Utilizing supervised models to infer consensus labels and their quality from data with multiple annotators

Hui Wen Goh
huiwen@cleanlab.ai
Cleanlab

Ulyana Tkachenko
ulyana@cleanlab.ai
Cleanlab

Jonas Mueller
jonas@cleanlab.ai
Cleanlab

Abstract

Real-world data for classification is often labeled by multiple annotators. For analyzing such data, we introduce CROWDLAB, a straightforward approach to estimate: (1) A consensus label for each example that aggregates the individual annotations (more accurately than aggregation via majority-vote or other algorithms used in crowdsourcing); (2) A confidence score for how likely each consensus label is correct (via well-calibrated estimates that account for the number of annotations for each example and their agreement, prediction-confidence from a trained classifier, and trustworthiness of each annotator vs. the classifier); (3) A rating for each annotator quantifying the overall correctness of their labels. While many algorithms have been proposed to estimate related quantities in crowdsourcing, these often rely on sophisticated generative models with iterative inference schemes, whereas CROWDLAB is based on simple weighted ensembling. Many algorithms also rely solely on annotator statistics, ignoring the features of the examples from which the annotations derive. CROWDLAB in contrast utilizes any classifier model trained on these features, which can generalize between examples with similar features. In evaluations on real-world multi-annotator image data, our proposed method provides superior estimates for (1)-(3) than many alternative algorithms.

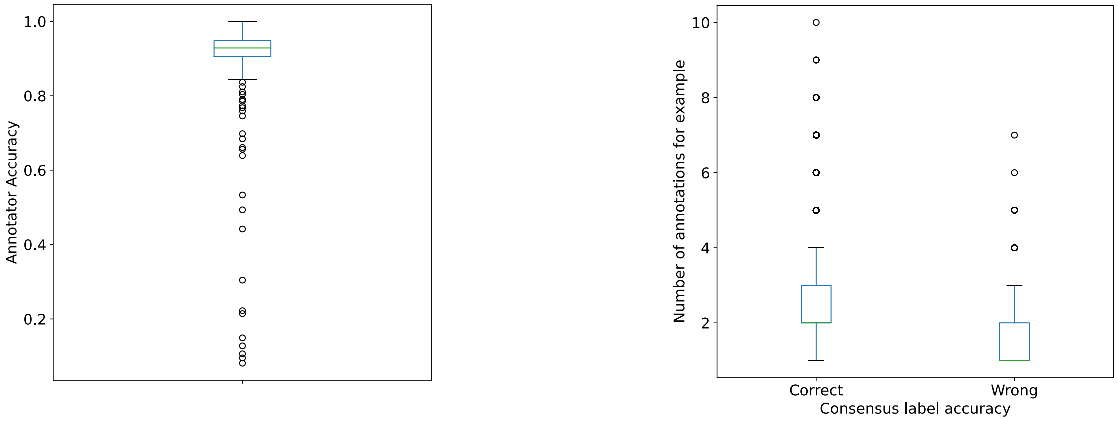
1 Introduction

Training data for multiclass classification are often labeled by multiple annotators, with some redundancy between annotators to ensure high-quality labels. Such settings have been studied in crowdsourcing research [Monarch, 2021b, Paun et al., 2018], where it is often assumed that *many* annotators have labeled each example [Carpenter, 2008, Khetan et al., 2018]. This is often prohibitively expensive, so we here consider general settings where each example in the dataset is merely labeled by at least *one* annotator, and each annotator labels many examples (but still only a subset of the dataset). Each annotation corresponds to the selection of one class $y \in \{1, \dots, K\}$ which the annotator believes to be most appropriate for this example.

While certain ML classification models can be trained in a special manner to account for the multiple labels per example [Nguyen et al., 2014, Peterson et al., 2019], a more straightforward approach commonly utilized is to aggregate the labels for each example into a single *consensus label*, e.g. via majority-vote or crowdsourcing algorithms like Dawid-Skene [Dawid and Skene, 1979]. Any classifier can then be trained on these consensus labels via off-the-shelf code. Here we propose a method[1] that leverages any already-trained classifier to: (1) establish accurate consensus labels, (2) estimate their quality, and (3) estimate the quality of each annotator [Monarch, 2021c]. The latter two aims help us determine which data is least trustworthy (and should perhaps be additionally verified in subsequent rounds of annotation [Bernhardt et al., 2022]). **CROWDLAB** (Classifier Refinement Of croWDsourced LABels) is based on a straightforward weighted ensemble of the classifier predictions and individual annotations, with weights assigned according to the (estimated) trusthworthiness of each component. CROWDLAB is easy to implement/understand, computationally efficient (non-iterative), and extremely flexible. It works with any classifier and training procedure, as well as any classification dataset (including those containing examples only labeled by one annotator).

Motivations. Figure 1 illustrates how many real-world multi-annotator datasets look, showing a large disparity in annotator quality as well as many examples whose consensus label will be incorrect if we rely on majority vote (which is often done in practice). Unsurprisingly, consensus labels are more likely to be incorrect for those examples with fewer annotations. Thus an effective method to estimate consen-

[1] Code: https://github.com/cleanlab/cleanlab
Reproduce our results: https://github.com/cleanlab/multiannotator-benchmarks



(a) Overall accuracy of each annotator

(b) Accuracy of each example's consensus label

Figure 1: Statistics of our *Hardest* dataset, where accuracy is always measured against underlying ground-truth labels. **(a)** Distribution over annotators showing the overall accuracy of each annotator's chosen labels. **(b)** Distribution over examples showing the number of annotations per example, grouped by whether the majority-vote consensus label is correct or not.

sus label quality should properly account for the number of annotations an example has received as well as the quality of the annotators who selected these labels. Many of the examples whose consensus label is wrong merely have a single annotation, which provides little information, and thus leveraging a trained classifier can help us better generalize to such examples to estimate their labels' quality (especially if the data contain other examples with similar feature values). But when utilizing a classifier, we also wish to account for the accuracy and confidence of its estimates. CROWDLAB is a straightforward approach to appropriately account for all of these factors.

Notation. Consider a dataset sampled from (feature, class label) pairs (X, Y) that is comprised of: n examples, m annotators, and K classes. Here is notation we use throughout:

- $[n] = \{1, 2, ..., n\}$ indexes examples in the dataset and X_i denotes the features of the ith example.
- Each example i belongs to one class, i.e. $Y_i \in [K] := \{1, \ldots, K\}$. This true class is unknown to us.
- $\mathcal{A}_j$ is the jth annotator for $j \in [m] := \{1, 2, ..., m\}$.
- $Y_{ij} \in [K]$ denotes the class annotator j chose for example i, with $Y_{ij} = \varnothing$ if $\mathcal{A}_j$ did not label this particular example. Each example receives at most most m annotations, with most examples receiving far fewer annotations under a reasonable data labeling budget.
- $\widehat{Y}_i$ denotes the consensus label for example i, representing our best estimate of its true class Y_i.
- $\mathcal{I}_j$ denotes the subset of examples labeled by annotator j, $\mathcal{I}_j := \{i \in [n] : Y_{ij} \neq \varnothing\}$. We assume each annotator has labeled multiple examples, i.e. $|\mathcal{I}_j| > 1$.
- $\mathcal{J}_i$ denotes the subset of annotators that labeled example i, $\mathcal{J}_i := \{j \in [m] : Y_{ij} \neq \varnothing\}$. To save labeling costs, some examples may only be labeled by a single annotator.
- $\mathcal{I}_+ := \{i \in [n] : |\mathcal{J}_i| > 1\}$ denotes the subset of examples labeled by more than one annotator.
- $q_i \in [0, 1]$ denotes a *consensus quality score* for consensus label $\widehat{Y}_i$, with values near 0 indicating consensus labels we are less confident are correct.
- $a_j \in [0, 1]$ denotes an overall *annotator quality score* for annotator j, with values near 0 indicating annotators we are less confident will choose a correct label for any given example.
- $\widehat{p}_{\mathcal{M}}(Y_i \mid X_i) \in \mathbb{R}^K$ denotes the predicted probability vector given by a (trained classifier) model that a particular example with features X_i belongs to each class k. $Y_{i,\mathcal{M}} \in [K]$ denotes the model predicted label for example i, i.e. $Y_{i,\mathcal{M}} = \arg\max_k \widehat{p}_{\mathcal{M}}(Y_i = k \mid X_i)$.
- $L(Y, p) \in [0, 1]$ denotes a *label quality score* [Kuan and Mueller, 2022] which estimates our confidence that a particular label $Y \in [K]$ is correct for example X, given vector $p \in \mathbb{R}^K$ estimating the likelihood that X belongs to each class. In this paper, we use *self-confidence* as the label quality score, $L(Y, p) = p(Y)$, representing the estimated probability that the example belongs to its labeled class. Kuan and Mueller [2022], Northcutt et al. [2021b] found this to be an empirically effective score for flagging label errors (in singly-labeled data) based on classifier predictions: $p(Y) \approx \widehat{p}_{\mathcal{M}}(Y \mid X)$.

