Cost-Saving Effect of Crowdsourcing Learning*

Lu Wang and **Zhi-Hua Zhou**
National Key Laboratory for Novel Software Technology, Nanjing University
Collaborative Innovation Center of Novel Software Technology and Industrialization
Nanjing 210023, China
{wangl, zhouzh}@lamda.nju.edu.cn

Abstract

Crowdsourcing is widely adopted in many domains as a popular paradigm to outsource work to individuals. In the machine learning community, crowdsourcing is commonly used as a cost-saving way to collect labels for training data. While a lot of effort has been spent on developing methods for inferring labels from a crowd, few work concentrates on the theoretical foundation of crowdsourcing learning. In this paper, we theoretically study the cost-saving effect of crowdsourcing learning, and present an upper bound for the minimally-sufficient number of crowd labels for effective crowdsourcing learning. Our results provide an understanding about how to allocate crowd labels efficiently, and are verified empirically.

1 Introduction

Crowdsourcing [Brabham, 2008] is a popular paradigm to outsource work to individuals and is widely adopted in various domains [Yang *et al.*, 2012; Afuah and Tucci, 2012; Li *et al.*, 2013; LeBras *et al.*, 2013]. In the machine learning community, crowdsourcing is commonly used as a cost-saving way to collect labels for training data. Specifically, unlabeled instances are outsourced to a large group of people, also known as a crowd, who will label some of these instances on their own knowledge and get paid accordingly. "True" labels are inferred from these labels given by the crowd. Then, a model would be learned with these crowd-labeled data.

Our focus in this paper is the cost-saving effect of crowdsourcing learning. In many applications, it is expected that crowdsourcing is cost-saving. In other words, crowdsourcing is expected to be of high performance while saving money at the same time. As to the crowdsourcing learning problem, we hope to get high-quality labels from a crowd and learn a model with these crowd-labeled data at a low cost. To achieve this goal, some issues must be considered.

First of all, it is necessary to ensure that a high-quality label can actually be induced from labels given by a crowd. It is not always the truth, and for some professional problems, experts play a non-substitutable role. For example, few patients are willing to rely on a crowd of amateurs to make important diagnoses. In general, compared with labels given by experts, labels from a crowd are cheaper but less accurate.

For obtaining high-quality labels, a feasible solution is to get every instance labeled multiple times by a crowd in order to gather more information. We call such a single label from the crowd as a crowd label for convenience. With these multiple crowd labels, a direct way to infer the true label is the majority voting strategy. In addition, many other strategies have been proposed based on different assumptions [Dawid and Skene, 1979; Whitehill *et al.*, 2009; Raykar *et al.*, 2010; Karger *et al.*, 2011; Zhou *et al.*, 2012; Oyama *et al.*, 2013; Zhang *et al.*, 2014; Tian and Zhu, 2015].

Second, it is noteworthy that the number of crowd labels required is concerned. A single crowd label may be cheap, but if one hopes to get high-quality labels by increasing the number of crowd labels, the cost budget may still run out. For an extreme example, if high-quality labels can only be attained by using an infinite crowd, then crowdsourcing cannot save any cost unless crowd labels are free.

The third issue lies in the fact that in crowdsourcing learning, the crowdsourcing step is just used to collect labels for training data, whereas the performance of the model learned with these data, instead of the quality of labels themselves, is concerned. There are some studies about learning from weak teachers or crowd labels [Dekel and Shamir, 2009; Yan *et al.*, 2011; Urner *et al.*, 2012; Zhong *et al.*, 2015], and learning from crowd labels is also closely related to the label noise problem [Angluin and Laird, 1987; Kearns, 1998; Frénay and Verleysen, 2014]. A distinct point in our setting is for crowdsourcing learning, we can conveniently draw crowd labels for an instance over and over again. This paper focuses on the cost-saving effect of crowdsourcing learning. Given a crowdsourcing learning task, one must collect at least a number of crowd labels for PAC learning; we call this number as the "minimally-sufficient" number, and in this paper we present an upper bound. Note that the number of crowd labels corresponds to the cost, and thus, this is actually an upper bound about the minimal cost for crowdsourcing learning.

Overall, our theoretical study discloses how many crowd labels, to the least, should be acquired and how the labeling tasks should be allocated. Some of our results are validated

*This work was supported by the NSFC (61333014) and 973 Program (2014CB340501).

empirically.

In the following we start with preliminaries and then present our main results, followed by experiments and conclusions.

2 Preliminaries

In the machine learning community, crowdsourcing is often used to collect labels for training data. Then, a model is learned with these crowd-labeled instances. We call such a process as *crowdsourcing learning*. For convenience, we first give some definitions for crowdsourcing learning.

