_1, \widehat{d}_{x_1}), \ldots, (x_{|\mathcal{T}|}, \widehat{d}_{x_{|\mathcal{T}|}})\big\}$. Then in the second stage we learn a model with $\mathcal{L}$ to predict the difficulty of the item in $\mathcal{X} - \mathcal{T}$. With the predicted difficulty, we set a threshold η to distinguish between *easy* and *hard* items (please see Section 4 for how to choose η).

Let $\Phi : \mathcal{X} \to \mathbb{R}^M$ denote the kernel mapping from the input space $\mathcal{X}$ to $\mathbb{R}^M$, considering the linear hypothesis space

$$\mathcal{H} = \{x \to h \cdot \Phi(x) + b : h \in \mathbb{R}^M, b \in \mathbb{R}\},$$

we assume that the difficulty is determined by the function that

$$f(x) = \frac{1}{1 + \exp\Big(-\big(h \cdot \Phi(x) + b\big)\Big)}$$

with h^* and b^*, i.e., for any item x,

$$d_x = \frac{1}{1+\exp\Big(-\big(h^*\cdot\Phi(x)+b^*\big)\Big)}. \quad (4)$$

With the training set $\mathcal{L}$, we try to find the optimal (h^*, b^*) with Kernel Ridge Regression by minimizing the following loss function:

$$\min_{h,b} Loss(h,b) = \sum_{i=1}^{m}\big(f(x_i)-\widehat{d}_{x_i}\big)^2 + \alpha\|h\|^2. \quad (5)$$

For this learning problem, our aim is to predict whether the item x should be labeled by the crowd to achieve high-quality label with small risk, i.e., $\widehat{d}_x \le \eta$ or not. For a good model $f(\cdot)$, the *easy* items in the training set $\mathcal{L}$ should also be predicted as the *easy* ones by $f(\cdot)$, i.e., if $\widehat{d}_{x_i} \le \eta$, $f(x_i) \le \eta$; otherwise, $f(x_i) > \eta$. So we introduce the constraints into the optimization and Equation 5 is transformed into:

$$\min_{h,b} Loss(h,b) = \sum_{i=1}^{m}\big(f(x_i)-\widehat{d}_{x_i}\big)^2 + \alpha\|h\|^2,$$
$$s.t., (\widehat{d}_{x_i}-\eta)\big(f(x_i)-\eta\big) \ge 0. \quad (6)$$

After introducing the parameter β, the optimization problem with constraints can be formalized as

$$\min_{h,b} Loss(h,b) = \sum_{i=1}^{m}\big(f(x_i)-\widehat{d}_{x_i}\big)^2 + \alpha\|h\|^2$$
$$+\beta\sum_{i=1}^{m}\mathbb{I}\big((\widehat{d}_{x_i}-\eta)(f(x_i)-\eta)\le 0\big). \quad (7)$$

The optimization problem in Equation 7 can not be solved efficiently, since the last term $\mathbb{I}(t \le 0)$ is not differentiable. We replace it with its surrogate loss function $\log_2(1+e^{-t})$ and get the optimization problem shown as

$$\min_{h,b} Loss(h,b) = \sum_{i=1}^{m}\big(f(x_i)-\widehat{d}_{x_i}\big)^2 + \alpha\|h\|^2$$
$$+\beta\sum_{i=1}^{m}\log_2\Big(1+\exp\big(-(\widehat{d}_{x_i}-\eta)(f(x_i)-\eta)\big)\Big). \quad (8)$$

Thus, by solving the optimization problem in Equation 8 with the training set $\mathcal{L}$, we can find the optimal solution (h^*, b^*) to predict the difficulty of the item in $\mathcal{X}-\mathcal{T}$. The process is described in Algorithm 1.

With the predicted difficulty $d_x = \frac{1}{1+\exp(-(h^*\cdot\Phi(x)+b^*))}$ and the threshold η, we can determine whether an item should be labeled by the crowd, i.e., if $d_x \le \eta$, x should be labeled by the crowd; otherwise, x should not be labeled by the crowd due to high risk. In the following section, we will discuss how to choose the parameter η.