We also employ the following standard mathematical notation: $|\mathcal{J}|$ denotes the cardinality of set $\mathcal{J}$, $\mathbb{1}(\cdot)$ denotes the indicator function which evaluates to 1 if its condition is True and 0 otherwise.

2 Methods

We assume some classifier model $\mathcal{M}$ has been trained to predict the given labels based on feature values. CROWDLAB can be used with any type of classifier $\mathcal{M}$ (and training procedure) as long as it is capable of outputting probabilistic predictions $\widehat{p}_{\mathcal{M}}(Y \mid X)$. To avoid overfit predictions, we fit $\mathcal{M}$ via cross-validation, which enables us to produce *held-out* predictions $\widehat{p}_{\mathcal{M}}(Y_i \mid X_i)$ for each example in the dataset (from a copy of $\mathcal{M}$ which never saw X_i during training). In our subsequent experiments, we train $\mathcal{M}$ on consensus labels derived via majority vote, but one could train the classifier on any other set of improved consensus labels or even on the individual labels from each annotator (simply duplicating multiply-annotated examples in the training set). The performance of all methods considered here that leverage $\mathcal{M}$ will accordingly benefit from improvements in the classifier's predictive accuracy, but CROWDLAB is the only method that aims to explicitly account for shortcomings of the classifier's predictions (which are inevitable due to estimation error).

2.1 Consensus Quality Scoring Methods

We start by outlining various methods to estimate our confidence that a given consensus label for each example is correct. These quality estimates $q_i \in [0, 1]$ may be applied to any given label no matter which method was used to establish consensus. Once we can estimate the quality of any one label for each example, our consensus label established under each method is simply chosen as the class associated with the highest consensus quality score (estimated by this method). This class can be identified efficiently for CROWDLAB.

Agreement [Monarch, 2021b]. The fraction of annotators who agree with consensus label (does not use classifier).

$$q_i = \frac{1}{|\mathcal{J}_i|} \sum_{j \in \mathcal{J}_i} \mathbb{1}(Y_{ij} = \widehat{Y}_i) \tag{1}$$

Label Quality Score [Kuan and Mueller, 2022]. Likelihood of each consensus label estimated by the trained classifier model: $q_i = L(\widehat{Y}_i, \widehat{p}_{\mathcal{M}}(Y_i \mid X_i))$. Used to evaluate labels in standard (single-label) classification, this baseline score ignores information from individual annotators.

CROWDLAB (Classifier Refinement Of croWDsourced LABels). CROWDLAB also employs the same label quality score for each consensus label, but applies it to a different class probability vector which modifies the prediction output by our classifier to account for the individual annotations given for a particular example: $q_i = L(\widehat{Y}_i, \widehat{p}_{\text{CR}}(Y_i \mid X_i, \{Y_{ij}\}))$. We estimate these class probabilities by means of a weighted ensemble aggregation [Fakoor et al., 2021]:

$$\widehat{p}_{\text{CR}}(Y_i \mid X_i, \{Y_{ij}\}) = \frac{w_{\mathcal{M}} \cdot \widehat{p}_{\mathcal{M}}(Y_i \mid X_i) + \sum_{j \in \mathcal{J}_i} w_j \cdot \widehat{p}_{\mathcal{A}_j}(Y_i \mid \{Y_{ij}\})}{w_{\mathcal{M}} + \sum_{j \in \mathcal{J}_i} w_j}$$

where $\widehat{p}_{\mathcal{M}} \in \mathbb{R}^K$ is the probability of each class predicted by our classifier, $\widehat{p}_{\mathcal{A}_j} \in \mathbb{R}^K$ is a similar likelihood vector for each annotator's prediction, and $w_j, w_{\mathcal{M}} \in \mathbb{R}$ are weights to account for the relative trustworthiness of each annotator and our classifier (details further below).

To present the remaining details, we first define a likelihood parameter P set as the average annotator agreement, across examples that have more than one annotation. P estimates the probability that an arbitrary annotator's label will match the majority-vote consensus label for an arbitrary example.

$$P = \frac{1}{|\mathcal{I}_+|} \sum_{i \in \mathcal{I}_+} \frac{1}{|\mathcal{J}_i|} \sum_{j \in \mathcal{J}_i} \mathbb{1}(Y_{ij} = \widehat{Y}_i)$$
$$\text{where } \mathcal{I}_+ := \{i \in [n] : |\mathcal{J}_i| > 1\} \tag{2}$$

We then simply define our annotator predicted probability vector used in (2.1) to be:

$$\widehat{p}_{\mathcal{A}_j}(Y_i = k \mid \{Y_{ij}\}) = \begin{cases} P & \text{when } Y_{ij} = k \\ \frac{1-P}{K-1} & \text{when } Y_{ij} \neq k \end{cases} \tag{3}$$

This simple likelihood is shared across annotators and only involves a single parameter P that is easily estimated from the data. Now let s_j represent annotator j's agreement with other annotators who labeled the same examples.

$$s_j = \frac{\sum_{i \in \mathcal{I}_j} \sum_{\ell \in \mathcal{J}_i, \ell \neq j} \mathbb{1}(Y_{ij} = Y_{i\ell})}{\sum_{i \in \mathcal{I}_j} (|\mathcal{J}_i| - 1)} \tag{4}$$

Let $A_{\mathcal{M}}$ be the (empirical) accuracy of our classifier with respect to the majority-vote consensus labels over the examples with more than one annotation (for which consensus is more trustworthy).

$$A_{\mathcal{M}} = \frac{1}{|\mathcal{I}_+|} \sum_{i \in \mathcal{I}_+} \mathbb{1}(Y_{i,\mathcal{M}} = \widehat{Y}_i) \tag{5}$$

Here $Y_{i,\mathcal{M}} \in [K]$ is the class predicted by our model for X_i. To normalize against a baseline, we calculate the accuracy A_{MLC} of always predicting the most common overall class Y_{MLC}, which is the class labeled the most by the annotators. This accuracy is also calculated on the subset of examples that have more than one annotator, $\mathcal{I}_+$ defined in (2).

$$A_{\text{MLC}} = \frac{1}{|\mathcal{I}_+|} \sum_{i \in \mathcal{I}_+} \mathbb{1}(Y_{\text{MLC}} = \widehat{Y}_i) \tag{6}$$

Based on this majority-class-accuracy baseline, we compute normalized versions of: each annotator's agreement with other annotators and the adjusted accuracy of the model.

$$w_j = 1 - \frac{1 - s_j}{1 - A_{\text{MLC}}} \tag{7}$$

$$w_{\mathcal{M}} = \left(1 - \frac{1 - A_{\mathcal{M}}}{1 - A_{\text{MLC}}}\right) \cdot \sqrt{\frac{1}{n}\sum_i |\mathcal{J}_i|} \tag{8}$$

CROWDLAB uses w_j and $w_{\mathcal{M}}$ to weight our annotators and model in its weighted ensemble.

Dawid-Skene [Dawid and Skene, 1979]. This Bayesian method specifies a generative model of the dataset annotations and employs iterative expectation-maximization (EM) to estimate each annotator's error rates in a class-specific manner. A key item subsequently estimated in this approach is $\widehat{p}_{\text{DS}}(Y_i \mid \{Y_{ij}\})$, the posterior probability vector of the true class Y_i for the ith example, given the dataset annotations $\{Y_{ij}\}$.

Define $\pi^{(j)}_{k,\ell}$ as the probability that annotator j labels an example as class ℓ when the true label of that example is k, i.e. the individual class confusion matrix for each annotator, which is the likelihood function of the Dawid-Skene generative model. The Dawid-Skene posterior distribution for a particular example is computed by taking product of each annotator's likelihood and some prior distribution π_{prior}.

$$\widehat{p}_{\text{DS}}(Y_i \mid \{Y_{ij}\}) \propto \pi_{\text{prior}} \cdot \prod_{j \in \mathcal{J}_i} \pi^{(j)}_{k,Y_{ij}} \tag{9}$$

Our work follows conventional practice taking the prior to be the unconditional empirical distribution of given labels across the dataset. A natural consensus quality score is the label quality score for each consensus label under the Dawid-Skene posterior class probabilities: $q_i = L(\widehat{Y}_i, \widehat{p}_{\text{DS}}(Y_i \mid \{Y_{ij}\}))$.

GLAD (Generative model of Labels, Abilities and Difficulties) [Whitehill et al., 2009]. Specifying a more complex generative model of the dataset annotations than Dawid-Skene, this Bayesian approach also employs iterative expectation-maximization (EM) to additionally estimate estimate α, the expertise of each annotator and β, the difficulty of each example. GLAD's likelihood is based on the following probability that an annotator chooses the same class as the consensus label:

$$p(Y_{ij} = \widehat{Y}_i | \alpha_j, \beta_i) = \frac{1}{1 + e^{-\alpha_j \beta_i}} \tag{10}$$

Like Dawid-Skene, GLAD uses the data likelihood to estimate the posterior probability of the true class Y_i for the ith example: $\widehat{p}_{\text{G}}(Y_i \mid \{Y_{ij}\})$. Here we use the same standard prior as for Dawid-Skene. Again a consensus quality score can naturally be obtained via the label quality score computed with respect to the GLAD posterior class probabilities: $q_i = L(\widehat{Y}_i, \widehat{p}_{\text{G}}(Y_i \mid \{Y_{ij}\}))$.