Definition 1 (The Ground-Truth Label, Crowd Label and Aggregated Label)**.** For a labeling task, an instance has an unknown true label called the *ground-truth label*. The instance is labeled one or multiple times by a crowd. Every single label given by a crowd is called a *crowd label*. The label inferred from these crowd labels is the *aggregated label*. A high-quality aggregated label disagrees with the ground-truth label with low probability. In other words, the error rate of a high-quality aggregated label is small.

Crowdsourcing could be divided into two steps. The first step (the *crowdsourcing step*) is distributing instances to a crowd and inferring the aggregated label from crowd labels. The second step (the *learning step*) is learning a model with crowd-labeled instances. We regard the total payment to the crowd as the cost of crowdsourcing learning.

2.1 The Crowdsourcing Step

In the crowdsourcing step, we have to ensure that high-quality aggregated labels can actually be induced from crowd labels and this process should be cost-saving compared with employing experts to get labels.

Given an instance, n crowd labels $Y_1, Y_2, \cdots, Y_n$ are collected. These crowd labels are independent and identically distributed. These labels have K possible values, where $K \geq 2$, that is, the label space $\mathcal{Y} = \{0, 1, \cdots, K-1\}$. For every $i \in \{1, 2, \cdots, n\}$, Y_i is distributed as

$$Y_i \sim \begin{bmatrix} 0, & 1, & \cdots, & K-1 \\ q_0, & q_1, & \cdots, & q_{K-1} \end{bmatrix}, \tag{1}$$

that is, $\Pr[Y_i = j] = q_j$. Without loss of generality, we assume that the ground-truth label $j^* \in \{0, 1, \cdots, K-1\}$. Moreover, to guarantee crowdsourcing to be viable, it is assumed that $\forall j \neq j^*, j \in \{0, 1, \cdots, K-1\}$, we have $q_{j^*} > q_j$. The assumption is reasonable since it simply implies that the ability of the crowd is better than a totally random behavior.

For simplicity, we adopt the *majority voting* strategy to induce the aggregated label. Let n_j denote the number of value-j labels in the n crowd labels. Then, for the aggregated label $\hat{Y}$, we have

$$\hat{Y} = \underset{j \in \{0,1,\cdots,K-1\}}{\arg\max} n_j. \tag{2}$$

To illustrate the label quality issue clearly, we consider the binary case first, that is, $K = 2$. We designate $p = q_{j^*}$ as the *crowd qualification*. In this case, p is the probability that a sample of crowd labels is correct in the sense of average and $p > \frac{1}{2}$. Let Z_i be an indicator variable:

$$Z_i = \begin{cases} 1, & Y_i = j^*; \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

To avoid ending in a tie, here we assume that n is an odd integer. The aggregated label disagrees with the ground-truth label with probability $\eta = \Pr\left[\frac{1}{n}\sum_{i=1}^{n} Z_i \leq \frac{1}{2}\right]$, where the expectation $\mathbb{E}[Z_i] = p$, and the variance $\mathbb{D}[Z_i] = p(1-p)$. By the central limit theorem (CLT), the error rate of the aggregated label satisfies

$$\eta = \Pr\left[\frac{1}{n}\sum_{i=1}^{n} Z_i \leq \frac{1}{2}\right] \tag{4}$$

$$= \Pr\left[\frac{\sum_{i=1}^{n} Z_i - np}{\sqrt{np(1-p)}} \leq \frac{\sqrt{n}(\frac{1}{2} - p)}{\sqrt{p(1-p)}}\right] \tag{5}$$

$$\text{(CLT)} \quad \approx \Phi\left(\frac{\sqrt{n}(\frac{1}{2} - p)}{\sqrt{p(1-p)}}\right), \tag{6}$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution:

$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t^2}{2}\right) \mathrm{d}t. \tag{7}$$

To show the functional relationship among p, n and η clearly, Figure 1 is plotted according to (6). It shows that the error rate of the aggregated label is controlled by the number of crowd labels in the crowdsourcing step.

In the following pages, unless with definite declaration, we will only talk about the binary case.

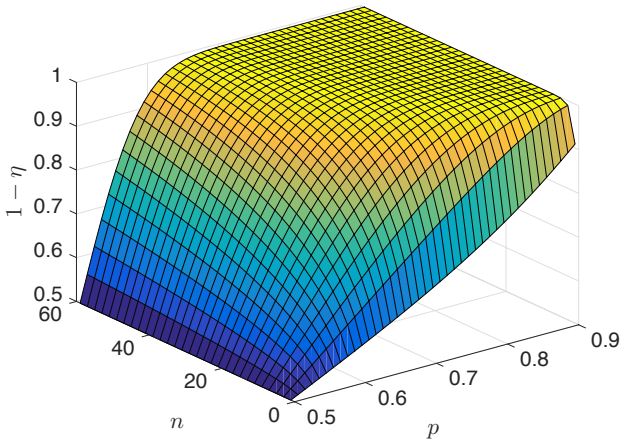


Figure 1: The error rate η of the aggregated label varies with the change of the number of crowd labels n and the crowd qualification p when adopting the majority voting strategy. It is plotted according to (6) where p ranges from 0.5 to 0.9 and n from 1 to 60.

