Adaptive Deep Learning from Crowds

Hang Yang[1] , **Zhiwu Li**[1,*] , and **Witold Pedrycz**[2,3]
[1]Macau Institute of Systems Engineering, Macau University of Science and Technology
[2]Department of Electrical and Computer Engineering, University of Alberta
[3]Systems Research Institute, Polish Academy of Sciences
hangy03@student.must.edu.mo, zwli@must.edu.mo, wpedrycz@ualberta.ca

Abstract

In the data-driven era, collecting high-quality labeled data requiring human labor is a common approach for training data-hungry models, called crowdsourcing. Recently, end-to-end learning from crowds has shown its flexibility and practicality. However, existing works in an end-to-end manner focus on learning after collecting labels, which results in noisy annotations and also requires cost. Inspired by computerized adaptive testing, we argue that the characteristics of workers should be mined as soon as possible to make the best use of talents. To this end, we propose an adaptive learning from crowds method, AdaCrowd, as a cost-effective solution. Specifically, we propose a probabilistic model to capture the informativeness of possible instances for each worker. The informativeness is considered to be the uncertainty of the annotation prediction model output in its current status. The adaptive learning procedure is optimized by maximizing data likelihood and can be used with existing crowdsourcing models. Extensive experiments are conducted on real-world datasets, LabelMe and CIFAR-10H. The experimental results, e.g., the reduction of annotations without performance degradation, demonstrate the effectiveness.

1 Introduction

Recent years have witnessed the remarkable success of deep neural network training on large-scale datasets [LeCun *et al.*, 2015; Vaswani *et al.*, 2017]. Crowdsourcing is a useful way to collect labeled data from human workers for these data-hungry deep models [Han *et al.*, 2025]. Task owner pays rewards to workers for their annotations in crowdsourcing platforms, and every annotation carries a cost, e.g., 100 annotations of a text classification task cost 2 Yuan in a Chinese platform maintained by Netease Fuxi[1]. However, the crowd-supervised dataset inevitably contains incorrect, noisy, and redundant labels, which causes the task budget to increase [Nguyen *et al.*, 2024; Chen *et al.*, 2022]. The simplest solution is to conduct an admission test to filter out insufficient workers. Since this in-advance test introduces additional test costs and wastes workers' time, many efforts have been made to reduce crowdsourcing costs according to knowledge from data [Wang and Zhou, 2016; Fang *et al.*, 2018; Yang *et al.*, 2018; Miao *et al.*, 2023].

A popular model training approach in crowdsourcing is the co-training of target models and label correction mechanisms in an end-to-end fashion. The basic paradigm is connecting the crowd layer, i.e., transition matrices of workers, behind the original classifier [Rodrigues and Pereira, 2018]. This end-to-end training enables any improvement of deep learning to be applied in crowdsourcing and shows flexibility and practicality. Existing works in cost saving focus on modeling cost complexity under quality requirements or dynamically determining the price for each worker. However, these theoretical results under label aggregation, such as majority voting, are hard to apply in end-to-end deep learning from crowds, where the hypothesis space and sample complexity are hard to analyze under the Probably Approximately Correct framework [Haussler and Warmuth, 1995].

To this end, we aim to save costs with end-to-end deep learning. Inspired by computerized adaptive testing (CAT) [Ghosh and Lan, 2021], the main idea is to estimate the ability of human workers using as few unlabeled instances as possible. CAT is widely used in education assessments such as GMAT [Rudner, 2010] and Duolingo [Brenzel and Settles, 2017], where the question is adaptively selected based on students' responses to estimate their ability efficiently. The selection criterion is usually informativeness, e.g., Fisher information, where the maximum value means the difficulty of the exercise equals the ability of the student, and the correct probability is 50%. However, in crowdsourcing, the labels of instances are unknown initially; therefore, the informativeness cannot be measured directly.

To tackle the above challenge, we propose an adaptive learning from crowds method, called AdaCrowd, to efficiently estimate worker parameters and train the target model using as few annotations as possible. Similar to CAT, the core goal is to choose the most informative instance from the candidate set for workers to label. When the worker labels the selected instance, the model parameter updates, and the new instance is selected for the worker until convergence. Note that in our method, there is no need to label some instances in ad-

[*] Corresponding author

[1] https://zb.163.com/mark/task

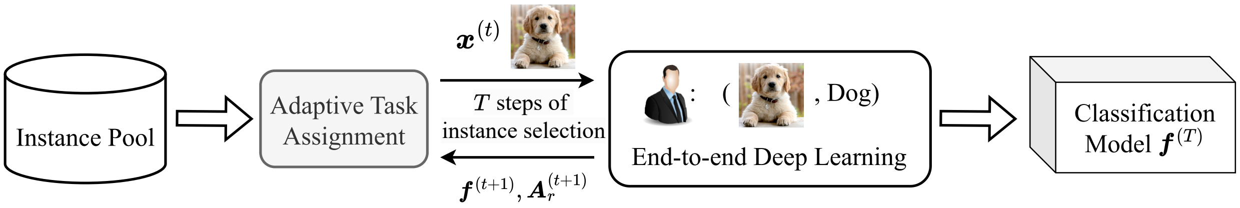

Figure 1: A toy example of the adaptive deep learning from crowds: At each annotation step t, an instance $x^{(t)}$ is selected and assigned to workers for annotating. Their annotations are used for end-to-end training. The updated classifier parameter $f^{(t+1)}$ and worker parameter $A_r^{(t+1)}$ are used to select proper instance $x^{(t+1)}$ in the next step $t+1$. After T steps, the crowdsourcing procedure will finish, and the final classifier $f^{(t)}$ can be used to predict unseen instances.

vance to test the worker's ability, although this test is adopted on many platforms. From this perspective, AdaCrowd can be considered an "unsupervised test".

Specifically, we first introduce an annotation probabilistic model to simulate the annotation procedure. Then, the informativeness of instances is assessed based on the uncertainty in the prediction model's outputs. This approach enables a more strategic allocation of workers to instances, ensuring that their expertise is leveraged most effectively. The adaptive learning process in AdaCrowd is fine-tuned by maximizing data likelihood, making it compatible with existing crowdsourcing methodologies.

Through empirical evaluation and visualization on real-world datasets, including LabelMe and CIFAR-10H, we showcase the effectiveness of the AdaCrowd. The results affirm that AdaCrowd can reduce the number of necessary training annotations without decreasing model performance.

The contributions of this research are summarized as follows:

- We propose AdaCrowd, a cost-efficient solution for crowdsourcing. To the best of our knowledge, AdaCrowd is the first research work on cost-saving in end-to-end deep learning from crowds.
- To adaptively choose an instance for each worker to annotate in every epoch, we incorporate the idea similar to computerized adaptive testing. We believe our work can help improve related CAT methods.
- The proposed adaptive crowdsourcing coincides with the out-of-distribution via Evidence Deep Learning. Our task is another application of the Theory of Evidence.