While many Bayesian annotation models have been proposed [Kara et al., 2015, Hovy et al., 2013, Carpenter, 2008], Dawid-Skene and GLAD are often used in practice [Toloka, Monarch, 2021c] and perform well in empirical benchmarks [Sheshadri and Lease, 2013, Paun et al., 2018, Sinha et al., 2018].

Dawid-Skene with Model [Monarch, 2021a]. Although very popular, the Dawid-Skene and GLAD methods do not utilize a classifier at all, and thus they struggle with sparsely labeled examples for which the only way to produce meaningful estimates is by generalizing over feature values X. A straightforward adaptation of these methods to leverage a classifier is to produce class predictions for each example (predict hard labels rather than probability vectors), and treat these predicted labels as if they were the outputs from an additional annotator [Monarch, 2021a]. Because methods like Dawid-Skene and GLAD automatically adjust for estimated annotator quality, they should theoretically account for the classifier's strengths/weaknesses.

For example, we adapt the Dawid-Skene approach in this fashion by: adding the model's predicted labels as an additional annotator (who has annotated every example), then computing the consensus quality score using the same Dawid-Skene method described above. The resulting posterior is now a function of the example's feature values as well (since classifier predictions depend on X_i).

GLAD with Model [Monarch, 2021a]. We follow the same approach to adapt GLAD to leverage the classifier: First add the model's predicted label for each example as labels from one additional annotator, and then compute the consensus quality score using the GLAD method described above.

Empirical Bayes. While the previous two methods do not account for the classifier's confidence in its individual predictions, we consider an alternative adaptation of Dawid-Skene that does. This method treats the model's prediction as a per-example prior distribution and the annotators' labels as observations to compute $\widehat{p}_{\text{EB}}(Y_i \mid X_i, \{Y_{ij}\})$, the posterior probability of the true class Y_i for the ith example, given the dataset annotations $\{Y_{ij}\}$ and an example-specific prior based on the feature values X_i. The likelihood function for each annotator is defined by the class confusion matrix estimated via the Dawid-Skene algorithm. Using the classifier-derived prior distribution and likelihoods, we can compute an Empirical Bayes posterior in the same way outlined for Dawid-Skene:

$$\widehat{p}_{\text{EB}}(Y_i \mid X_i, \{Y_{ij}\}) \propto \widehat{p}_{\mathcal{M}}(Y_i \mid X_i) \cdot \prod_{j \in \mathcal{J}_i} \pi^{(j)}_{k,Y_{ij}} \tag{11}$$

and compute a consensus quality score in the same manner: $q_i = L(\widehat{Y}_i, \widehat{p}_{\text{EB}}(Y_i \mid X_i, \{Y_{ij}\}))$.

Some have considered iterative variants of this hybrid generative/discriminative approach, in which the classifier is retrained to fit the resulting posterior and the above process is repeated with the new classifier [Raykar et al., 2010, Khetan et al., 2018, Rodrigues and Pereira, 2018, Platanios et al., 2020]. This however requires iterative training of a classifier over many rounds, as well as training the classifier with soft labels rather than a standard classification setting.

Active Label Cleaning [Bernhardt et al., 2022]. Also utilizing a trained classifier, Bernhardt et al. [2022] recently proposed to score multi-annotator consensus quality by subtracting the cross-entropy between classifier predicted probabilities and individual annotations by the entropy of the former.

$$\begin{aligned} q_i = &- \sum_{k=1}^{K} \widehat{p}_{\text{emp}}(Y_i = k \mid \{Y_{ij}\}_{j \in \mathcal{J}_i}) \cdot \log \widehat{p}_{\mathcal{M},i,k} \\ &- \left(- \sum_{k=1}^{K} \widehat{p}_{\mathcal{M},i,k} \cdot \log \widehat{p}_{\mathcal{M},i,k} \right) \end{aligned} \tag{12}$$

Here we abbreviate $\widehat{p}_{\mathcal{M},i,k} := \widehat{p}_{\mathcal{M}}(Y_i = k \mid X_i)$, and $\widehat{p}_{\text{emp}}(Y_i = k \mid \{Y_{ij}\}_{j \in \mathcal{J}_i})$ is the overall empirical distribution of class labels amongst the annotations for a particular example. Like CROWDLAB, this approach accounts for classifier confidence and all individual annotations, but it lacks CROWDLAB's ability to adjust for how trustworthy the individual annotators and classifier are.

2.2 Annotator Quality Scoring Methods

Beyond estimating consensus labels and their quality, we also consider ways to rank which annotators provide the best/worst labels. Here are methods to get a quality score $a_j \in [0, 1]$ for each annotator.

Agreement [Monarch, 2021b]. This basic approach scores annotators via the empirical accuracy of each annotator's labels with respect to the majority-vote consensus label. Examples with only one annotation not considered in this accuracy calculation to prevent overconfident estimates.

$$\begin{aligned} a_j &= \frac{1}{|\mathcal{I}_{j,+}|} \sum_{i \in \mathcal{I}_{j,+}} \mathbb{1}(Y_{ij} = \widehat{Y}_i) \\ \text{where } \mathcal{I}_{j,+} &:= \mathcal{I}_j \cap \mathcal{I}_+ = \{ i \in \mathcal{I}_j : |\mathcal{J}_i| > 1 \} \end{aligned} \tag{13}$$

Label Quality Score [Kuan and Mueller, 2022]. While the agreement-based scores rate annotators solely based on the observed annotator statistics, we can alternatively rely on our classfier predictions $\widehat{p}_{\mathcal{M}}$ to rate the average quality of all labels provided by any one annotator.

$$a_j = \frac{1}{|\mathcal{I}_j|} \sum_{i \in \mathcal{I}_j} L\big(Y_{ij}, \widehat{p}_{\mathcal{M}}(Y_i \mid X_i)\big) \tag{14}$$

CROWDLAB. Our method takes into account both the label quality score of each annotated label (computed based on our classifier) as well as the agreement between each annotator's label with: other annotators' labels and the consensus label. First, we estimate the average label quality score of labels given by each annotator, as in (14), but here using the estimated class probabilities $\widehat{p}_{\text{CR}}$ output by the CROWDLAB method described in Sec. 2.1:

$$Q_j = \frac{1}{|\mathcal{I}_j|} \sum_{i \in \mathcal{I}_j} L\big(Y_{ij}, \widehat{p}_{\text{CR}}(Y_i \mid X_i, \{Y_{ij}\})\big) \tag{15}$$

Next, we compute each annotator's agreement with consensus among examples with over one annotation.

$$A_j = \frac{1}{|\mathcal{I}_{j,+}|} \sum_{i \in \mathcal{I}_{j,+}} \mathbb{1}(Y_{ij} = \widehat{Y}_i) \tag{16}$$

Here $\mathcal{I}_{j,+}$ is defined in (13) and the consensus labels $\widehat{Y}_i$ are established via the CROWDLAB method from Sec. 2.1. We then use the existing model and annotator weights $w_{\mathcal{M}}, w_j$ (computed as part of the CROWDLAB consensus quality score in (7) and (8)) to find a single aggregate weight to compare all annotators against the classifier model.

$$\begin{aligned} \bar{w} &= \frac{w_{\mathcal{M}}}{w_{\mathcal{M}} + w_0} \\ \text{where } w_0 &= \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} w_j \cdot |\mathcal{J}_i| \end{aligned} \tag{17}$$

Note that $\bar{w}$ is shared across all annotators. A quality score for each annotator is finally computed via a weighted average of the label quality score and the annotator agreement with the consensus labels:

$$a_j = \bar{w} Q_j + (1 - \bar{w}) A_j \tag{18}$$

Dawid-Skene [Dawid and Skene, 1979]. We follow the conventional use of Dawid-Skene to rate a particular annotator via the probability that they agree with the true label. This is directly estimated for each possible true label as part of the per-annotator class confusion matrix used by the Dawid-Skene method (see Sec. 2.1), such that one can score each annotator using the trace of their confusion matrix.

$$a_j = \frac{1}{K} \sum_{k=1}^{K} \pi_{k,k}^{(j)} \tag{19}$$

GLAD [Whitehill et al., 2009]. Expertise of each annotator as estimated by GLAD method (see Sec. 2.1): $a_j = \alpha_j$.

Dawid-Skene with Model [Monarch, 2021a]. Add the classfier's predicted labels as an additional annotator (who labeled every example), and score each real annotator's quality using the Dawid-Skene method above.

GLAD with Model [Monarch, 2021a]. Add the classifier's predicted labels as an additional annotator (who labeled every example), and then score each real annotator's quality using the GLAD method above.

2.3 Why CROWDLAB can produce better estimates than other methods

- In settings with few (or only one) labels for an example, the agreement/Dawid-Skene/GLAD scores become unreliable [Paun et al., 2018], but CROWDLAB can utilize additional information provided by a classifier that may be able to generalize to this example (if other dataset examples with similar feature values have more trustworthy consensus labels, e.g. if they received more annotations).