2.2 The Learning Step

In the learning step, crowd-labeled instances will be given as training data of a learning algorithm. In this case, what we actually care about is the model learned with the crowd-labeled training data instead of the label quality of training data themselves. The issue is how the performance of the learned model is influenced by crowdsourcing and then how crowdsourcing should be used for learning.

A classic learning problem is described as below. $\mathcal{H}$ is a set of functions, of which the domain is $\mathcal{X}$ and the range is $\mathcal{Y}$. $\mathcal{H}$ is called the *hypothesis class* and every member in $\mathcal{H}$ is called a *hypothesis*. $\mathcal{D}$ is an unknown distribution over $\mathcal{X}$ and f is an unknown function $\mathcal{X} \to \mathcal{Y}$. For simplicity, it is assumed that $f \in \mathcal{H}$, which is called the *realizability assumption*. A *learner* is given access to an oracle $EX(f, \mathcal{D})$, which outputs instances one at a time randomly and independently according to $\mathcal{D}$ and labels them by f. The task is to identify f in $\mathcal{H}$.

$S = ((x_1, y_1), \cdots, (x_m, y_m))$ is a finite sequence in $\mathcal{X} \times \mathcal{Y}$. This is the learner's input and is generated by m calls of $EX(f, \mathcal{D})$. S is also known as the *training data* or the *training set*. The learner's output is $h_S \in \mathcal{H}$. To measure the success of the learner, we define the *true error* of a function $h : \mathcal{X} \to \mathcal{Y}$, to be

$$L_{(\mathcal{D},f)}(h) = \Pr_{x \sim \mathcal{D}}[h(x) \neq f(x)], \tag{8}$$

which is the probability that h disagrees with f on distribution $\mathcal{D}$. In the best case, h_S agrees with f in the whole domain, that is, $L_{(\mathcal{D},f)}(h_S) = 0$.

Since $\mathcal{D}$ and f are unknown to the learner, the true error is not available. We can not identify f directly by comparing the true error of different hypotheses. Instead, *empirical risk minimization (ERM)* is a common learning paradigm generating a hypothesis h that minimizes the *training error*

$$L_S(h) = \frac{|\{i \in \{1, 2, \cdots, m\} : h(x_i) \neq y_i\}|}{m}, \tag{9}$$

which is the proportion of cases where h disagrees with f on the training data S. Since the sequence S is drawn from $EX(f, \mathcal{D})$, intuitively, if a hypothesis h performs pretty well on a large S, the true error of h could be small with high probability.

Nonetheless, in crowdsourcing learning, labels of training data are inferred from crowd labels in the crowdsourcing step. The aggregated labels induced from the crowd labels are not always identical to the ground-truth label. In this case, we have no access to $EX(f, \mathcal{D})$. Instead, we assume that the training data are generated by a noisy oracle $EX_\eta(f, \mathcal{D})$. $EX_\eta(f, \mathcal{D})$ generates an instance by first drawing an instance (x, y) from $EX(f, \mathcal{D})$ and then flipping the label y with probability η. $EX_\eta(f, \mathcal{D})$ is weaker than $EX(f, \mathcal{D})$, since $EX_\eta(f, \mathcal{D})$ does not know the ground-truth label.

In the learning step, we learn a model with access to $EX_\eta(f, \mathcal{D})$. The label generated by $EX_\eta(f, \mathcal{D})$ corresponds to the aggregated label and η is the error rate of the aggregated label. η could be controlled in the crowdsourcing step by changing the number of crowd labels per instance. The total number of crowd labels used corresponds to the cost of crowdsourcing learning. Since our focus is the cost-saving effect of crowdsourcing learning, the number of crowd labels required in the whole process lies at the heart of the problem.