2 Related Works

Learning from Crowds. A fundamental model in crowdsourcing is the Dawid-Skene (D&S) model [Dawid and Skene, 1979]. The D&S model assumes that each worker possesses a confusion matrix, which outlines the probabilities of their annotations matching the true labels. Recently, the end-to-end models that jointly learn the deep neural network and worker parameters directly arose as an EM-free approach. The first work in end-to-end models is CrowdLayer [Rodrigues and Pereira, 2018], which applies the learnable crowd layer after the classifier for confusion modeling. After that, TraceReg [Tanno *et al.*, 2019] introduces a regularization term in mapping the classifier output onto the worker-specified output. Besides, CoNAL [Chu *et al.*, 2021] goes further by distinguishing a common confusion from the individual confusion of each worker. Coupled Confusion Correction [Zhang *et al.*, 2024] simultaneously trains two models to correct the transition matrices learned from each other.

Active learning has also been introduced in crowdsourcing with different models such as logistic regression [Yan *et al.*, 2011], Bayesian networks [Zhao *et al.*, 2014], and active learning with SVM [Zhong *et al.*, 2015]. Our work is nascent and differs from these works since, in end-to-end learning, existing methods are hard to scale in deep networks with confusion layers. The proposed approach, adaptively assigning the best instance to workers, can be seen as a prepositive work that could serve as a plug-in to enhance existing works.

Crowdsourcing Cost. The pioneering work in crowdsourcing cost-saving is the adaptive task-assigning with gold standard tasks for classification [Ho *et al.*, 2013]. This work builds the relationship between the target error rate and the times of querying humans. After this, SmartCrowd formats the problem of worker-to-task assignment in knowledge-intensive crowdsourcing, e.g., Wiki writing, as an optimization problem [Basu Roy *et al.*, 2015]. Probably approximately correct (PAC) is also used to study the cost-saving effect of crowdsourcing learning theoretically, and an upper bound for the minimally sufficient number of crowd labels can be given [Wang and Zhou, 2016]. Then, the cost complexity, also based on PAC, is proposed to model the trade-off between costs and quality [Fang *et al.*, 2018].

3 Background

3.1 End-to-end Learning from Crowds

Notation. Suppose that there are R workers labeling N instances as belonging to K possible classes. $\boldsymbol{x}_i \in \mathcal{X}$ refers to the i-th instance, and $\boldsymbol{y}_{ri} \in \mathcal{Y}$ refers to the one-hot label from the r-th worker on the i-th instance, where $\mathcal{X}$ is the instance space depending on task and $\mathcal{Y}$ is the label space, i.e., a $K-1$ probability simplex is considered:

$$\forall \boldsymbol{y}_{ir} \in \mathcal{Y} : \sum_{k=1}^{K} y_{rik} = 1, y_{rik} \geq 0. \tag{1}$$

Educational Test	Crowdsourcing	Notation
Questions	Instances	$\boldsymbol{x}_i$
Students	Workers	$\boldsymbol{A}_r$
Responses	Annotations	$\boldsymbol{y}_{ri}$
Item response theory	End-to-end deep model	$\boldsymbol{A}_r\boldsymbol{f}(\boldsymbol{x}_i)$
Test information	Uncertainty	$\boldsymbol{\alpha}_{ri}$
Question selection	Instance selection	π

Table 1: Corresponding concepts in crowdsourcing task assignment and educational test.

We denote the instance set as $X = \{\boldsymbol{x}_i\}_{i=1}^N$, the annotation set as $Y = \{\boldsymbol{y}_{ir}\}$, where $\boldsymbol{y}_{ir} = [y_{ri1}, y_{ri2}, \ldots, y_{riK}]$, and the unknown instance truth set as $Z = \{z_i\}_{i=1}^N$. Let us represent the classifier as $\boldsymbol{f} : \mathcal{X} \to \mathcal{Y}$, and the workers' transition matrices as $\{\boldsymbol{A}_r\}_{r=1}^R$, where $\boldsymbol{A}_r$ satisfies that its columns are conditional probability distributions.

End-to-End Training. In the general end-to-end training paradigm, classifier $\boldsymbol{f}$ and worker parameters $\{\boldsymbol{A}_r\}_{r=1}^R$ are connected to a deeper network. Note that the classifier is sequentially combined with a feature extractor and a linear layer. The objective function is expressed as follows:

$$\min_{\boldsymbol{f},\{\boldsymbol{A}_r\}_{r=1}^R} -\frac{1}{|Y|}\sum_{\boldsymbol{y}_{ri}\in Y}\sum_{k=1}^{K} y_{rik}\log[\boldsymbol{A}_r\boldsymbol{f}(\boldsymbol{x}_i)]_k. \qquad (2)$$

After proper network initialization, optimal parameters of crowdlayer $\{\boldsymbol{A}_r\}_{r=1}^R$ and classifier $\boldsymbol{f}$ can be estimated with stochastic gradient descent. Assuming that the ground truth confusion matrices are $\{\boldsymbol{A}_r^{\natural}\}_{r=1}^R$, the difference between the learned parameters $\{\boldsymbol{A}_r\}_{r=1}^R$ and $\{\boldsymbol{A}_r^{\natural}\}_{r=1}^R$ are bounded [Ibrahim *et al.*, 2023].

3.2 Problem Setting

We focus on cost-saving by reducing the number of annotations in this research work. The intuitive way is improving the task assignment mechanism, similar to the CAT [Zhuang *et al.*, 2022]. The correspondence between CAT and crowdsourcing task assignment is shown in Table 1. The core concept is instance selection, which makes a decision on which instance and possible annotation can best help learn the target model. Here, we give the setting of instance selection.

Instance Selection. Given instance pool X and worker pool Q, our task is to design a strategy π to select an instance, i.e., $\boldsymbol{x}_r^{(t)} \sim \pi^{(t)}(X, r)$, in each step t according to current model parameters $\boldsymbol{f}^{(t)}$ and $\{\boldsymbol{A}_r^{(t)}\}_{r=1}^R$. The selection strategy should balance two goals: *importance* of each instance for estimating the worker parameter and *coverage* of all selected instances for training the target model.

With the instance selection, adaptive crowdsourcing is described as follows.

Adaptive Crowdsourcing. At step t $(1 \leq t \leq T)$, there is one instance $\boldsymbol{x}_r^{(t)}$ sampled from distribution $\pi^{(t)}(X, r)$ for worker r. Then, the worker labels this instance, i.e., gives a one-hot observation. Both the classifier and worker parameters in step $(t + 1)$ are updated with this new labeled sample $(\boldsymbol{x}_r^{(t)}, \boldsymbol{y}_r^{(t)})$. The number of steps T may vary with workers. For simplicity and comparability, we set T to be number-fixed or proportion-fixed as a hyperparameter. After learning from selected annotations sequentially after T steps, we get the target model $\boldsymbol{f}^{(T)}$. The performance of the target model is measured by computing accuracy in an unseen test set.