- For examples that received a large number of annotations, CROWDLAB assigns less relative weight to the classifier predictions and its consensus quality score converges toward the observed annotator agreement. This quantity becomes more reliable when based on a large number of annotations [Paun et al., 2018], in which case relying on other sources of information becomes unnecessary. For examples where all annotations agree, an increase in the number of such annotations will typically correspond to an increased CROWDLAB consensus score. The *Label Quality Score* alone fails to exhibit this desirable property.

- Methods like Dawid-Skene estimate $K \times K$ confusion matrices per annotator, which may be statistically challenging when some annotators provide few labels [Paun et al., 2018]. CROWDLAB merely estimates a single likelihood parameter P shared across all classes/annotators in (3) as well as a single per annotator statistic w_j. Both can be better estimated from a limited number of observations.

- Popular crowd-sourcing methods like Dawid-Skene or GLAD are iterative algorithms, with high computational costs when their convergence is slow [Sinha et al., 2018, Stephens, 2000], whereas CROWDLAB does not require iterative updates and is fully deterministic (for a given classifier).

3 Experiments

Datasets. To evaluate various methods, we employ real-world multi-annotator data with naturally occurring label errors. We run three benchmarks based on different subsets of the CIFAR-10H data [Peterson et al., 2019] which we call: *Hardest*, *Uniform*, *Complete* (see Appendix A.1 for details and Appendix B and C for additional results). CIFAR-10H contains multiple labels for images in the CIFAR-10 test set [Krizhevsky and Hinton, 2009], obtained from a large set of new human annotators. As a source of ground truth labels, we simply use the corresponding labels for each image from the original CIFAR-10 dataset [Krizhevsky and Hinton, 2009]. Northcutt et al. [2021a] found the original CIFAR-10 labels to contain few errors in verification studies, and they have been adopted as ground truth labels in other research as well [Kuan and Mueller, 2022].

Models. To study how methods perform across different types of classifiers with varying accuracy, we applied every method twice, once using a ResNet-18 classifier [He et al., 2016] and another time with a Swin Transformer model [Liu et al., 2021]. Both classifiers are trained on the same data (majority-vote consensus labels) in the same manner. Here the Swin Transformer represents a high quality model, whereas ResNet-18 represents a less accurate model (that is still commonly used in practice).

Metrics. To measure each of our three previously stated estimation tasks, we employ the following metrics:

1. To evaluate *how well methods can estimate consensus labels* from multiply-annotated data, we measure the **accuracy** of the inferred consensus label for each example against its ground truth label.

2. To evaluate *how well methods can estimate the quality of each given consensus label*, we compare estimated quality score q_i for each example against a binary target indicating whether or not the consensus label matches the ground truth label. If our goal is to use the quality scores to flag those examples whose consensus label is currently incorrect, this is a form of information retrieval [Kuan and Mueller, 2022]. Thus our consensus quality scores are evaluated via precision/recall metrics: **AUROC**, **AUPRC**, and **Lift** at various cutoffs (which is directly proportional to Precision@T). To focus our evaluation purely on the estimation of label quality, throughout this section, we use each method to estimate quality scores for a single set of consensus labels established via majority vote. We always score the same of consensus labels here because our above evaluation already quantifies how good the consensus labels are from different methods, and we do not want this to confound our evaluation of how well different methods can estimate label quality. While our rigorous evaluation of label quality estimation here is applied to majority-vote consensus labels, in practice, each scoring method can be used to estimate the quality of consensus labels derived via any other approach.

3. To evaluate *how well methods can estimate the quality of each annotator*, we measure the **Spearman correlation** between a_j and ACC$_j$ over all annotators j, where: a_j denotes our estimated annotator quality score (Sec. 2.2) and ACC$_j$ denotes the accuracy of the j-th annotator's chosen labels with respect to the ground truth labels (considering only the subset of examples labeled by annotator j). A method that achieves high Spearman correlation must produce annotator quality scores that are lower for those annotators whose labels tend to be wrong the most often.

Note that all such metrics are for evaluation purposes only, and would not be computable in real applications of our methodology due to a lack of ground truth labels. For evaluating consensus quality scores, AUROC measures how well these scores are able to differentiate correct and incorrect consensus labels. AUPRC accounts for the precision/recall of the consensus quality scores in flagging an incorrect consensus label, in a manner that is more sensitive to proportion of errors in the majority-vote consensus label errors than AUROC [Davis and Goadrich, 2006]. The Lift at T metric measures how much more likely we are to encounter an incorrect consensus label among the top T ranked examples that have the worst consensus quality score.

4 Results

Figures 2, S1, S2, and Tables 1, S2, S3 demonstrate that CROWDLAB overall performs the best across our evaluations for consensus and annotator quality scores, and also typically produces the most accurate consensus labels. For most methods considered in this paper, all evaluation metrics improve when used with the Swin Transformer vs. ResNet-18 model. This illustrates how a better classifier can be utilized to get more improvement in consensus labels and consensus/annotator quality estimates. Effective methods for multi-annotator analysis must remain compatible with future innovations in classifier technology.

Considering only classifier predictions and consensus labels (rather than individual annotator information), the *Label Quality Score* also effectively estimates consensus quality when we have an accurate model (Swin Transformer). Predictions from a strong classifier suffice to estimate label quality without additional information provided by individual annotators [Kuan and Mueller, 2022]. However *Label Quality Score* performs worse than other methods with a lower accuracy classifier (ResNet-18). This demonstrates the value of accounting for the individual annotations and overall model accuracy in CROWDLAB, which performs well relative to other methods regardless of the classifier's accuracy. Treating the classifier as an additional annotator for the Dawid-Skene and GLAD methods improves their performance, but not enough to match CROWDLAB, which better accounts for the classifier's confidence. While the *Empirical Bayes* method also accounts for classifier confidence to augment Dawid-Skene, it similarly unable to match CROWDLAB, demonstrating why our method considers *how much* to weigh the model based on its estimated trustworthiness relative to the annotators.

We also compare CROWDLAB against a variant of this method which lacks the per-annotator quality estimation (i.e. all annotator weights w_j are equal), and find this variant underperforms as it estimates *too little* information about the annotators (results in Appendix E). On another *Uniform* dataset in which there are 1-5 annotations for each example occurring with equal frequencies, CROWDLAB is able to produce better estimates for tasks (1)-(3) than other methods (results in Appendix B). On another *Complete* dataset with many more (~ 50) annotations per example, such that simple annotator agreement and majority vote produce highly accurate estimates, CROWDLAB retains its strong performance compared to other methods (results in Appendix C). We also run all methods with an unrealistically accurate classifier on all datasets (results in Appendix D), a setting that favors the *Label Quality Score*, and find that CROWDLAB still outperforms the other methods. This breadth of settings highlights the utility of CROWDLAB across a wide range of applications involving mediocre/stellar classfier models and datasets with varying numbers of collected annotations.

5 Discussion

Unlike other ways to utilize classifiers with crowdsourcing algorithms, CROWDLAB considers a model's estimated confidence and how accurate it is relative to individual annotators. Methods such as Dawid-Skene with Model and GLAD with Model take into account the model predictions but fail to take into model confidence and accuracy, whereas CROWDLAB carefully considers how good the classifier model is relative to the annotators. Our proposed methodology is compatible with any classifier and training strategy, ensuring its out-of-the-box performance will improve as new models and training tricks are invented.

Naturally, the efficacy of CROWDLAB depends on being able to train a reasonably performant classifier, unlike generative models of annotator statistics like Dawid-Skene. Fortunately, training good classifiers is easy nowadays with AutoML [Erickson et al., 2020] and versatile techniques for calibration, data augmentation, and transfer learning [Thulasidasan et al., 2019]. Another limitation of our approach is its marginal benefit in settings where every example is labeled by a large number of annotators, in which simple annotator agreement effectively measures quality (Appendix C). Labeling budgets however prevent many applications from benefitting from this law of large numbers.

Our research introduces practical and accurate estimates for consensus labels and their quality as well as annotator quality; we expect their broader impact to be an improvement in supervised learning and analytics with datasets labeled by multiple annotators. As with most classification projects, CROWDLAB users must remain wary of overconfident model predictions with limited ability to generalize, which may lead to overly optimistic estimates of quality. CROWDLAB is highly modular and future work can improve its various components such as the: classifier and label quality score $L(\cdot)$ utilized, or the estimation of annotator weights and classifier vs. annotator accuracy. Extending the methodology to settings with possible collusion between annotators may also be of interest [Song et al., 2021].