4 The Parameter η

η is a parameter in the process, smaller η will bring less *easy* items and higher label quality, while larger η will lead to more *easy* items but lower label quality. η depends on the required label quality or the ratio of items that need to be labeled by the crowd, we will discuss how to choose it in this section.

Algorithm 1 Difficulty Learning

Input:
Unlabeled items $\mathcal{X}$, the crowd $\mathcal{W}$, the threshold η, the parameter γ and the kernel Φ.

Process:
1. Select number of $\gamma|\mathcal{X}|$ items randomly from $\mathcal{X}$ as $\mathcal{T}$;
2. Each item $x_i \in \mathcal{T}$ is presented to N workers from the crowd $\mathcal{W}$ and its label $\widehat{y}_i$ is inferred by majority voting;
3. Calculate the estimated difficulty $\widehat{d}_{x_i}$ for each item $x_i \in \mathcal{T}$ according to Equation 2 and get the training set $\mathcal{L} = \big\{(x_1, \widehat{d}_{x_1}), \ldots, (x_{|\mathcal{T}|}, \widehat{d}_{x_{|\mathcal{T}|}})\big\}$;
4. Solve the optimization problem to get (h^*, b^*) with the training set $\mathcal{L}$ according to Equation 8.

Output:
The difficulty $d_x = \frac{1}{1+\exp(-(h^*\cdot\Phi(x)+b^*))}$ for each item $x \in \mathcal{X}-\mathcal{T}$.

- **Label Quality.** *Choose η with respect to the required label quality.*

  Proposition 1 indicates that the upper bound on the error rate of labels depends on the difficulty of the item. We can derive the maximum difficulty $d_{x\max}$ according to the required label accuracy and set the threshold $\eta = d_{x\max}$.

- **Item Ratio.** *Choose η with respect to the required item ratio.*

  In some applications, the volume of unlabeled items is huge. The taskmaster must set a minimum ratio R of these unlabeled items that need to be labeled by the crowd to save labeling cost, since employing the crowd costs less than employing specialized workers. Thus, R is the ratio of *easy* items whose difficulty is no larger than η. With the training set $\mathcal{L} = \{(x_1, \widehat{d}_{x_1}), \ldots, (x_{|\mathcal{T}|}, \widehat{d}_{x_{|\mathcal{T}|}})\}$, we set η according to the following Equation 9, i.e., we use the ratio of *easy* items in the training set $\mathcal{L}$ as an estimation of R.

$$\eta = \min\Big\{\eta\in[0,1] : \frac{\big|\{x\in\mathcal{L} : d_x \le \eta\}\big|}{|\mathcal{L}|} \ge R\Big\}. \quad (9)$$

- **Label Information.**

  In other applications, the taskmaster may require that both the ratio of items that need to be labeled by the crowd and the label accuracy on these labeled items should be as high as possible. Let A denote the label accuracy on the *easy* items that need to be labeled by the crowd, A can be denoted as

$$A = \frac{\big|\{x\in\mathcal{X}-\mathcal{T} : \widehat{y}=y \wedge d_x \le \eta\}\big|}{\big|\{x\in\mathcal{X}-\mathcal{T} : d_x \le \eta\}\big|}.$$

  The ratio of *easy* items can be denoted as

$$R = \frac{\big|\{x\in\mathcal{X}-\mathcal{T} : d_x \le \eta\}\big|}{\big|\{x\in\mathcal{X}-\mathcal{T}\big|}.$$

We define *Label Information* I in Equation 10, i.e., the probability that the item in $\mathcal{X}-\mathcal{T}$ can be labeled correctly by the crowd.

$$I = A \cdot R = \frac{\left|\{x \in \mathcal{X}-\mathcal{T} : \widehat{y} = y \wedge d_x \leq \eta\}\right|}{\left|\{x \in \mathcal{X}-\mathcal{T}\right|}. \quad (10)$$

If the taskmaster requires that both the ratio of labeled items and the label accuracy on these labeled items are as high as possible, it implies that the threshold η should be the one that maximizes the label information I.