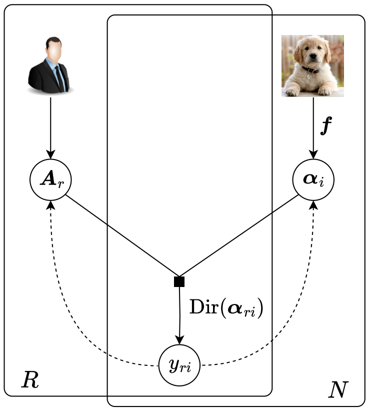

Figure 2: Graphical model of annotation generation. The annotation y_{ri} is a sample from the Dirichlet distribution with parameter $\boldsymbol{\alpha}_{ri}$, which is influenced by the worker's parameter $\boldsymbol{A}_r$ and the encoded instance feature $\boldsymbol{\alpha}_i$.

4 Method

4.1 Probabilistic Model

To measure the uncertainty of an instance on the current model state, we model the annotation process as a probabilistic model, as shown in Figure 2. Instead of treating $\boldsymbol{f}(\boldsymbol{x}_i)$ as the softmax output, we take the output $\boldsymbol{f}(\boldsymbol{x}_i)$ as the parameter of the Dirichlet distribution $\boldsymbol{\alpha}_i$ about the multinomial probability of this instance. Therefore, $\boldsymbol{A}_r$ is not the confusion matrix with the sum of each row equaling 1, but a transformation matrix without simplex constraint. After labeling by worker r, the parameter of Dirichlet distribution changes to $\boldsymbol{\alpha}_{ri} = \boldsymbol{A}_r\boldsymbol{\alpha}_i$. Then, the annotation can be seen as a sample from posterior Dirichlet, i.e., $\boldsymbol{p}_{ri} \sim \mathrm{Dir}(\boldsymbol{\alpha}_{ri})$. Therefore, denoting the annotation set as Y, the likelihood of data is:

$$L(X, Y) = \mathbb{E}_{\boldsymbol{p}_{ri}\sim\mathrm{Dir}(\boldsymbol{\alpha}_{ri})}[\prod_{i=1}^{N}\prod_{\boldsymbol{y}_{ri}\in Y}\prod_{k=1}^{K} y_{rik}^{p_{rik}}], \qquad (3)$$

where $p_{rik} = [\boldsymbol{p}_{ri}]_k$.

The likelihood is the function of parameters $\boldsymbol{f}, \{\boldsymbol{A}_r\}_{r=1}^R$, therefore, the maximum likelihood estimation (MLE) of this probability model is computed by minimizing the negative of log-likelihood, i.e.,

$$\boldsymbol{f}, \{\boldsymbol{A}_r\}_{r=1}^R = \operatorname*{argmin}_{\boldsymbol{f},\{\boldsymbol{A}_r\}_{r=1}^R} \mathbb{E}_{\boldsymbol{p}_{ri}\sim\mathrm{Dir}(\boldsymbol{\alpha}_{ri})}[-\log L(X, Y)]. \qquad (4)$$

4.2 Overview of AdaCrowd

The model architecture is illustrated in Figure 3. The main components are the backbone network, the evidential learning

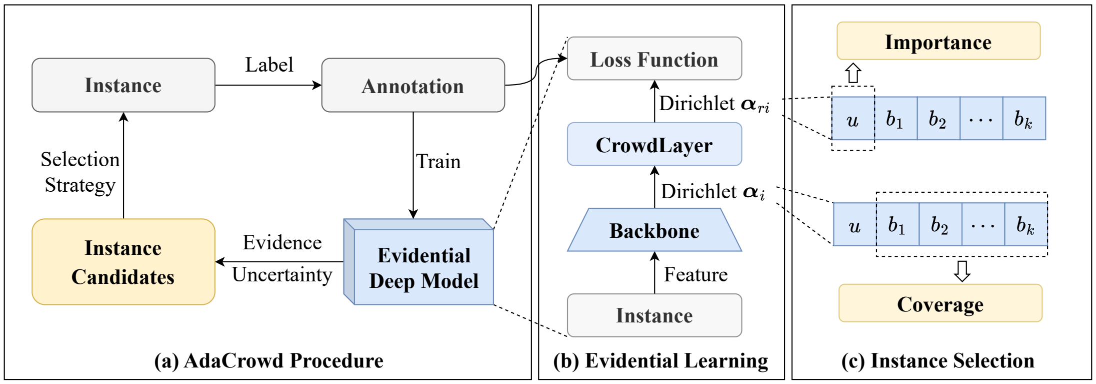

Figure 3: Model overview of AdaCrowd.

module, and the instance selection module. The backbone network is a task-specific pre-trained model, and the setting of the backbone is shown in the experiments. We describe the overview of the modules as follows.

- Evidential Learning Module: We introduce evidential learning to measure the uncertainty. By modifying the traditional CrowdLayer, evidential learning can be applied to existing end-to-end crowdsourcing models. The modified network is optimized directly by log-likelihood without sampling.
- Instance Selection Module: We design a mechanism with uncertainty and evidence for selecting the most suitable ones for workers. The importance and overall coverage are both considered for making the best of both worker parameter estimation and target model training.

4.3 Evidential Learning

Evidential deep learning is currently a popular technique in out-of-distribution detection, aiming to find samples that never appear in the training set [Sensoy *et al.*, 2018]. Here we show how to compute belief and uncertainty for an instance $\boldsymbol{x} \in \mathcal{X}$.

With backbone network $\boldsymbol{f}$, the last softmax layer is replaced with a linear layer, and the output shape is the number of classes K. Therefore, the output of the backbone is:

$$\boldsymbol{e} = [e_k]_{k=1}^K = [e_1, e_2, \cdots, e_K] = \boldsymbol{f}(\boldsymbol{x}). \tag{5}$$

For each possible class k, the belief mass is denoted as b_k, and the overall uncertainty mass is denoted as u. These mass values are non-negative, and their summary is 1, i.e., $u + \sum_{k=1}^K b_k = 1$. To construct these mass values, the $\boldsymbol{e}$ is normalized to (u, $\boldsymbol{b}$), i.e.,

$$S = \sum_{k=1}^K (e_k + 1), u = \frac{K}{S}, b_k = \frac{e_k}{S}, \tag{6}$$

where e_k is obtained from Eq. (5) and referred to as evidence because it quantifies the level of support gathered from data for classifying a sample into a specific class.

Therefore, the above value can be used to define a Dirichlet distribution with parameter $\boldsymbol{\alpha}$, where $\alpha_k = e_k + 1$. The density of the Dirichlet distribution is:

$$\mathrm{Dir}(\boldsymbol{p};\boldsymbol{\alpha}) = \frac{1}{B(\boldsymbol{\alpha})}\prod_{i=1}^{K} p_i^{\alpha_i - 1}, \tag{7}$$

where $\boldsymbol{p}$ is a probability satisfying simplex constraint, $\alpha_k = e_k + 1 = [\boldsymbol{f}(\boldsymbol{x})]_k + 1$, and the length of Dirichlet distribution is S in Eq. (6).