3 Main Results

Theorem 1. *Let $\mathcal{H}$ be a finite hypothesis class. Let $\delta, \epsilon \in (0, 1)$, $p \in (\frac{1}{2}, 1)$, $\gamma = 2p - 1$ and let m' be an integer that satisfies*

$$m' \geq \min_{\eta_b \in (0, 1-p]} \frac{4}{\gamma^2 \epsilon^2} \ln\left(\frac{2|\mathcal{H}|}{\delta}\right) \frac{1}{(1-2\eta_b)^2} \ln\frac{1}{\eta_b} \tag{10}$$

$$= \frac{4}{\gamma^2 \epsilon^2} \ln\left(\frac{2|\mathcal{H}|}{\delta}\right) \min\left\{C, \frac{1}{\gamma^2} \ln\frac{1}{1-p}\right\}, \tag{11}$$

where

$$C = \min_{x \in (0, \frac{1}{2})} \frac{1}{(1-2x)^2} \ln\frac{1}{x} \approx 3.5782. \tag{12}$$

Then, for any labeling function f and for any distribution $\mathcal{D}$, for which the realizability assumption holds, given i.i.d. sampled instances which will be labeled repeatedly by a crowd with crowd qualification p, we have that m' crowd labels are sufficient to learn a hypothesis h which holds that

$$\Pr[L_{(D,f)}(h) \geq \epsilon] \leq \delta. \tag{13}$$

Remarks: Theorem 1 is the main theoretical result of this paper. Literally speaking, it shows that, for a sufficiently large m, m crowd labels are sufficient to learn a model which is *probably* (with confidence $1 - \delta$) *approximately* (up to an error of ϵ) correct (PAC) for the crowdsourcing learning task. In other words, given a crowdsourcing learning task, one must collect at least a number of crowd labels for PAC learning; we call this number as the "minimally-sufficient" number. In addition, Theorem 1 presents an upper bound for the minimally-sufficient number. Note that the number of crowd labels corresponds to the cost, and thus, this is actually an upper bound about the minimal cost for crowdsourcing learning.

As will be shown in the proof to Theorem 1, η_b denotes the upper bound for the error rate of aggregated labels in the crowdsourcing step. In the crowdsourcing learning setting, η_b is adjustable by controlling the number of crowd labels per instance. Different η_bs correspond to different allocation schemes for crowd labels. To ensure a small η_b, more crowd labels per instance are required in the crowdsourcing step, while less instances will be required in the learning step. It is a trade-off between the number of crowd labels per instance and the number of instances for the crowdsourcing learning task. Note that

$$m' = O\left(\frac{1}{(1-2\eta_b)^2} \ln\frac{1}{\eta_b}\right), \tag{14}$$

and let

$$\tilde{m}(x) = \frac{1}{(1-2x)^2} \ln\frac{1}{x}, \tag{15}$$

where $x \in (0, \frac{1}{2})$. The functional relationship is plotted in Figure 2, and η_b^* is the minimum point of the function, that is,

$$\eta_b^* = \arg\min_{x \in (0, \frac{1}{2})} \frac{1}{(1-2x)^2} \ln\frac{1}{x} \approx 0.084. \tag{16}$$

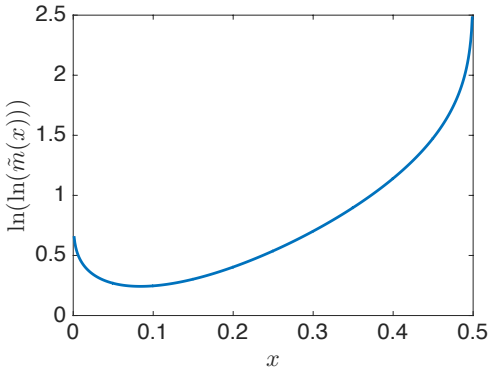


Figure 2: A graph for $\tilde{m}(x) = \frac{1}{(1-2x)^2} \ln\left(\frac{1}{x}\right)$, $0 < x < \frac{1}{2}$, of which the horizontal axis represents x and the vertical axis represents $\ln(\ln(\tilde{m}(x)))$.

Theorem 1 indicates η_b^* could be a good choice for η_b to reduce the number of crowd labels required. In this sense, Theorem 1 sheds a light on how to allocate crowd labels in order to save the total cost in crowdsourcing learning. Generally speaking, it inspires us to choose an appropriate η_b for the quality of aggregated labels by specifying the number of crowd labels per instance. To be specific, if aggregated labels are of poor quality, we should increase the number of crowd labels per instance rather than labeling more fresh instances; if a single crowd label or the aggregated label performs well enough, it is preferable to label more fresh instances.

Before presenting the proof to Theorem 1, we need to introduce some lemmas.