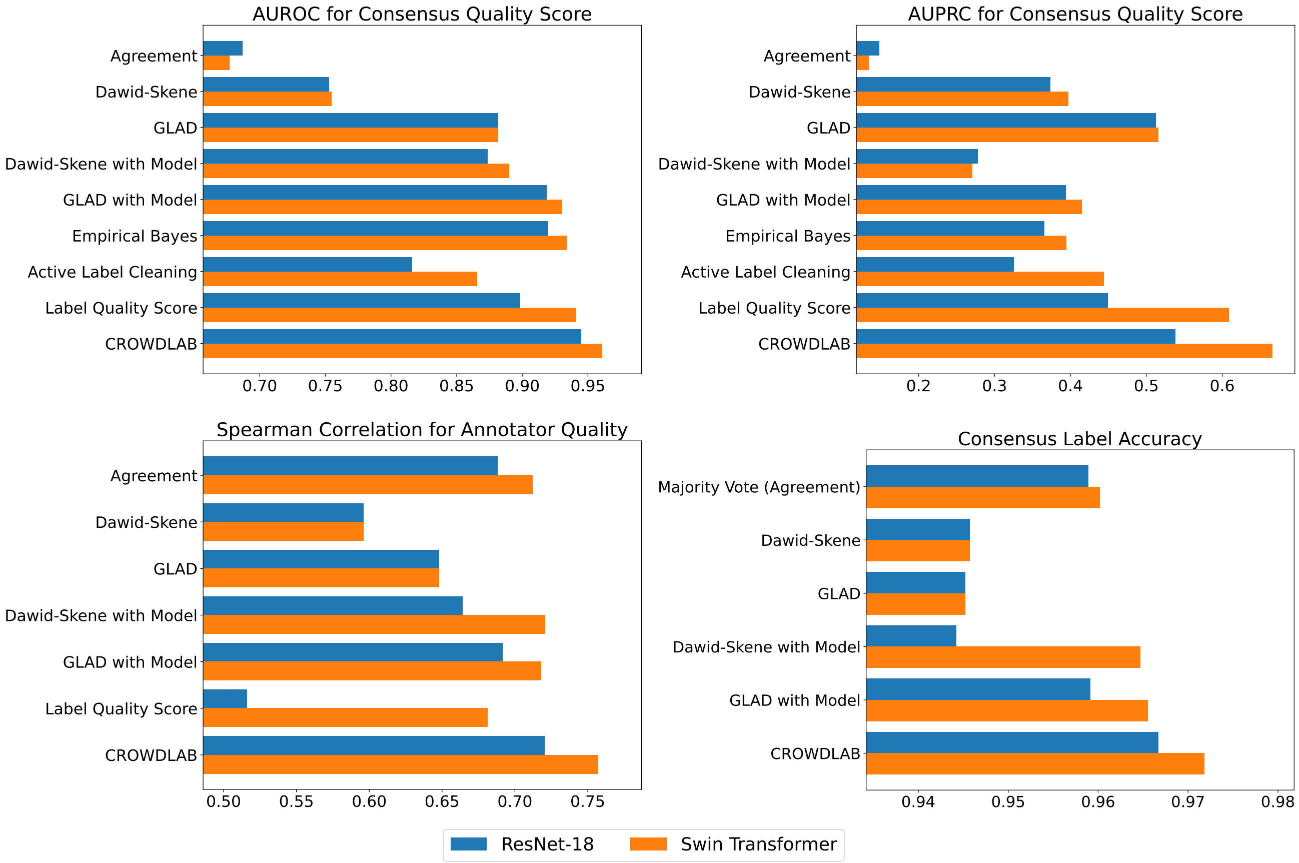


Figure 2: Benchmarking multi-annotator methods on the *Hardest* dataset.

Model	**Method**	**Lift @ 10**	**Lift @ 50**	**Lift @ 100**	**Lift @ 300**	**Lift @ 500**
ResNet-18	Agreement	4.87	5.84	6.33	5.27	4.72
ResNet-18	Dawid-Skene	12.89	11.79	13.26	10.74	8.51
ResNet-18	GLAD	14.6	15.69	15.88	13.93	9.96
ResNet-18	Dawid-Skene with Model	12.54	11.47	8.6	5.56	5.09
ResNet-18	GLAD with Model	14.67	13.69	14.43	11.25	10.81
ResNet-18	Empirical Bayes	12.17	12.17	11.68	11.11	10.27
ResNet-18	Active Label Cleaning	17.03	19.95	16.3	10.22	7.88
ResNet-18	Label Quality Score	19.46	21.41	19.22	13.38	10.22
ResNet-18	CROWDLAB	24.33	22.38	17.76	14.27	11.82
Swin	Agreement	2.51	6.03	6.28	4.86	4.17
Swin	Dawid-Skene	12.89	14.36	14.55	10.99	8.66
Swin	GLAD	14.6	15.69	15.88	14.11	10.04
Swin	Dawid-Skene with Model	14.16	16.43	12.46	9.16	7.99
Swin	GLAD with Model	8.7	17.39	17.1	15.36	11.71
Swin	Empirical Bayes	12.56	9.55	11.06	11.81	11.76
Swin	Active Label Cleaning	25.13	22.11	21.36	12.81	9.25
Swin	Label Quality Score	25.13	22.61	21.86	16.92	12.81
Swin	CROWDLAB	25.13	24.62	20.85	17.76	13.82

Table 1: Evaluating the precision of various consensus quality scoring methods on the *Hardest* dataset. Lift@T is directly proportional to Precision@T, and reports what fraction of the top-T ranked consensus labels are actually incorrect normalized by the fraction of incorrect consensus labels expected for a random set of examples.

References

M. Bernhardt, D. C. Castro, R. Tanno, A. Schwaighofer, K. C. Tezcan, M. Monteiro, S. Bannur, M. P. Lungren, A. Nori, B. Glocker, et al. Active label cleaning for improved dataset quality under resource constraints. *Nature communications*, 13(1):1–11, 2022.

B. Carpenter. Multilevel bayesian models of categorical data annotation. 2008.

J. Davis and M. Goadrich. The relationship between precision-recall and ROC curves. In *International Conference on Machine learning*, pages 233–240, 2006.

A. P. Dawid and A. 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, 1979.

N. Erickson, J. Mueller, A. Shirkov, H. Zhang, P. Larroy, M. Li, and A. Smola. AutoGluon-Tabular: Robust and accurate AutoML for structured data. *arXiv preprint arXiv:2003.06505*, 2020.

R. Fakoor, T. Kim, J. Mueller, A. J. Smola, and R. J. Tibshirani. Flexible model aggregation for quantile regression. *arXiv preprint arXiv:2103.00083*, 2021.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Hivemind and Cloudfactory. Crowd vs. managed team: A study on quality data processing at scale. URL https://go.cloudfactory.com/hubfs/02-Contents/3-Reports/Crowd-vs-Managed-Team-Hivemind-Study.pdf.

D. Hovy, T. Berg-Kirkpatrick, A. Vaswani, and E. Hovy. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2013.

Y. E. Kara, G. Genc, O. Aran, and L. Akarun. Modeling annotator behaviors for crowd labeling. *Neurocomputing*, 160:141–156, 2015.

D. Karger, S. Oh, and D. Shah. Iterative learning for reliable crowdsourcing systems. *Advances in Neural Information Processing Systems*, 2011.

A. Khetan, Z. C. Lipton, and A. Anandkumar. Learning from noisy singly-labeled data. In *International Conference on Learning Representations*, 2018.

A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

J. Kuan and J. Mueller. Model-agnostic label quality scoring to detect real-world label errors. In *ICML DataPerf Workshop*, 2022.

Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

R. Monarch. Treating model predictions as a single annotator. In *Human-in-the-loop machine learning*. Manning Publications, 2021a.

R. Monarch. *Human-in-the-loop machine learning*. Manning Publications, 2021b.

R. Monarch. Quality control for data annotation. In *Human-in-the-loop machine learning*. Manning Publications, 2021c.

Q. Nguyen, H. Valizadegan, and M. Hauskrecht. Learning classification models with soft-label information. *Journal of the American Medical Informatics Association*, 21(3): 501–508, 2014.

C. G. Northcutt, A. Athalye, and J. Mueller. Pervasive label errors in test sets destabilize machine learning benchmarks. In *Proceedings of the 35th Conference on Neural Information Processing Systems Track on Datasets and Benchmarks*, December 2021a.

C. G. Northcutt, L. Jiang, and I. L. Chuang. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70:1373–1411, 2021b.

S. Paun, B. Carpenter, J. Chamberlain, D. Hovy, U. Kruschwitz, and M. Poesio. Comparing Bayesian models of annotation. *Transactions of the Association for Computational Linguistics*, 6:571–585, 2018.

J. C. Peterson, R. M. Battleday, T. L. Griffiths, and O. Russakovsky. Human uncertainty makes classification more robust. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

E. A. Platanios, M. Al-Shedivat, E. Xing, and T. Mitchell. Learning from imperfect annotations. *arXiv preprint arXiv:2004.03473*, 2020.

V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy. Learning from crowds. *Journal of Machine Learning Research*, 11(4), 2010.

F. Rodrigues and F. Pereira. Deep learning from crowds. In *Proceedings of the AAAI conference on artificial intelligence*, 2018.

A. Sheshadri and M. Lease. Square: A benchmark for research on computing crowd consensus. In *First AAAI conference on human computation and crowdsourcing*, 2013.

V. B. Sinha, S. Rao, and V. N. Balasubramanian. Fast Dawid-Skene: A fast vote aggregation scheme for sentiment classification. *arXiv preprint arXiv:1803.02781*, 2018.