In real-world crowdsourcing, most items can be labeled correctly by the crowd, the number of *hard* items is much less than that of *easy* ones and the probability distribution of item's difficulty can be estimated from the training set $\mathcal{L}$. In the following part of this section, we give an example for choosing the parameter η. From the training set $\mathcal{L}$ constructed in Section 5, we get the cumulative distribution function of item's difficulty, which is depicted as the black dashed line in Figure 1. Considering the truncated Beta distribution whose probability density function is shown in Equation 11 with parameters s and t, Z is the normalization term such that $\int_0^{\frac{1}{2}} \frac{1}{Z}\frac{x^{s-1}(1-x)^{t-1}}{B(s,t)} = 1$, we get its cumulative distribution function with $s = 0.231$ and $t = 0.774$, which is depicted as the blue solid line in Figure 1.

$$\frac{1}{Z}\frac{x^{s-1}(1-x)^{t-1}}{B(s,t)}, \text{ where } B(s,t) = \frac{\Gamma(s)\Gamma(t)}{\Gamma(s+t)} \quad (11)$$

It can be found that the cumulative distribution function of item's difficulty in the training set $\mathcal{L}$ is very close to the cumulative distribution function of the truncated Beta distribution with $s = 0.231$ and $t = 0.774$. So we assume that the probability distribution of item's difficulty follows the truncated Beta distribution to facilitate the calculation. For *easy* items, with Proposition 1 we get that $A \geq 1 - \exp\left(-2N(1/2-\eta)^2\right)$ since $d_x \leq \eta$; with the truncated Beta distribution, we get that the ratio of *easy* items is

$$R = \int_0^{\eta} \frac{1}{Z}\frac{x^{s-1}(1-x)^{t-1}}{B(s,t)}.$$

So we get that

$$\begin{aligned} I &= A \cdot R \\ &\geq \frac{1}{Z}\left(1 - \exp(-2N(1/2-\eta)^2)\right) \int_0^{\eta} \frac{x^{s-1}(1-x)^{t-1}}{B(s,t)}. \end{aligned}$$

Thus, the optimal η^* can be denoted as

$$\begin{aligned} \eta^* = \arg\max &\left(1 - \exp(-2N(1/2-\eta)^2)\right) \\ &\cdot \int_0^{\eta} \frac{x^{s-1}(1-x)^{t-1}}{B(s,t)}. \end{aligned}$$

With the parameters $s = 0.231$, $t = 0.774$ and $N = 9$, the optimal $\eta^* \approx 0.16$.

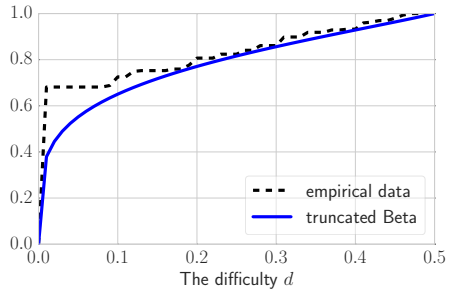


Figure 1: Cumulative distribution function.

5 Experiments

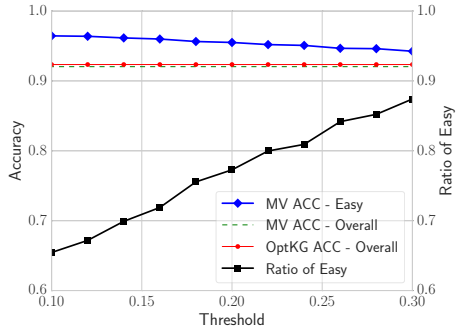
In order to study whether we can obtain high-quality labels by learning to distinguish between *easy* and *hard* items we conduct experiments in this section.

We perform experiments on the real data. In the task, there are 1499 different images and the workers are required to label whether tree appears in the image or not. The crowd that labels the data consists of 38 graduate students. Each image is labeled by several workers selected randomly from the crowd. Due to various reasons, some workers did not return their labels for some images. Finally, we drop those images labeled by less than 5 workers and get 1495 images. The number of labels provided by workers for each image ranges from 5 to 16. We use fisher vector of each image as its feature, which is a 1248-dimension vector.