Lemma 1. *Given an instance with n crowd labels independently and identically distributed according to parameters $\boldsymbol{q} = [q_0, q_1, \cdots, q_{K-1}]$, where the ground-truth label $j^* \in \{0, 1, \cdots, K-1\}$, and $\gamma = \min_{j \neq j^*} q_{j^*} - q_j > 0$, for the error rate of the aggregated label to be upper-bounded by η_b, it is sufficient that*

$$n \geq \frac{2}{\gamma^2} \ln\left(\frac{K-1}{\eta_b}\right). \tag{17}$$

Proof. Let $Z_i^{j_1,j_2}$ be an indicator variable such that

$$Z_i^{j_1,j_2} = \begin{cases} 1, & Y_i = j_1; \\ -1, & Y_i = j_2; \\ 0, & \text{otherwise}, \end{cases} \tag{18}$$

of which the support is $[-1, 1]$ and the expectation $\mathbb{E}\left[Z_i^{j_1,j_2}\right] = q_{j_1} - q_{j_2}$. When adopting the majority voting strategy, the error rate of the aggregated label η satisfies:

$$\eta \leq \Pr[\exists j \neq j^*, n_{j^*} \leq n_j] \tag{19}$$

$$\text{(Union Bound)} \quad \leq \sum_{j \neq j^*} \Pr[n_{j^*} \leq n_j] \tag{20}$$

$$= \sum_{j \neq j^*} \Pr\left[\sum_{i=1}^{n} Z_i^{j^*,j} \leq 0\right] \tag{21}$$

$$\text{(Hoeffding)} \quad \leq \sum_{j \neq j^*} \exp\left(-\frac{(q_{j^*} - q_j)^2 n}{2}\right) \tag{22}$$

$$\leq (K-1) \exp\left(-\frac{\gamma^2 n}{2}\right). \tag{23}$$

To upper-bound η with η_b, by (23), we have

$$\eta \leq (K-1) \exp\left(-\frac{\gamma^2 n}{2}\right) \leq \eta_b. \tag{24}$$

Thus, if $n \geq \frac{2}{\gamma^2} \ln\left(\frac{K-1}{\eta_b}\right)$, we have $\eta \leq \eta_b$. □

Corollary 1. *In the binary case, given an instance with n crowd labels independently and identically distributed according to the crowd qualification p, where $p > \frac{1}{2}$ and $\gamma = 2p - 1$, for the error rate of the aggregated label to be upper-bounded by η_b, it is sufficient that*

$$n \geq \frac{2}{\gamma^2} \ln\left(\frac{1}{\eta_b}\right). \tag{25}$$

Proof. It is a corollary of Lemma 1. By setting $K = 2$, $p = q_{j^*}$ and $\gamma = 2p - 1$, we have this corollary in the binary case. □

Lemma 1 and Corollary 1 show that high-quality aggregated labels can actually be inferred from crowd labels. Specifically, the error rate of the aggregated label converges linearly to 0 with the number of crowd labels n, and the parameter γ (related to the crowd qualification p in the binary case) determines the rate of convergence. Similar results have been achieved [Wang and Zhou, 2015] under different assumptions [Dawid and Skene, 1979]. It is interesting to see that it is very similar to some theoretical results in ensemble learning which generates predictions by combining multiple weak learners [Zhou, 2012], suggesting that crowdsourcing learning might get inspirations from ensemble learning.

In general, labels given by experts cost much more than crowd labels. If the crowdsourcing step is indeed cost-saving, it is preferable to collect crowd labels to induce high-quality aggregated labels rather than employing experts for labeling.

It is significant to investigate to what extent the crowdsourcing step is cost-saving. For convenience, we assume that the number of crowd labels n is an odd integer. In this case, the error rate of the aggregated label is exactly

$$\eta = \sum_{i=1}^{\lfloor n/2 \rfloor} \binom{n}{i} p^i (1-p)^{n-i}. \tag{26}$$

Let c_{cr} and c_{em} denote the cost per crowd label and the cost per label given by experts respectively. Let η^* denote the

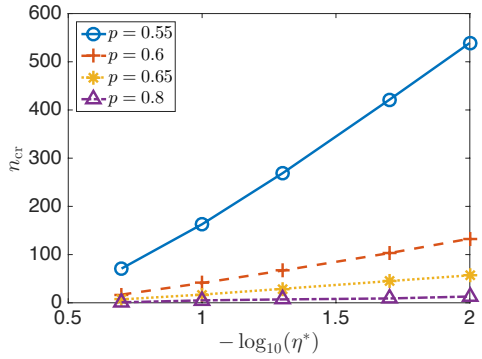


Figure 3: The functional relationship between n_{cr} and η^* for different p. n_{cr} is the number of required crowd labels, η^* is the required error rate for the aggregated label and p is the crowd qualification. The figure is plotted according to (27), where η^* ranges from 0.01 to 0.2.