In the crowdsourcing setting, we assume the instance is $\boldsymbol{x}_i$, and the worker's parameter is $\boldsymbol{A}_r$. After inference of the backbone, the CrowdLayer is used to integrate the worker's characteristics into the instance's final annotation, i.e.,

$$\boldsymbol{\alpha}_{ri} = \boldsymbol{A}_r \boldsymbol{\alpha}_i = \boldsymbol{A}_r(\boldsymbol{f}(\boldsymbol{x}_i) + \mathbf{1}), \tag{8}$$

where $\mathbf{1}$ is the fulling one array.

According to Eq. (4), the loss function is as follows. For simplicity, we denote $\mathcal{L}(\boldsymbol{x}_{ri}, \boldsymbol{y}_{ri}; \boldsymbol{f}, \boldsymbol{A}_{\boldsymbol{r}})$ as $\mathcal{L}(r, i)$.

$$\mathcal{L}(r,i) = \int_{\boldsymbol{p}\in\mathcal{S}_k} \left[\sum_{k=1}^K -y_{rik}\log\left(p_{rik}\right)\right] \mathrm{Dir}(\boldsymbol{p};\boldsymbol{\alpha}) d\boldsymbol{p}, \tag{9}$$

where $\mathrm{Dir}(\boldsymbol{p};\boldsymbol{\alpha}) = \frac{1}{B(\boldsymbol{\alpha}_{ri})}\prod_{k=1}^K p_{rik}^{\alpha_{rik}-1}$, and $\mathcal{S}_k$ is the open set of $K-1$ simplex.

Similar to the study in [Sensoy *et al.*, 2018], the loss function is derived as:

$$\mathcal{L}(r,i) = \sum_{k=1}^K y_{rik}\left(\psi\left(\sum_{k=1}^K \alpha_{rik}\right) - \psi\left(\alpha_{rik}\right)\right), \tag{10}$$

where $\psi(z) \simeq \ln z - \frac{1}{2z}$ is the digamma function.

To reduce the covariance evidence and probability, the Kullback–Leibler (KL) divergence [Kullback and Leibler, 1951] is used to regularize the learning process, i.e.,

$$\hat{\boldsymbol{\alpha}}_{ri} = \boldsymbol{y}_{ri} + (1 - \boldsymbol{y}_{ri}) \odot \boldsymbol{\alpha}_{ri}, \tag{11}$$

$$D_{KL}(r,i) = KL(\mathrm{Dir}(\boldsymbol{p};\hat{\boldsymbol{\alpha}}_{ri})||\mathrm{Dir}(\boldsymbol{p};\mathbf{1})). \tag{12}$$

The final loss function is:

$$\mathcal{L}_{\text{overall}}(r,i) = \mathcal{L}(r,i) + \lambda D_{KL}(r,i), \tag{13}$$

where λ is a trade-off parameter.

4.4 Instance Selection

After training, our end-to-end model can infer the belief and uncertainty mass of possible annotations of instances before worker labeling. In short, the importance module finds instances with maximum uncertainty. However, only considering uncertainty will decrease the diversity of instances. The coverage module leverages belief to cover more instances in the overall workers' tasks.

Importance Module. Denote the instance candidates set as $X = \{\boldsymbol{x}_i\}_{i=1}^{N}$. For worker r, in each step t, all instances are inferred to generate $\boldsymbol{\alpha}_{ri}$. The uncertainty mass is:

$$u_{ri} = \frac{K}{S_{ri}} = \frac{K}{\sum_{k=1}^{K} \alpha_{rik}}, \tag{14}$$

where K is the number of classes.

The top-L most uncertain instances are selected as candidates for the coverage module, i.e., $X_r^{(t)} = \mathrm{Top}(X, u_{ri}, L)$.

With the mild assumptions that (1) annotations are i.i.d. variables, (2) ground-truth $\boldsymbol{f}^{\natural}$ exists and is bounded with the hypothesis set, (3) the near-class specialist exists, and (4) the near-anchor point exists, we have the following theorem.

Theorem 1. *[Ibrahim and Fu, 2021] The optimal solution of deep learning from crowds exists, and the distance between this solution $\boldsymbol{f}$ and ground-truth $\boldsymbol{f}^{\natural}$ is bounded.*

However, adaptive training may violate the fourth assumption. For class k, a near-anchor point matching ground-truth predictor $\boldsymbol{f}^{\natural}$ may not exist. Therefore, we propose the coverage module to alleviate it.

Coverage Module. After obtaining candidates $X_r^{(t)}$, we exploit the global belief information to extend the coverage. The global belief for x_i is computed by $\boldsymbol{\alpha}_i = \boldsymbol{f}^{(t)}(\boldsymbol{x}_i)$ instead of $\boldsymbol{\alpha}_{ri}$. Then, the belief for class k is:

$$b_{ik} = \frac{\alpha_{ik}}{S_i} = \frac{\alpha_{ik}}{\sum_{k=1}^{K} \alpha_{ik}}. \tag{15}$$

This coverage module bridges workers' selected instances. From step 1 to $t-1$, all labeled instances is saved as $X^{(1:t-1)}$ with size N', and after selecting the new instance $\boldsymbol{x}_r^{(t)}$, the union is $\{\boldsymbol{x}_r^{(t)}\} \cup X^{(1:t-1)}$. The accumulation of belief mass is $\boldsymbol{b}' = \sum_{i=1}^{N'} \boldsymbol{b}_i$.

To improve the class-level coverage, the instance is selected to minimize the variance of $\boldsymbol{b}^{(t)}$, i.e.,

$$\boldsymbol{x}_r^{(t)} = \underset{\boldsymbol{x}_i \in X_r^{(t)}}{\mathrm{argmax}}\, \mathrm{Var}[b_{i1} + b'_1, b_{i2} + b'_2, \cdots, b_{iK} + b'_K]. \tag{16}$$

The pseudo-code of the overall approach is shown in Algorithm 1. It is worth mentioning that after each step, all collected annotations are trained in E epochs to avoid underfitting. The collection procedure is asynchronous and parallel in the real-world scenario. In our experiments, this sample selection is implemented as training on the whole dataset with a mask, and the mask is updated after data collection.

Different from the existing evidential learning and active learning methods, some important features of the proposed method are discussed in the following perspectives.

Algorithm 1 Pseudo-code of AdaCrowd

Input: Instance pool X.