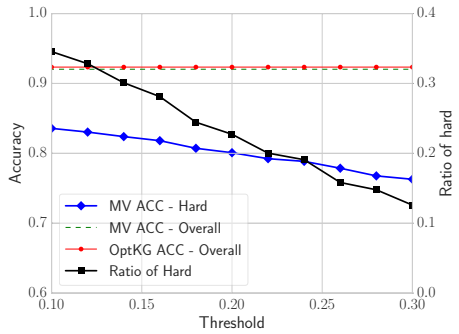
We randomly select 20% images to generate the training set $\mathcal{L}$ and the difficulty of these images are estimated according to Equation 2. The optimization problem in Equation 8 is solved to predict the difficulty of the images in $\mathcal{X}-\mathcal{L}$, where the kernel function Φ is set to be the linear kernel, $\alpha = 10^{-4}$ and $\beta = 3$. The parameter η is set between 0.1 and 0.3. Each experiment is repeated for 50 runs and the average performance is depicted in Figure 2.

Figure 2(a) presents the ratio of images that are predicted as the *easy* ones and the accuracy of the labels inferred from the crowd on *easy* items with respect to different choices of η. The line *Ratio of Easy* denotes the ratio of images that are predicted as the *easy* ones; the line *MV ACC-Easy* denotes the accuracy of labels on *easy* items; the line *MV ACC-Overall* denotes the accuracy of labels inferred by majority voting when all items are labeled by the crowd; the line *OptKG ACC-Overall* denotes the accuracy of labels inferred by the state-of-art crowdsourcing method OptKG [Chen *et al.*, 2013; 2015] when all items are labeled by the crowd. From Figure 2(a) we find that the label accuracy of *easy* items is much higher than that of *MV ACC-Overall* and that of *OptKG ACC-Overall*. For example, when $\eta = 0.2$ the ratio of *easy* items is 0.773 and the label accuracy of *easy* items is 95.5%, while the accuracy of *MV ACC-Overall* is 92.0% and the accuracy of *OptKG ACC-Overall* is 92.3%.

We also present the results on *hard* items in Figure 2(b), which shows the ratio of items that are predicted as the *hard* ones and the accuracy of the labels inferred on these *hard* items with respect to different choices of η. From Figure



(a) Accuracy and ratio w.r.t. *easy* items.



(b) Accuracy and ratio w.r.t. *hard* items.

Figure 2: Results on the data.

2(b) we find that the label accuracy of *hard* items is much lower than that of *MV ACC-Overall* and that of *OptKG ACC-Overall*. For example, when $\eta = 0.2$, the ratio of *hard* items is 0.227 and the label accuracy of these *hard* items is only 80.1%. The current crowd could not provide high-quality labels to these *hard* items, we should find a more knowledgable crowd or employ specialized workers to label them.

Figure 2 validates that we could obtain high-quality labels by learning to distinguish between *easy* and *hard* items. When the taskmaster requires high label quality, we should choose a small η; when the taskmaster requires both high label quality and large ratio of items that need to be labeled by the crowd, we should choose a medium η.

In order to show whether the size of training set $\mathcal{L}$ influences the label accuracy, we run experiments with different sizes of $\mathcal{L}$, i.e., from 10% to 40% with an interval 5%, and the results for a fixed $\eta = 0.2$ are depicted in Figure 3. From Figure 3 we can find that the label accuracy on *easy* items varies from 95.0% to 95.9% increasingly with the size of $\mathcal{L}$ varying from 10% to 40%. It indicates that the label quality is not very sensitive to the size of $\mathcal{L}$.