error rate that experts can achieve. In the crowdsourcing step, η^* is the required error rate of the aggregated label and it is expected to ensure that $\eta \leq \eta^*$. Let n_{cr} denote the minimum number of crowd labels to satisfy $\eta \leq \eta^*$. Specifically,

$$n_{\text{cr}} = \min \left\{ n \in \mathbb{N}^* : \sum_{i=1}^{\lfloor n/2 \rfloor} \binom{n}{i} p^i (1-p)^{n-i} \leq \eta^* \right\}. \tag{27}$$

In this case the crowdsourcing step is cost-saving if and only if

$$n_{\text{cr}} \cdot c_{\text{cr}} < c_{\text{em}}. \tag{28}$$

The values of p, η^*, c_{cr} and c_{em} jointly determine whether (28) is satisfied.

The functional relationship between n_{cr} and η^* for different p is shown in Figure 3 according to (27).

As Figure 3 shows, given an instance, as long as the labeling of the crowd is better than totally random behavior, which means $p > \frac{1}{2}$, the error rate of the aggregated label converges to 0 with the number of crowd label n. In addition, the rate of convergence determined by the crowd qualification p is also very important for the number of required crowd labels and thus important for the cost of the crowdsourcing step. For example, given $\eta^* = 0.05$, if $p = 0.65$, we have $n_{\text{cr}} = 29$; while if $p = 0.55$, we have $n_{\text{cr}} = 269$. In the first case, if the crowd label price c_{cr} is relatively small, the crowdsourcing step is cost-saving. In the second case, it is hard to make similar conclusions. Generally speaking, only if crowd labels are cheap and perform well enough, the crowdsourcing step is cost-saving.

In the learning step, if ground-truth labels are available, some number of training instances are enough for PAC learning as follows.

Lemma 2 ([Blumer *et al.*, 1986]). *Let $\mathcal{H}$ be a finite hypothesis class. Let $\delta, \epsilon \in (0,1)$ and let m be an integer that satisfies*

$$m \geq \frac{1}{\epsilon} \ln \left(\frac{|\mathcal{H}|}{\delta} \right). \tag{29}$$

Then, for any f and $\mathcal{D}$, for which the realizability assumption holds, given a sequence S of size m generated by $EX(f, \mathcal{D})$, if a hypothesis $h \in \mathcal{H}$ satisfies that $L_S(h) = 0$, we have

$$\Pr[L_{(D,f)}(h) \geq \epsilon] \leq \delta. \tag{30}$$

However, in crowdsourcing learning, only the aggregated labels are available. The learner has access to $EX_\eta(f, \mathcal{D})$ rather then $EX(f, \mathcal{D})$. For a function $h : \mathcal{X} \to \mathcal{Y}$, let the true error $L_{(\mathcal{D},f)}(h)$ be abbreviated as $\ell(h)$ and let $\ell'(h)$ denote the probability that a labeled instance from $EX_\eta(f, \mathcal{D})$ disagrees with h. Then, we have

$$\ell'(h) = \ell(h) \cdot (1-\eta) + (1-\ell(h)) \cdot \eta \tag{31}$$

$$= (1-2\eta) \cdot \ell(h) + \eta. \tag{32}$$

If $\eta < \frac{1}{2}$, $\ell'(h)$ is monotonically increasing with $\ell(h)$. The relationship between ℓ and ℓ' suggests that some number of training instances could still be enough for PAC learning with access to $EX_\eta(\mathcal{D}, f)$ as below.

Lemma 3 ([Angluin and Laird, 1987]). *Let $\mathcal{H}$ be a finite hypothesis class. Let $\delta, \epsilon \in (0,1)$, $\eta_b \in (0, \frac{1}{2})$ and let m be an integer that satisfies*

$$m \geq \frac{2}{\epsilon^2 (1-2\eta_b)^2} \ln \left(\frac{2|\mathcal{H}|}{\delta} \right). \tag{33}$$

Then, for any f and $\mathcal{D}$, for which the realizability assumption holds, given a sequence S of size m generated by $EX_\eta(f, \mathcal{D})$, where $\eta \leq \eta_b$, if a hypothesis $h \in \mathcal{H}$ minimizes $L_S(h)$, we have

$$\Pr[L_{(D,f)}(h) \geq \epsilon] \leq \delta. \tag{34}$$

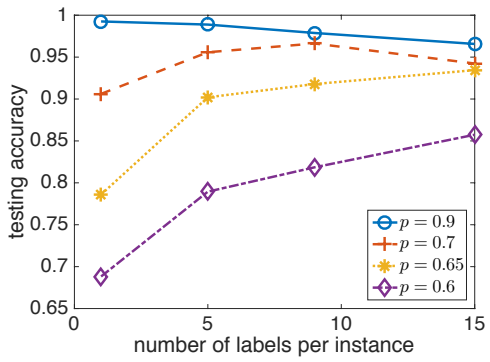
Lemma 3 shows that although the aggregated labels are not ground-truth, the ERM rule over a finite hypothesis class will still be *probably* (with confidence $1-\delta$) *approximately* (up to an error of ϵ) correct (PAC). In other words, a well-performed model can actually be learned with crowd-labeled training data.