Output: Target classifiers $\boldsymbol{f}^{(T)}$, and the workers' transition matrices $\{\boldsymbol{A}_r\}_{r=1}^{R}$

```
Initialize classifiers $\boldsymbol{f}^{(0)}$, the workers' transition matrices $\{\boldsymbol{A}_r\}_{i=1}^{R}$ with identity weights.
for $t$ = 1, ..., $T$ do
    for parallel worker $r$ do
        Inference all instances with the evidential model.
        Compute the uncertainty by Eq. (14).
        Compute the accumulated belief by Eq. (15).
        Select instance by Eq. (16).
        Obtain the annotation $\boldsymbol{y}_{ri}$.
    end for
    for $E$ epochs do
        Update parameters of classifier and worker matrices by Eq. (13).
    end for
end for
```

1. By utilizing the implicit uncertainty in data between multiple workers and instances, AdaCrowd circumvents golden labels for explicitly testing workers, instead estimating worker characteristics in the labeling process.
2. Compared with coreset mining, such as learning with redundant and noisy data, AdaCrowd is deployed throughout the whole crowdsourcing method rather than only on clean data after label collection.
3. Compared with evidential deep learning, the goal of AdaCrowd is to measure the information of annotation instead of out-of-distribution detection.

4.5 Theoretical Analysis

The convergence analysis is provided here. The distance between parameter $\boldsymbol{f}^{(t)}, \boldsymbol{A}_r^{(t)}$ in step t and truth $\boldsymbol{f}^{\natural}, \boldsymbol{A}_r^{\natural}$ is bounded. In crowdsourcing learning, the empirical risk minimization (ERM) is directly about $\boldsymbol{A}_r\boldsymbol{f}$, therefore the bound of $E[\|\boldsymbol{f}^{(t)}\boldsymbol{A}_r^{(t)} - \boldsymbol{f}^{\natural}\boldsymbol{A}_r^{\natural}\|]$ is trivial as common stochastic gradient descent (SGD) convergence analysis. The key challenge is analyze $E[\|\boldsymbol{f}^{(t)} - \boldsymbol{f}^{\natural}\|], E[\|\boldsymbol{A}_r^{(t)} - \boldsymbol{A}_r^{\natural}\|]$. Here we give the convergence analysis of $E[\|\boldsymbol{A}_r^{(t)} - \boldsymbol{A}_r^{\natural}\|]$. The adaptive learning is assumed to select a representative subset, known as the coreset of the whole dataset [Mirzasoleiman *et al.*, 2020]. With this coreset assumption, the theorem is as follows.

Theorem 2 (Expected Estimation Error Bound)**.** *Assume that the loss function $\mathcal{L}$, i.e., cross-entropy in classification, is $\mu-$strongly convex, and D is the selected annotation subset with size n that approximates the gradient of full annotations by the error at most ϵ. Then with learning rate η, it holds:*

$$E[\|\boldsymbol{A}_r^{(t+1)} - \boldsymbol{A}_r^{\natural}\|^2] \leq (1-\mu\eta)^{t+1} E[\|\boldsymbol{A}_r^{(0)} - \boldsymbol{A}_r^{\natural}\|^2] + \frac{2\epsilon H}{\mu^2} + \frac{n^2\eta C^2}{\mu}, \tag{17}$$

where C is an upper-bound on the norm of the gradient of the loss function, H is a constant value determined by the initial state, and $d_0 = E[\|\boldsymbol{A}_r^{(0)} - \boldsymbol{A}_r^{\natural}\|^2]$ is the initial distance to the

Dataset	LabelMe	CIFAR-10H-Top100
# workers	59	100
# instances	1,000	8,621
# annotations	2,550	20,000
avg accuracy	69.20%	77.50%

Table 2: Statistics of the Datasets.

optimum $\boldsymbol{A}_r^\natural$. Moreover, if learning rate is $\eta = \frac{1}{\mu}$, it gives:

$$E[\|\boldsymbol{A}_r^{(t+1)} - \boldsymbol{A}_r^\natural\|^2] \leq \frac{2\epsilon H + n^2C^2}{\mu^2}. \tag{18}$$

The proof is given in the supplementary material.

5 Experiments

In this section, we conduct experiments to answer the following research questions:

- RQ1: Can AdaCrowd perform better with existing crowdsourcing models than random selection with fewer annotations?
- RQ2: How does AdaCrowd perform in choosing important data while keeping instance coverage compared to baseline methods?
- RQ3: How does AdaCrowd improve accuracy in fixed training steps, i.e., saving crowdsourcing cost, with the adaptive instance selection reasonably?

5.1 Dataset Description and Analysis

The experiments are conducted on crowdsourcing datasets of image classification: LabelMe and CIFAR-10H, which can be found: LabelMe[2], CIFAR-10H annotation[3], image[4].

LabelMe [Russell *et al.*, 2008; Rodrigues and Pereira, 2018] is an open-source dataset collected from Amazon Mechanical Turk. There are 59 workers, 1,000 instances, and 8 classes: highway, inside city, building, street, forest, coast, mountain, and open country.

CIFAR-10H [Battleday *et al.*, 2020] is an subset of well-known CIFAR-10 dataset. There are 2,571 workers, 10,000 instances, and 10 different classes in the raw version of CIFAR-10H: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. Every worker has labeled 200 images. To keep things practical, our experiments select the top 100 lowest-accuracy workers. After that, there are 100 workers and 8,621 instances in the CIFAR-10H.

The statistics for the datasets are shown in Table 2. The correct rate of workers is counted as shown in Figure 4. According to Table 2 and Figure 4, we find that the datasets have the following properties:

- Accuracy: The average accuracy of workers in CIFAR-10H 95% is much higher than LabelMe 69.20%. In CIFAR-10H, only the top 100 workers with the lowest accuracy are chosen in experiments. In Figure 4, the top-100 workers are the blue part. After that, the average accuracy is 77.50%.
- Annotate Times: In LabelMe, all instances were annotated no more than three times, while the times are inconsistent. In CIFAR-10H, all instances were annotated about 50 times. Therefore, in experiments, we set the number of steps T in LabelMe varying and in CIFAR-10H fixed across workers.

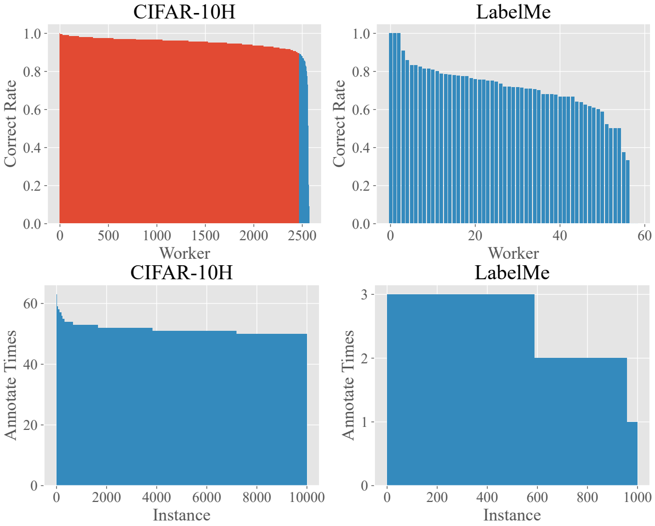

Figure 4: Data analysis of Datasets CIFAR-10H and LabelMe. Top: Correct Rate of Workers, Bottom: Annotate Time of Instances. The red part is discarded, and the blue part is adopted.