C. Song, K. Liu, and X. Zhang. Collusion detection and ground truth inference in crowdsourcing for labeling tasks. *Journal of Machine Learning Research*, 22:190–1, 2021.

M. Stephens. Dealing with label switching in mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(4):795–809, 2000.

S. Thulasidasan, G. Chennupati, J. A. Bilmes, T. Bhattacharya, and S. Michalak. On mixup training: Improved calibration and predictive uncertainty for deep neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Toloka. Crowd-kit: Computational quality control for crowdsourcing. URL https://toloka.ai/en/docs/crowd-kit.

J. Whitehill, T.-f. Wu, J. Bergsma, J. Movellan, and P. Ruvolo. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in Neural Information Processing Systems*, 2009.

Appendix: Utilizing supervised models to infer consensus labels and their quality from data with multiple annotators

A Experiment Details

Our experiments employ two of the most currently popular architectures for image classification, which are intended to be representative of different types of models one might use in practice. Training of the Swin Transformer and ResNet classifiers was done as by Kuan and Mueller [2022], using 5-fold cross-validation starting with ImageNet-pretrained weights fine-tuned in each fold via AutoML [Erickson et al., 2020]. We do not evaluate annotator quality scores from the *Active Label Cleaning* method because rating annotators was left as future work in the paper of Bernhardt et al. [2022]. Annotator quality estimates from the *Empirical Bayes* approach match those from *Dawid-Skene* and are also omitted from our plots.

A.1 Datasets

The original CIFAR-10 dataset [Krizhevsky and Hinton, 2009] is fairly easy to label [Northcutt et al., 2021a], and the annotator agreement on the complete CIFAR-10H data [Peterson et al., 2019] is unrealistically high for a representative benchmark. This is because the images are not only relatively easy to label but there are also a large number of annotators (~ 50) per image in CIFAR-10H (it is uncommon to have so many annotators per example in practice). Hence, our primary benchmark uses a subset of the CIFAR-10H annotator labels. This subset starts with the 25 worst annotators and then incrementally add annotators from worst to best (based on their accuracy vs. ground-truth labels) until each of the 10,000 examples have at least 1 annotation (resulting in a dataset with 511 annotators in total). During this process, we restricted the selection of each new annotator to add to the current subset to only those which labeled at least one example also labeled by one of the annotators in the current subset. We call this variation of CIFAR-10H the *Hardest* dataset benchmarked in this paper, and believe it is more representative of real-world data labeling applications, where the proportion of label errors tends to be far higher than in CIFAR-10H and the number of annotators far lower [Hivemind and Cloudfactory].

To ensure the robustness of our conclusions, we also evaluated all methods on two other datasets: a *Uniform* subset of CIFAR-10H (only considering some randomly chosen annotators such that each example has between 1-5 annotations with an equal number of examples receiving 1 annotation, 2 annotations, etc.), and the *complete* CIFAR-10H dataset (with all annotator labels, which is far more than typically collected in most applications). Results for these other datasets are in Appendix B and C, and are based on separate classifier models trained for each dataset. In all cases, we only consider images from the *test set* of CIFAR-10 (here treated as multiply-labeled training data), since these are the only images labeled by many annotators in CIFAR-10H.

Labels predicted by	**Accuracy (w.r.t. ground truth labels)**
ResNet-18	0.879
Swin Transformer	0.940
Swin Transformer trained with true labels	0.948
Annotator (Average)	0.909

Table S1: Classification accuracy for the *Hardest* dataset achieved by various predictors: ResNet-18 and Swin Transformer classifiers trained on majority-vote consensus labels (i.e. the models used in the benchmark results of Figure 2), Swin Transformer trained on true labels, which represents an unrealistically good classifier (see Appendix D), as well as the average annotator in the dataset.

B Results for Uniform Dataset

To evaluate our methods in another setting, we construct a different subset of CIFAR10-H and re-run our benchmark on this new dataset. In this *Uniform* dataset, each example now has between 1 to 5 labels, where the number of labels per example are uniformly distributed. This dataset contains 421 annotators and 10,000 examples. Here the annotators are just randomly selected from the CIFAR10-H pool, and are thus higher quality than in the *Hardest* dataset. The following results demonstrate that CROWDLAB is also the best method overall for this *Uniform* dataset.

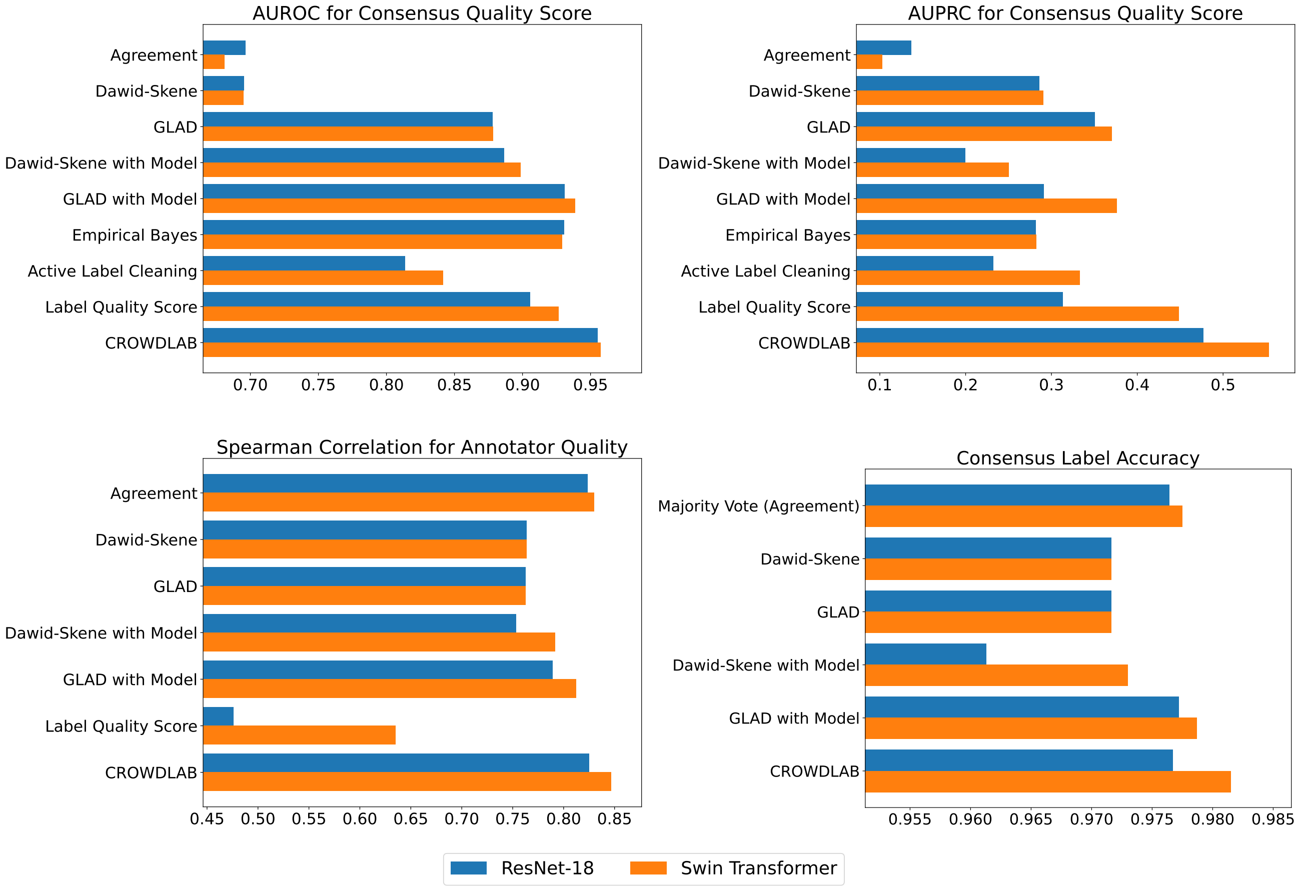


Figure S1: Benchmarking multi-annotator methods on the *Uniform* dataset.

Model	Quality Method	Lift @ 10	Lift @ 50	Lift @ 100	Lift @ 300	Lift @ 500
ResNet-18	Agreement	21.19	11.86	9.75	9.04	7.12
ResNet-18	Dawid-Skene	31.69	24.65	19.01	12.09	8.52
ResNet-18	GLAD	31.69	22.54	25.7	13.03	9.23
ResNet-18	Dawid-Skene with Model	10.34	11.37	7.24	5.43	4.86
ResNet-18	GLAD with Model	17.54	18.42	15.79	13.01	12.72
ResNet-18	Empirical Bayes	12.71	20.34	18.22	13.98	11.44
ResNet-18	Active Label Cleaning	33.9	24.58	16.1	10.03	7.88
ResNet-18	Label Quality Score	33.9	26.27	22.46	12.43	9.75
ResNet-18	CROWDLAB	42.37	33.05	27.97	16.95	13.81
Swin	Agreement	8.89	8.0	7.56	7.85	6.49
Swin	Dawid-Skene	28.17	27.46	20.07	11.74	8.45
Swin	GLAD	35.21	25.35	27.11	13.03	9.23
Swin	Dawid-Skene with Model	33.33	16.3	12.22	8.89	8.15
Swin	GLAD with Model	23.47	23.47	23.0	19.87	14.74
Swin	Empirical Bayes	13.33	17.78	17.78	14.96	12.44
Swin	Active Label Cleaning	44.44	30.22	24.0	14.07	10.13
Swin	Label Quality Score	35.56	32.89	29.33	18.67	13.6
Swin	CROWDLAB	40.0	35.56	32.0	21.19	15.2

Table S2: Evaluating the precision of various consensus quality scoring methods on the *Uniform* dataset. Lift@T is directly proportional to Precision@T, and reports what fraction of the top-T ranked consensus labels are actually incorrect normalized by the fraction of incorrect consensus labels expected for a random set of examples.