It is noteworthy that Lemma 3 is originally a theoretical result for the label noise setting. However, in the crowdsourcing learning setting, unlike in the label noise setting, η_b is adjustable by changing the number of crowd labels per instance. Estimating the noise rate is a challenging task in the label noise setting [Menon *et al.*, 2015; Liu and Tao, 2016], but beyond the scope of this paper. For the moment, we just explore the effect of η_b on the cost for crowdsourcing learning while ignoring the estimation of η_b.

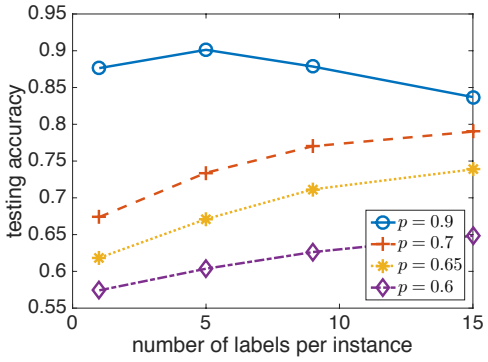
Corollary 1 shows how many crowd labels per instance are enough to make the error rate of the aggregated label to be upper-bounded by η_b and Lemma 3 shows that with the upper bound η_b Now we can give the proof to Theorem 1.

Proof to Theorem 1. Given η_b, Corollary 1 gives the upper bound of the minimally-sufficient number of crowd labels per instance, that is,

$$n = \frac{2}{\gamma^2} \ln \left(\frac{1}{\eta_b} \right); \tag{35}$$



(a) *Mushrooms*



(b) *Splice*

Figure 4: Testing accuracy versus number of labels per instance for different crowd qualification p with fixed total number of crowd labels. Average results from 20 random partitions for testing data are reported.

Lemma 3 gives the upper bound of the minimally-sufficient number of crowd-labeled instances, that is,

$$m = \frac{2}{\epsilon^2(1-2\eta_b)^2} \ln\left(\frac{2|\mathcal{H}|}{\delta}\right). \tag{36}$$

By (35) and (36), the sufficient number of crowd labels is

$$m' \geq n \cdot m \tag{37}$$

$$= \frac{4}{(1-2\eta_b)^2\gamma^2\epsilon^2} \ln\left(\frac{2|\mathcal{H}|}{\delta}\right) \ln\left(\frac{1}{\eta_b}\right), \tag{38}$$

where $\eta_b \in (0, 1-p]$. See η_b as a variable and minimize (38), then we have (11). □

4 Experiments

To verify our theoretical results of Theorem 1, we design some experiments following an empirical study on repeated labeling [Sheng *et al.*, 2008].

Two real-world datasets [1] are adopted, of which the dataset *Mushrooms* has 112 features and 8124 instances, while the dataset *Splice* has 60 features and 3175 instances. For each dataset, 30% of instances are used as testing data and the others as a pool from which instances are sampled for training. A training instance is labeled one or multiple times. These labels are independent and identically distributed. Each label is identical to the ground-truth label with probability p. We adopt the majority voting strategy to induce aggregated labels of training data.

We devise four ways to allocate crowd labels, of which the numbers of crowd labels per instance are $[1, 5, 9, 15]$ respectively and the total number of crowd labels is fixed to be 1800. The issue in the experiments is whether it is worthwhile to label an instance multiple times rather than labeling more fresh instances.

J48 decision trees in Weka [Witten and Frank, 1999] are used in our experiments, and results on *Mushrooms* and *Splice* are shown in Figure 4. Testing accuracy values are calculated on testing data with ground-truth labels. Average results from 20 random partitions for testing data are reported.

We discuss in detail about the results on the dataset *Mushrooms* as an example to demonstrate the experimental results clearly.

- $p = 0.9$. A single crowd label performs pretty well. Labeling an instance multiple times is a waste compared with labeling more fresh instances.
- $p = 0.7$. A single crowd label performs well. Labeling an instance several times is appropriate. However, as the number of crowd labels per instance increases, the gain of performance brought by repeated labeling is reduced. When the quality of the aggregated labels is good enough, labeling more fresh instances is preferable.
- $p = 0.6$ and $p = 0.65$. A single crowd label is of poor quality. We have to label an instance multiple times to improve the quality of aggregated labels.