6 Conclusion

In this paper, we propose an approach for obtaining high-quality labels from the crowd by learning to distinguish between *easy* and *hard* items before they are sent out to workers, which is complementary to the line of work that tries to obtain reliable labels via crowdsourcing for machine learning

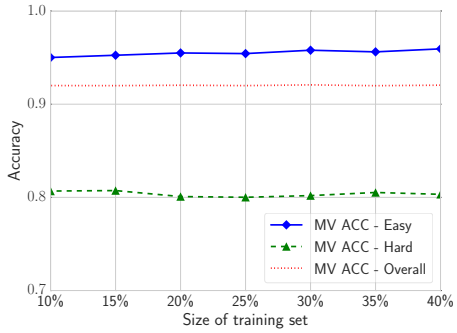


Figure 3: Accuracy with different sizes of $\mathcal{L}$ for $\eta = 0.2$.

or other applications. It allows for better post-processing of *hard* items, i.e., we should find a more knowledgable crowd or employ specialized workers to label them, and provides a possibility that the items could be completed by different crowds with respect to their difficulty. We regard this work as preliminary and expect that more researches will follow this direction.

Acknowledgments

This work was supported by the NSFC (61673202, 61673201), 973 Program (2014CB340501), and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

[Buchbinder and Naor, 2005] Niv Buchbinder and Joseph Naor. Online primal-dual algorithms for covering and packing problems. In *ESA*, pages 689–701, Palma de Mallorca, Spain, 2005.

[Chen *et al.*, 2013] Xi Chen, Qihang Lin, and Dengyong Zhou. Optimistic knowledge gradient policy for optimal budget allocation in crowdsourcing. In *ICML*, pages 64–72, Atlanta, GA, 2013.

[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.

[Gadiraju *et al.*, 2015] Ujwal Gadiraju, Ricardo Kawase, Stefan Dietze, and Gianluca Demartini. Understanding malicious behavior in crowdsourcing platforms: The case of online surveys. In *ACM CHI*, pages 1631–1640, Seoul, Republic of Korea, 2015.

[Ho and Vaughan, 2012] Chien-Ju Ho and Jennifer W. Vaughan. Online task assignment in crowdsourcing markets. In *AAAI*, Toronto, Canada, 2012.

[Ho *et al.*, 2013] Chien-Ju Ho, Shahin Jabbari, and Jennifer W. Vaughan. Adaptive task assignment for crowdsourced classification. In *ICML*, pages 534–542, Atlanta, GA, 2013.

[Hoeffding, 1963] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13–30, 1963.

[Howe, 2006] Jeff Howe. The rise of crowdsourcing. *Wired*, 2006.

[Howe, 2008] Jeff Howe. Why the power of the crowd is driving the future of business. *Crown Business*, 2008.

[Karger *et al.*, 2011] David R. Karger, Sewoong Oh, and Devavrat Shah. Iterative learning for reliable crowdsourcing systems. In *NIPS*, pages 1953–1961. MIT Press, Cambridge, MA, 2011.

[Karger *et al.*, 2014] David R. Karger, Sewoong Oh, and Devavrat Shah. Budget-optimal task allocation for reliable crowdsourcing systems. *Operations Research*, 62(1):1–24, 2014.

[Kazai *et al.*, 2011] Gabriella Kazai, Jaap Kamps, Marijn Koolen, and Natasa Milic-Frayling. Crowdsourcing for book search evaluation: impact of hit design on comparative system ranking. In *ACM SIGIR*, pages 205–214, Beijing, China, 2011.

[Liu *et al.*, 2012] Qiang Liu, Jian Peng, and Alexander T. Ihler. Variational inference for crowdsourcing. In *NIPS*, pages 701–709. MIT Press, Cambridge, MA, 2012.

[Liu *et al.*, 2013] Qiang Liu, Alexander T. Ihler, and Mark Steyvers. Scoring workers in crowdsourcing: How many control questions are enough? In *NIPS*, pages 1914–1922. MIT Press, Cambridge, MA, 2013.

[Raykar and Agrawal, 2014] Vikas C. Raykar and Priyanka Agrawal. Sequential crowdsourced labeling as an epsilon-greedy exploration in a markov decision process. In *AISTATS*, pages 832–840, Reykjavik, Iceland, 2014.

[Raykar and Yu, 2012] Vikas C. Raykar and Shipeng Yu. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *Journal of Machine Learning Research*, 13:491–518, 2012.