The above analysis shows that crowdsourcing usually assigns the same instance to multiple workers, and the workers give correct annotations in most cases. As a result, our proposed approach is suitable for the cost-saving effort in crowdsourcing.

5.2 Experimental Setup

Model Setting. Our models are implemented with the PyTorch library, and the codes are released on our repository[5]. Following the previous work, the pre-trained VGG-16 network is used as the backbone of the classifier for the LabelMe and the CIFAR-10H dataset. Similarly, we follow the arrangement in the previous work that splits and saves the training set, validation set, and test set. In the LabelMe dataset, the augmented training set contains 10,000 images, the validation set contains 500 images, and the testing set contains 1,188 images. In the CIFAR-10H dataset, the training and validation sets contain 10,000 images from the original CIFAR-10H test set, and 2,000 unseen images from the CIFAR-10 dataset are sampled for evaluation.

Hyperparameter Setting. The trade-off parameter λ is increase with training step: $\lambda(t) = \min(1, t/T_w)$. The learning rate λ is selected from [0.0005, 0.001, 0.005, 0.01]. The weight decay is selected from [0, 0.003, 0.009, 0.01]. The

[2] http://labelme.csail.mit.edu/

[3] https://github.com/jcpeterson/cifar-10h

[4] https://www.cs.toronto.edu/~kriz/cifar.html

[5] https://github.com/MISE-MUST/AdaCrowd

Method	CrowdLayer	TraceReg	CoNAL	GeoCrowdNet
Random	83.22 ± 0.20	85.34 ± 0.32	81.70 ± 0.31	83.23 ± 0.34
Softmax	73.45 ± 0.64	84.83 ± 0.25	77.54 ± 0.03	79.09 ± 5.71
MaxInf	72.63 ± 2.61	76.62 ± 0.54	78.61 ± 1.31	80.11 ± 0.21
MaxGrad	71.45 ± 0.63	76.07 ± 0.83	78.01 ± 0.30	79.81 ± 0.51
AdaCrowd	**86.08 ± 0.16***	**85.94 ± 0.17***	**81.82 ± 1.80**	83.84 ± 0.86
AdaCrowd w/o imp	83.92 ± 0.30	81.65 ± 0.21	79.96 ± 0.72	80.47 ± 1.35
AdaCrowd w/o cov	85.99 ± 0.48	81.26 ± 0.21	80.31 ± 0.25	**84.32 ± 0.16**
Full	86.51 ± 0.11	85.95 ± 0.26	82.22 ± 0.26	84.68 ± 0.30

Table 3: Performance of Accuracy on LabelMe dataset (*: $p < 0.05$)

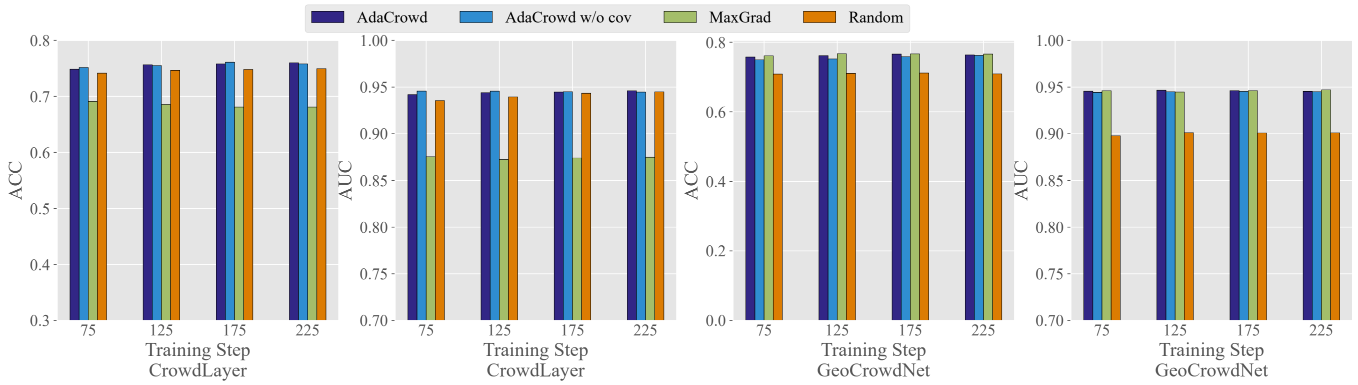

Figure 5: Performance of Accuracy and AUC on CIFAR-10H dataset.

annealing step T_w is selected from [5, 10, 15, 20]. The epoch in each step E is selected from the range $[1, 5]$. According to the validation set, λ is set to 0.001, the weight decay is set to 0, E is set to 2, and T_w is set to 5.

5.3 Performance Comparison (RQ1)

To evaluate the performance of the proposed AdaCrowd, four well-known crowdsourcing approaches are chosen as baselines: CrowdLayer [Rodrigues and Pereira, 2018], TraceReg [Tanno *et al.*, 2019], CoNAL [Chu *et al.*, 2021], and GeoCrowdNet [Ibrahim *et al.*, 2023]. And the instance selection methods Random, Softmax, MaxGrad, and MaxInf are chosen for comparison. The softmax means selecting an instance with the highest logits across classes but minimal probability across instances. MaxGrad and MaxInf select instances by the expected change of parameters, where MaxGrad leverages the gradient, and MaxInf leverages the influence function. Meanwhile, we ablate the importance and coverage modules to show their improvement. The test accuracy and Area Under Curve (AUC) are used as metrics.

The results in LabelMe and CIFAR-10H datasets are shown in Table 3 and Figure 5. Further, we observe that:

(1) The full-data training usually shows the highest performance across all methods. The AdaCrowd method performs better than other adaptive methods. The MaxGrad slightly fails the MaxInf because the influence function performs better in estimating the change of parameters.

(2) Softmax underperforms relative to AdaCrowd; the lower performance could stem from overconfidence in output probabilities, which can be misleading when dealing with noisy crowdsourced labels.

(3) AdaCrowd outperforms AdaCrowd w/o imp and AdaCrowd w/o cov in most cases, while it performs close to the top for GeoCrowdNet, slightly surpassed by AdaCrowd w/o cov, suggesting the improvements in importance and coverage modules, while impact varies by crowdsourcing method.

(4) The variability, indicated by the standard deviation, is generally low, showing that these adaptive training strategies are consistent across crowdsourcing methods.

5.4 Coverage in Instance Selection (RQ2)

To gain a better insight into instance selection, we take a close look at the breadth of instances selected by the algorithm, and the instance coverages of different compared methods are measured. The intuition here is that under constant annotations from constant steps, more instances from more classes should be included for training models. In other words, if many annotations about the same instance are collected, the generalization of the model will degrade.