As to the dataset *Splice*, similar results are presented as well, as observed in Figure 4. Moreover, some related empirical results can be found in some previous work [Sheng *et al.*, 2008; Ipeirotis *et al.*, 2014].

5 Conclusion

In this paper, we theoretically investigate some basic issues about crowdsourcing learning. In particular, we present an upper bound for the minimally-sufficient number of crowd labels, i.e., the minimal cost required for effective crowdsourcing learning. Our theoretical results also shed a light on how to allocate crowd labels for cost-saving.

This is a very preliminary attempt for the theoretical foundation of crowdsourcing learning. Further studies about more complex assumptions and labeling strategies are desired for future work.

[1] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

References

[Afuah and Tucci, 2012] Allan Afuah and Christopher L Tucci. Crowdsourcing as a solution to distant search. *Academy of Management Review*, 37(3):355–375, 2012.

[Angluin and Laird, 1987] Dana Angluin and Philip D. Laird. Learning from noisy examples. *Machine Learning*, 2(4):343–370, 1987.

[Blumer *et al.*, 1986] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Classifying learnable geometric concepts with the Vapnik-Chervonenkis dimension. In *STOC*, pages 273–282, 1986.

[Brabham, 2008] Daren C Brabham. Crowdsourcing as a model for problem solving: An introduction and cases. *Convergence*, 14(1):75–90, 2008.

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

[Dekel and Shamir, 2009] Ofer Dekel and Ohad Shamir. Good learners for evil teachers. In *ICML*, pages 233–240, 2009.

[Frénay and Verleysen, 2014] Benoît Frénay and Michel Verleysen. Classification in the presence of label noise: A survey. *IEEE Trans. on Neural Networks and Learning Systems*, 25(5):845–869, 2014.

[Ipeirotis *et al.*, 2014] Panagiotis G. Ipeirotis, Foster J. Provost, Victor S. Sheng, and Jing Wang. Repeated labeling using multiple noisy labelers. *Data Mining and Knowledge Discovery*, 28(2):402–441, 2014.

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

[Kearns, 1998] Michael J. Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45(6):983–1006, 1998.

[LeBras *et al.*, 2013] Ronan LeBras, Richard Bernstein, Carla P. Gomes, Bart Selman, and R. Bruce van Dover. Crowdsourcing backdoor identification for combinatorial optimization. In *IJCAI*, pages 2840–2847, 2013.

[Li *et al.*, 2013] Boyang Li, Stephen Lee-Urban, George Johnston, and Mark Riedl. Story generation with crowdsourced plot graphs. In *AAAI*, pages 598–604, 2013.

[Liu and Tao, 2016] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 38(3):447–461, 2016.

[Menon *et al.*, 2015] Aditya Krishna Menon, Brendan van Rooyen, Cheng Soon Ong, and Bob Williamson. Learning from corrupted binary labels via class-probability estimation. In *ICML*, pages 125–134, 2015.

[Oyama *et al.*, 2013] Satoshi Oyama, Yukino Baba, Yuko Sakurai, and Hisashi Kashima. Accurate integration of crowdsourced labels using workers' self-reported confidence scores. In *IJCAI*, pages 2554–2560, 2013.

[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.

[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 *KDD*, pages 614–622, 2008.

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

[Urner *et al.*, 2012] Ruth Urner, Shai Ben-David, and Ohad Shamir. Learning from weak teachers. In *AISTATS*, pages 1252–1260, 2012.

[Wang and Zhou, 2015] Wei Wang and Zhi-Hua Zhou. Crowdsourcing label quality: A theoretical analysis. *Science China Information Sciences*, 58(11):1–12, 2015.

[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 *Advances in Neural Information Processing Systems 22*, pages 2035–2043, 2009.

[Witten and Frank, 1999] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, San Francisco, CA, 1999.

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

[Yang *et al.*, 2012] Dejun Yang, Guoliang Xue, Xi Fang, and Jian Tang. Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, pages 173–184, 2012.

[Zhang *et al.*, 2014] Yuchen Zhang, Xi Chen, Dengyong Zhou, and Michael I. Jordan. Spectral methods meet EM: A provably optimal algorithm for crowdsourcing. In *NIPS*, pages 1260–1268, 2014.

[Zhong *et al.*, 2015] Jinhong Zhong, Ke Tang, and Zhi-Hua Zhou. Active learning from crowds with unsure option. In *IJCAI*, pages 1061–1068, 2015.

[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, 2012.

[Zhou, 2012] Zhi-Hua Zhou. *Ensemble Methods: Foundations and Algorithms*. CRC Press, Boca Raton, FL, 2012.