[Raykar *et al.*, 2009] Vikas C. Raykar, Shipeng Yu, Linda H. Zhao, Anna K. Jerebko, Charles Florin, Gerardo Hermosillo Valadez, Luca Bogoni, and Linda Moy. Supervised learning from multiple experts: whom to trust when everyone lies a bit. In *ICML*, Quebec, Canada, 2009.

[Raykar *et al.*, 2010] Vikas C. Raykar, Shipeng Yu, Linda H. Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. Learning from crowds. *Journal of Machine Learning Research*, 11:1297–1322, 2010.

[Shah and Zhou, 2015] Nihar B. Shah and Dengyong Zhou. Double or nothing: Multiplicative incentive mechanisms for crowdsourcing. In *NIPS*, pages 1–9. MIT Press, Cambridge, MA, 2015.

[Shah and Zhou, 2016] Nihar B. Shah and Dengyong Zhou. No oops, you won't do it again: Mechanisms for self-correction in crowdsourcing. In *ICML*, pages 1–10, New York, NY, 2016.

[Shah *et al.*, 2015] Nihar B. Shah, Dengyong Zhou, and Yuval Peres. Approval voting and incentives in crowdsourcing. In *ICML*, pages 10–19, Lille, France, 2015.

[Sheng *et al.*, 2008] Victor S. Sheng, Foster J. Provost, and Panagiotis G. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *ACM SIGKDD*, pages 614–622, Las Vegas, NV, 2008.

[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 *EMNLP*, pages 254–263, Honolulu, HI, 2008.

[Sorokin and Forsyth, 2008] Alexander Sorokin and David A. Forsyth. Utility data annotation with amazon mechanical turk. In *CVPR Workshop on Internet Vision*, pages 1–8, Anchorage, AK, 2008.

[Tian and Zhu, 2015] Tian Tian and Jun Zhu. Max-margin majority voting for learning from crowds. In *NIPS*, pages 1612–1620. MIT Press, Cambridge, MA, 2015.

[Vuurens *et al.*, 2011] Jeroen Vuurens, Arjen P. de Vries, and Carsten Eickhoff. How much spam can you take? an analysis of crowdsourcing results to increase accuracy. In *ACM SIGIR Workshop on Crowdsourcing for Information Retrieval*, pages 21–26, Beijing, China, 2011.

[Wang and Zhou, 2016] Lu Wang and Zhi-Hua Zhou. Cost-saving effect of crowdsourcing learning. In *IJCAI*, pages 2111–2117, New York, NY, 2016.

[Wauthier and Jordan, 2011] Fabian L. Wauthier and Michael I. Jordan. Bayesian bias mitigation for crowdsourcing. In *NIPS*, pages 1800–1808. MIT Press, Cambridge, MA, 2011.

[Welinder *et al.*, 2010] Peter Welinder, Steve Branson, Serge J. Belongie, and Pietro Perona. The multidimensional wisdom of crowds. In *NIPS*, pages 2424–2432. MIT Press, Cambridge, MA, 2010.

[Whitehill *et al.*, 2009] Jacob Whitehill, Paul Ruvolo, Tingfan Wu, Jacob Bergsma, and Javier R. Movellan. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *NIPS*, pages 2035–2043. MIT Press, Cambridge, MA, 2009.

[Yan *et al.*, 2011] Yan Yan, Róomer Rosales, Glenn Fung, and Jennifer G. Dy. Active learning from crowds. In *ICML*, pages 1161–1168, Bellevue, WA, 2011.

[Yuen *et al.*, 2011] Man-Ching Yuen, Irwin King, and Kwong-Sak Leung. A survey of crowdsourcing systems. In *SocialCom*, pages 766–773, Boston, MA, 2011.

[Zhou *et al.*, 2012] Dengyong Zhou, John C. Platt, Sumit Basu, and Yi Mao. Learning from the wisdom of crowds by minimax entropy. In *NIPS*, pages 2204–2212. MIT Press, Cambridge, MA, 2012.

[Zhou *et al.*, 2014] Dengyong Zhou, Qiang Liu, John C. Platt, and Christopher Meek. Aggregating ordinal labels from crowds by minimax conditional entropy. In *ICML*, pages 262–270, Beijing, China, 2014.