For each step, the number of instances of each class is first recorded. Then, the statistical value of all classes about coverage can be obtained. We focus on the mean value, the lower quartile, and the upper quartile of the first 50 training steps, which are shown as the curve and the shadow in Figure 6.

As portrayed in Figure 6, we can find that:

(1) The curve of the MaxGrad is usually below, which shows that the MaxGrad method tends to select the replicated

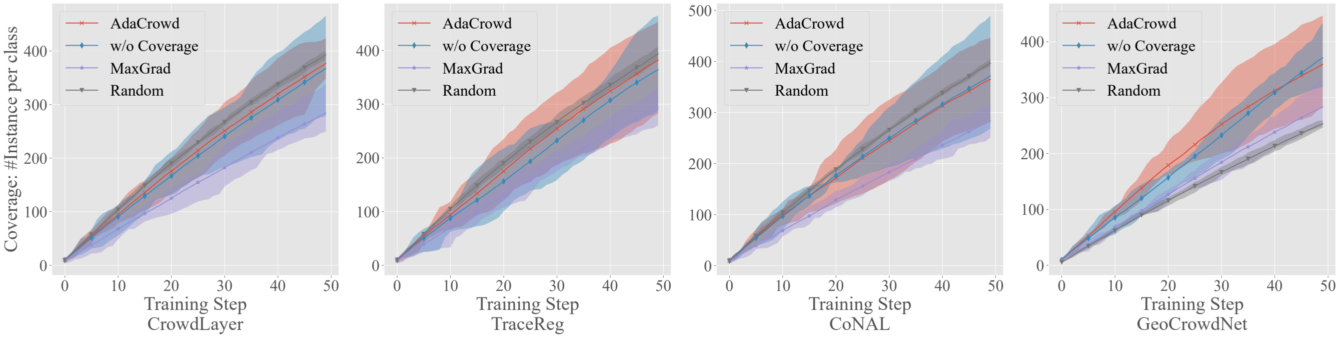

Figure 6: Instance Coverage Comparison on CIFAR-10H dataset.

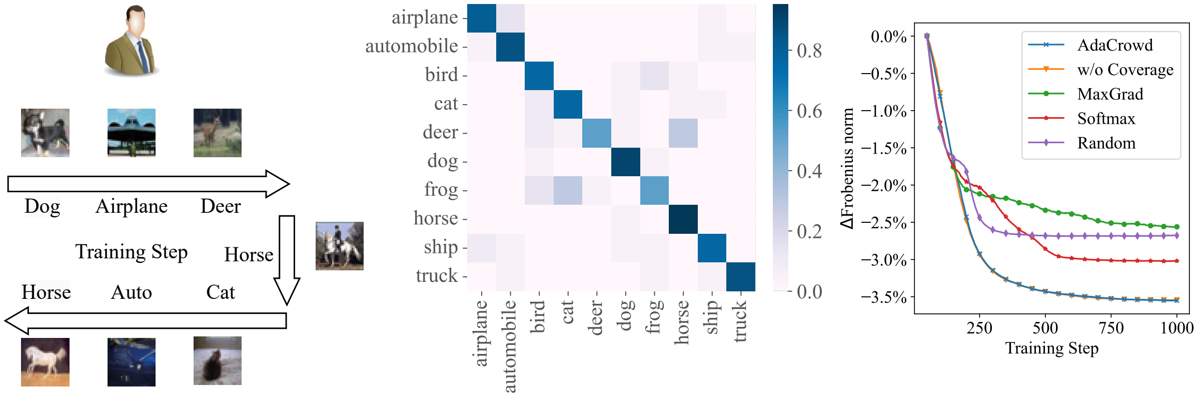

Figure 7: Case study of a worker in CIFAR-10H: selected instances, transition matrix, and training process.

instances compared with other methods.

(2) The shadow area of the Random is the smallest, which shows that the distribution is very concentrated, and the Random method uniformly samples instances from all classes.

(3) Compared with the above methods, the proposed AdaCrowd can select different instances with different emphasis between classes when sampling, which guarantees the diversity of chosen annotations and instances.

(4) The ablation method performs similarly to the original method, but the mean value is slightly lower, which shows that the coverage module can help to reduce replication.

5.5 Case Study (RQ3)

The No.74 worker in the CIFAR-10H is chosen as a case, and the result is shown in Figure 7. After following the training process, we have the following observations.

The left frame is the training steps from 50 to 56. We find that these selected instances are difficult to label correctly. Besides, the annotations are closely related to the confusion matrix, as shown in the middle frame.

One of the main goals of AdaCrowd is to estimate the transition matrix of workers with fewer annotations. To measure the performance, the transition matrices are softmax-normalized. For $\boldsymbol{A}_r$, the softmax-normalized matrix is: $[\text{Softmax}(\boldsymbol{A}_r)]_{ij} = \frac{\exp[\boldsymbol{A}_r]_{ij}}{\sum_{k=1}^{K}\exp[\boldsymbol{A}_r]_{ik}}$. Let us denote the ground truth by $\boldsymbol{A}_r^*$, and the normalized transition matrix by $\boldsymbol{A}_r$, and the error matrix by $\boldsymbol{E}_r = \text{Softmax}(\boldsymbol{A}_r) - \text{Softmax}(\boldsymbol{A}_r^*)$. Then, the Frobenius norm of the error matrix is considered as the measurement. For this $K \times K$ matrix $\boldsymbol{E}_r$, the norm is $||\boldsymbol{E}_r||_{\text{F}} = \sqrt{\sum_{i=1}^{K}\sum_{j=1}^{K}[\boldsymbol{E}_r]_{ij}^2}$.

The reduction of the Frobenius norm starting from step 50 is recorded in training steps as shown in the right frame. According to the curves in the figure, we can find that the proposed AdaCrowd performs better on this metric compared with other methods, especially in the long-term training steps. Although some compared methods, such as MaxGrad, are better in the first few steps, the convergence value is far worse than that of the proposed method.

6 Conclusion

In this research, we propose AdaCrowd, an adaptive learning method to efficiently utilize crowdsourced datasets by leveraging worker characteristics early in the data collection process. AdaCrowd optimizes the use of workers' abilities by dynamically assessing the informativeness of instances based on prediction uncertainties. This method not only enhances the quality of the training data but also reduces the number of necessary annotations without compromising model performance. Our experiments on datasets such as LabelMe and CIFAR-10H validate the effectiveness of AdaCrowd, demonstrating its potential as a cost-effective solution for training models in the crowdsourcing approach.

Acknowledgements

This work is supported by the Science and Technology Development Fund (FDCT) under Grant number 0029/2023/RIA1.

References

[Basu Roy *et al.*, 2015] Senjuti Basu Roy, Ioanna Lykourentzou, Saravanan Thirumuruganathan, Sihem Amer-Yahia, and Gautam Das. Task assignment optimization in knowledge-intensive crowdsourcing. *The VLDB Journal*, 24(4):467–491, August 2015.