C Results for Complete Dataset

We also evaluate our methods on the full original CIFAR-10H dataset [Peterson et al., 2019]. This *Complete* dataset contains 2571 annotators where each annotator labels 200 examples, such that each of the 10,000 images has approximately 50 annotations. The *Complete* dataset has by far the highest number of annotations per example, and more than are available in most real-world multi-annotator datasets. With so many annotations per example, basic annotator agreement methods are highly effective. CROWDLAB works similarly well, highlighting its adaptive nature.

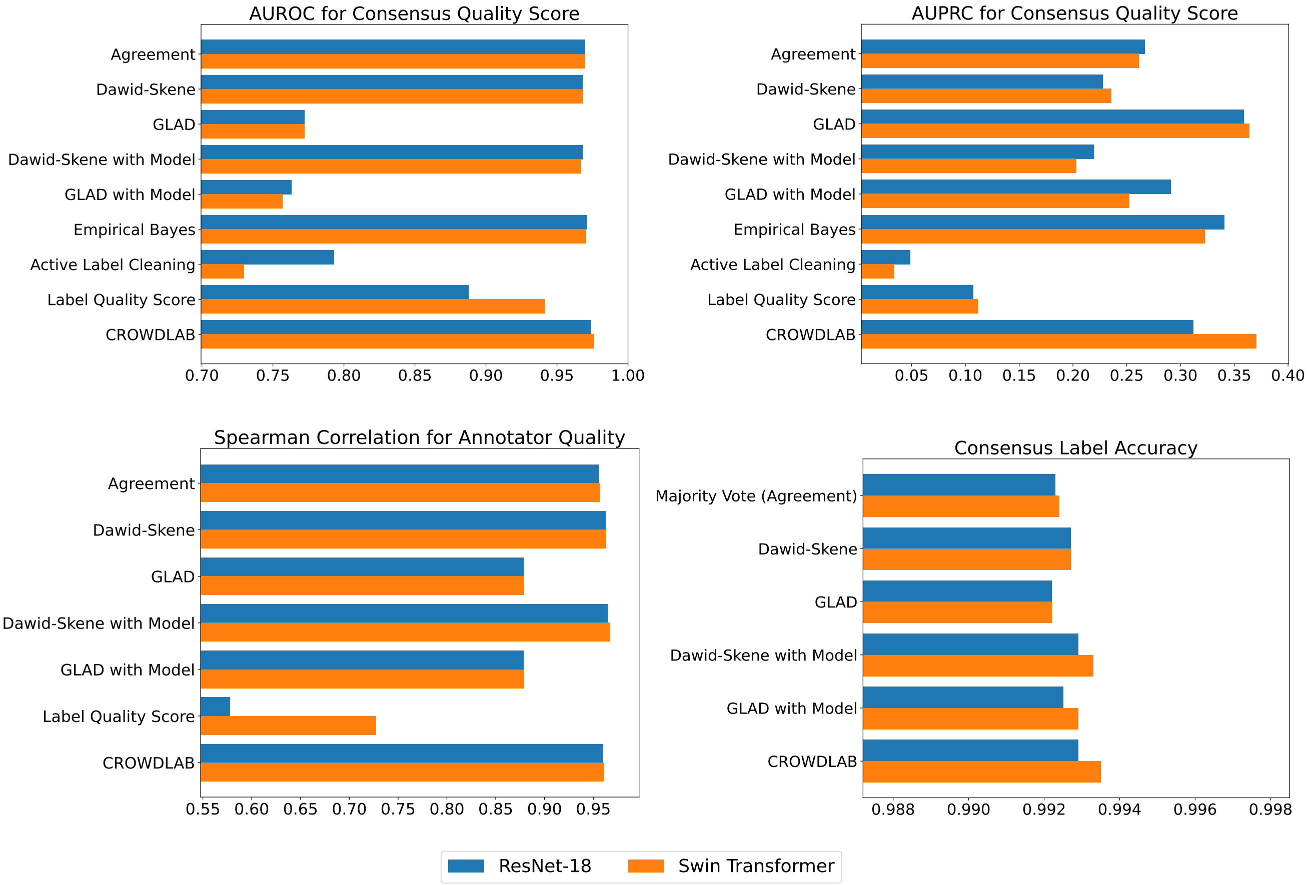


Figure S2: Benchmarking multi-annotator methods on the *Complete* dataset.

Model	Quality Method	Lift @ 10	Lift @ 50	Lift @ 100	Lift @ 300	Lift @ 500
ResNet-18	Agreement	25.97	33.77	44.16	26.41	17.92
ResNet-18	Dawid-Skene	27.4	41.1	39.73	25.57	17.81
ResNet-18	GLAD	89.74	66.67	47.44	18.38	11.03
ResNet-18	Dawid-Skene with Model	28.17	36.62	38.03	26.29	17.75
ResNet-18	GLAD with Model	53.33	61.33	46.67	17.78	10.67
ResNet-18	Empirical Bayes	77.92	49.35	44.16	26.84	18.18
ResNet-18	Active Label Cleaning	0.0	10.39	12.99	8.66	8.83
ResNet-18	Label Quality Score	38.96	20.78	19.48	13.42	9.09
ResNet-18	CROWDLAB	38.96	49.35	44.16	27.71	18.44
Swin	Agreement	26.32	34.21	43.42	26.32	17.89
Swin	Dawid-Skene	41.1	41.1	39.73	25.57	17.81
Swin	GLAD	89.74	66.67	47.44	18.38	11.03
Swin	Dawid-Skene with Model	14.93	38.81	37.31	25.87	17.61
Swin	GLAD with Model	28.17	53.52	46.48	17.37	10.42
Swin	Empirical Bayes	78.95	50.0	42.11	26.32	17.89
Swin	Active Label Cleaning	0.0	0.0	5.26	7.46	6.84
Swin	Label Quality Score	26.32	21.05	21.05	17.11	13.68
Swin	CROWDLAB	65.79	65.79	48.68	28.07	18.68

Table S3: Evaluating the precision of various consensus quality scoring methods on the *Complete* dataset. Lift@T is directly proportional to Precision@T, and reports what fraction of the top-T ranked consensus labels are actually incorrect normalized by the fraction of incorrect consensus labels expected for a random set of examples.

D Model Trained on True CIFAR-10 Labels

In this section, we investigate how the methods perform when utilizing a highly accurate model. We obtain such a model by training a Swin Transformer on the ground truth labels rather than consenus labels estimated from the given annotations (this would not be possible in real applications). All benchmark results presented in this section are with respect to this unrealistically good classifier.