[Battleday *et al.*, 2020] Ruairidh M Battleday, Joshua C Peterson, and Thomas L Griffiths. Capturing human categorization of natural images by combining deep networks and cognitive models. *Nature Communications*, 11(1):5418, October 2020.

[Brenzel and Settles, 2017] Jeffrey Brenzel and Burr Settles. The duolingo english test—design, validity, and value. *DET Whitepaper (Short)*, pages 1–3, September 2017.

[Chen *et al.*, 2022] Ziqi Chen, Liangxiao Jiang, and Chaoqun Li. Label augmented and weighted majority voting for crowdsourcing. *Information Sciences*, 606:397–409, August 2022.

[Chu *et al.*, 2021] Zhendong Chu, Jing Ma, and Hongning Wang. Learning from crowds by modeling common confusions. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pages 5832–5840, Virtual Conference, May 2021.

[Dawid and Skene, 1979] Alexander Philip Dawid and Allan M Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1):20–28, March 1979.

[Fang *et al.*, 2018] Yili Fang, Hailong Sun, Pengpeng Chen, and Jinpeng Huai. On the cost complexity of crowdsourcing. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 1531–1537, Stockholm, Sweden, July 2018.

[Ghosh and Lan, 2021] Aritra Ghosh and Andrew Lan. Bobcat: Bilevel optimization-based computerized adaptive testing. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, pages 2410–2417, Virtual Conference, August 2021.

[Han *et al.*, 2025] Tao Han, Huaixuan Shi, Xinyi Ding, Xi-Ao Ma, Huamao Gu, and Yili Fang. Mixture of experts based multi-task supervise learning from crowds. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence*, pages 14256–14264, Philadelphia, PA, United States, April 2025.

[Haussler and Warmuth, 1995] David Haussler and Manfred Warmuth. *The Probably Approximately Correct (PAC) and Other Learning Models*. Springer, Boston, MA, United States, 1995.

[Ho *et al.*, 2013] Chien-Ju Ho, Shahin Jabbari, and Jennifer Wortman Vaughan. Adaptive task assignment for crowdsourced classification. In *Proceedings of the 30th International Conference on Machine Learning*, pages 534–542, Atlanta, GA, United States, June 2013.

[Ibrahim and Fu, 2021] Shahana Ibrahim and Xiao Fu. Crowdsourcing via annotator co-occurrence imputation and provable symmetric nonnegative matrix factorization. In *Proceedings of the 38th International Conference on Machine Learning*, pages 4544–4554, Virtual Conference, July 2021.

[Ibrahim *et al.*, 2023] Shahana Ibrahim, Tri Nguyen, and Xiao Fu. Deep learning from crowdsourced labels: Coupled cross-entropy minimization, identifiability, and regularization. In *Proceedings of the 11th International Conference on Learning Representations*, pages 1–39, Kigali, Rwanda, May 2023.

[Kullback and Leibler, 1951] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, March 1951.

[LeCun *et al.*, 2015] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015.

[Miao *et al.*, 2023] Xiaoye Miao, Huanhuan Peng, Yunjun Gao, Zongfu Zhang, and Jianwei Yin. On dynamically pricing crowdsourcing tasks. *ACM Transactions on Knowledge Discovery from Data*, 17(2):1–27, February 2023.

[Mirzasoleiman *et al.*, 2020] Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models. In *Proceedings of the 37th International Conference on Machine Learning*, pages 6950–6960, Virtual Conference, July 2020.

[Nguyen *et al.*, 2024] Tri Nguyen, Shahana Ibrahim, and Xiao Fu. Noisy label learning with instance-dependent outliers: Identifiability via crowd wisdom. In *Proceedings of the 38th Annual Conference on Neural Information Processing Systems*, pages 97261–97298, Vancouver, BC, Canada, December 2024.

[Rodrigues and Pereira, 2018] Filipe Rodrigues and Francisco Câmara Pereira. Deep learning from crowds. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence and 30th Innovative Applications of Artificial Intelligence Conference and 8th AAAI Symposium on Educational Advances in Artificial Intelligence*, page 1611–1618, New Orleans, LA, United States, February 2018.

[Rudner, 2010] Lawrence M. Rudner. Demystifying the gmat: Computer adaptive testing. *Graduate Management Admission Council: Deans Digest*, page 1, June 2010.

[Russell *et al.*, 2008] Bryan C. Russell, Antonio Torralba, Kevin P. Murphy, and William T. Freeman. Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1):157–173, May 2008.

[Sensoy *et al.*, 2018] Murat Sensoy, Lance Kaplan, and Melih Kandemir. Evidential deep learning to quantify classification uncertainty. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 3183–3193, Montréal, QC, Canada, December 2018.

[Tanno *et al.*, 2019] Ryutaro Tanno, Ardavan Saeedi, Swami Sankaranarayanan, Daniel C. Alexander, and Nathan Silberman. Learning from noisy labels by regularized estimation of annotator confusion. In *Proceedings of the 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11236–11245, Long Beach, CA, United States, June 2019.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6000–6010, Long Beach, CA, United States, December 2017.

[Wang and Zhou, 2016] Lu Wang and Zhi-Hua Zhou. Cost-saving effect of crowdsourcing learning. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, pages 2111–2117, New York, NY, United States, July 2016.

[Yan *et al.*, 2011] Yan Yan, Glenn M Fung, Rómer Rosales, and Jennifer G Dy. Active learning from crowds. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 1161–1168, Bellevue, WA, United States, June 2011.

[Yang *et al.*, 2018] Jingru Yang, Ju Fan, Zhewei Wei, Guoliang Li, Tongyu Liu, and Xiaoyong Du. Cost-effective data annotation using game-based crowdsourcing. *Proceedings of the VLDB Endowment*, 12(1):57–70, September 2018.

[Zhang *et al.*, 2024] Hansong Zhang, Shikun Li, Dan Zeng, Chenggang Yan, and Shiming Ge. Coupled confusion correction: Learning from crowds with sparse annotations. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence and 36th Conference on Innovative Applications of Artificial Intelligence and 14th Symposium on Educational Advances in Artificial Intelligence*, pages 16732–16740, Vancouver, BC, Canada, March 2024.

[Zhao *et al.*, 2014] Liyue Zhao, Yu Zhang, and Gita Sukthankar. An active learning approach for jointly estimating worker performance and annotation reliability with crowdsourced data. *arXiv Preprint arXiv:1401.3836*, pages 1–18, January 2014.

[Zhong *et al.*, 2015] Jinhong Zhong, Ke Tang, and Zhi-Hua Zhou. Active learning from crowds with unsure option. In *Proceedings of the 24th International Conference on Artificial Intelligence*, pages 1061–1067, Buenos Aires, Argentina, July 2015.

[Zhuang *et al.*, 2022] Yan Zhuang, Qi Liu, Zhenya Huang, Zhi Li, Shuanghong Shen, and Haiping Ma. Fully adaptive framework: Neural computerized adaptive testing for online education. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, pages 4734–4742, Vancouver, BC, Canada, February 2022.