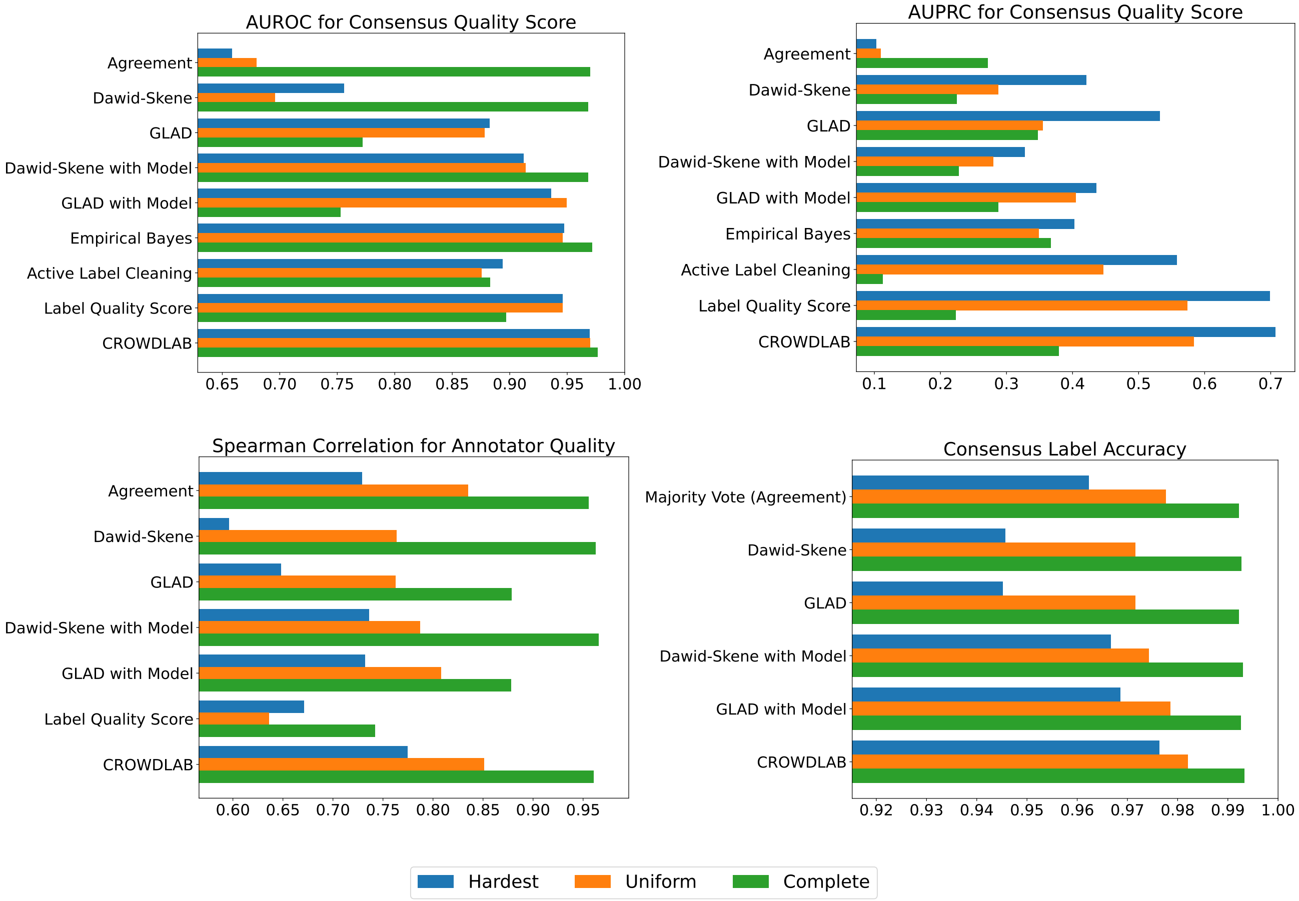


Figure S3: Benchmark results using unrealistically good classifier fit to true labels for each dataset.

Quality Method	Lift @ 10	Lift @ 50	Lift @ 100	Lift @ 300	Lift @ 500
Agreement	2.65	3.18	3.98	4.16	3.77
Dawid-Skene	14.73	15.47	15.1	11.48	8.95
GLAD	14.6	15.69	16.24	14.6	10.07
Dawid-Skene with Model	24.02	18.62	15.02	10.11	9.13
GLAD with Model	19.11	21.66	20.06	16.45	11.97
Empirical Bayes	5.31	7.43	10.34	12.73	12.84
Active Label Cleaning	23.87	25.46	25.2	16.18	11.03
Label Quality Score	23.87	24.93	24.93	20.07	14.64
CROWDLAB	26.53	25.99	24.14	19.19	14.8

Table S4: Evaluating the lift (i.e. precision) of various consensus quality scoring methods on the *Hardest* dataset, here employing our unrealistically good classifier trained with true labels.

Quality Method	**Lift @ 10**	**Lift @ 50**	**Lift @ 100**	**Lift @ 300**	**Lift @ 500**
Agreement	8.93	10.71	8.48	7.44	6.34
Dawid-Skene	28.17	25.35	20.42	12.21	8.73
GLAD	31.69	24.65	27.82	13.03	9.23
Dawid-Skene with Model	31.01	19.38	14.73	9.95	9.69
GLAD with Model	13.95	25.12	23.72	20.78	15.44
Empirical Bayes	13.39	19.64	20.98	19.05	14.38
Active Label Cleaning	40.18	39.29	32.14	17.11	11.34
Label Quality Score	40.18	40.18	36.16	19.79	14.38
CROWDLAB	44.64	33.04	34.38	22.32	15.98

Table S5: Evaluating the lift (i.e. precision) of various consensus quality scoring methods on the *Uniform* dataset, here employing our unrealistically good classifier trained with true labels.

Quality Method	**Lift @ 10**	**Lift @ 50**	**Lift @ 100**	**Lift @ 300**	**Lift @ 500**
Agreement	25.64	35.9	43.59	26.5	17.95
Dawid-Skene	27.4	38.36	39.73	25.57	17.81
GLAD	89.74	66.67	47.44	18.38	11.28
Dawid-Skene with Model	42.86	40.0	38.57	26.67	17.71
GLAD with Model	40.54	54.05	45.95	17.12	10.54
Empirical Bayes	89.74	48.72	44.87	26.92	18.21
Active Label Cleaning	38.46	23.08	19.23	14.53	11.54
Label Quality Score	64.1	51.28	38.46	18.38	12.56
CROWDLAB	89.74	61.54	42.31	26.92	18.46

Table S6: Evaluating the lift (i.e. precision) of various consensus quality scoring methods on the *Complete* dataset, here employing our unrealistically good classifier trained with true labels.

E Variant of our Method Without Per Annotator Weights

Here we present results for a simpler variant of CROWDLAB that we also explored, henceforth called *No Perannotator Weights*. The two approaches are overall the same, except while CROWDLAB considers each annotator individually and assigns them a separate weight w_j, *No Perannotator Weights* aggregates all the annotators and treats them as one "average annotator" to be weighed against the classifier model. Details of the *No Perannotator Weights* approach are presented below.

E.1 Consensus Quality Method

Just as in CROWDLAB, we estimate the quality of consensus labels via the label quality score based on estimated class probabilities. In the *No Perannotator Weights* variant, these probabilities are computed via a slightly different weighted average:

$$\widehat{p}_{\text{NPW}}(Y_i \mid X_i, \{Y_{ij}\}) = \frac{w_{\mathcal{M}} \cdot \widehat{p}_{\mathcal{M}}(Y_i \mid X_i) + w_{\mathcal{A}} \cdot \widehat{p}_{\mathcal{A}}(Y_i \mid \{Y_{ij}\})}{w_{\mathcal{M}} + w_{\mathcal{A}}} \tag{20}$$

where $w_{\mathcal{M}} = w \cdot \frac{1}{n} \sum_i \sqrt{|\mathcal{J}_i|},\ w_{\mathcal{A}} = (1-w) \cdot \sqrt{|\mathcal{J}_i|}$ are one weight for the model and one weight applied to all annotators. Both depend on w, whose definition follows a similar strategy as used in CROWDLAB for individual annotators, but here applied to their aggregate output.

First let's recall these quantities from Sec. 2.1: s_j represents annotator j's agreement with other annotators who labeled the same examples and is defined in (4), A_j represents the accuracy of each annotator's labels with respect to the majority-vote consensus label for examples with more than one annotation and is defined in (16). In this variant, we compute an average annotator accuracy $\bar{A}$ by taking the average of each annotator's accuracy weighted simply by the number of examples each annotator labeled (rather than their estimated trustworthiness).

$$\bar{A} = \frac{\sum_j A_j \cdot |\mathcal{I}_j|}{\sum_j |\mathcal{I}_j|}$$

Let $A_{\mathcal{M}}$ represent the accuracy of the model with respect to the majority-vote consensus labels among examples with more than one annotation, as defined in (5). We then choose our weight $w = A_{\mathcal{M}}/(A_{\mathcal{M}} + \bar{A})$ to balance model accuracy vs. that of the average annotator.

While CROWDLAB uses a separate class likelihood vector for each annotator, this variant only considers their aggregate class likelihood

$$\widehat{p}_{\mathcal{A}}(Y_i = k \mid \{Y_{ij}\}) = \frac{1}{|\mathcal{J}_i|} \sum_{j \in \mathcal{J}_i} P_j \quad \text{where } P_j = \begin{cases} s_j & \text{when } Y_{ij} = k \\ \frac{1-s_j}{K-1} & \text{when } Y_{ij} \neq k \end{cases}$$

E.2 Annotator Quality Method

In the *No Perannotator Weights* variant, we score the quality of each annotator via:

$$a_j = w \cdot Q_j + (1-w) \cdot A_j$$

Here Q_j the average label quality score of labels given by each annotator, computed via (15) as in CROWDLAB, but here based on class probabilities $\widehat{p}_{\text{NPW}}$ estimated using *No Perannotator Weights* defined in (20) in place of $\widehat{p}_{\text{CR}}$. A_j and w are defined as above in Sec. E.1.

E.3 Benchmarking CROWDLAB with/without per annotator weights

Ignoring the strengths and weakness of each individual annotator when aggregating them is overall detrimental to CROWDLAB. However the performance reduction due to this modification is surprisingly small, given how important accounting for annotators' relative quality is stated to be in the crowdsourcing literature [Hovy et al., 2013, Karger et al., 2011, Kara et al., 2015, Dawid and Skene, 1979, Whitehill et al., 2009]. Rather the key aspects behind the success of CROWDLAB are its careful consideration of: how much to trust the classifier model vs. the aggregate annotations along with how many annotations were provided for each example. Studying additional variants of CROWDLAB with either of these two pieces removed produced very poor results in our benchmarks.

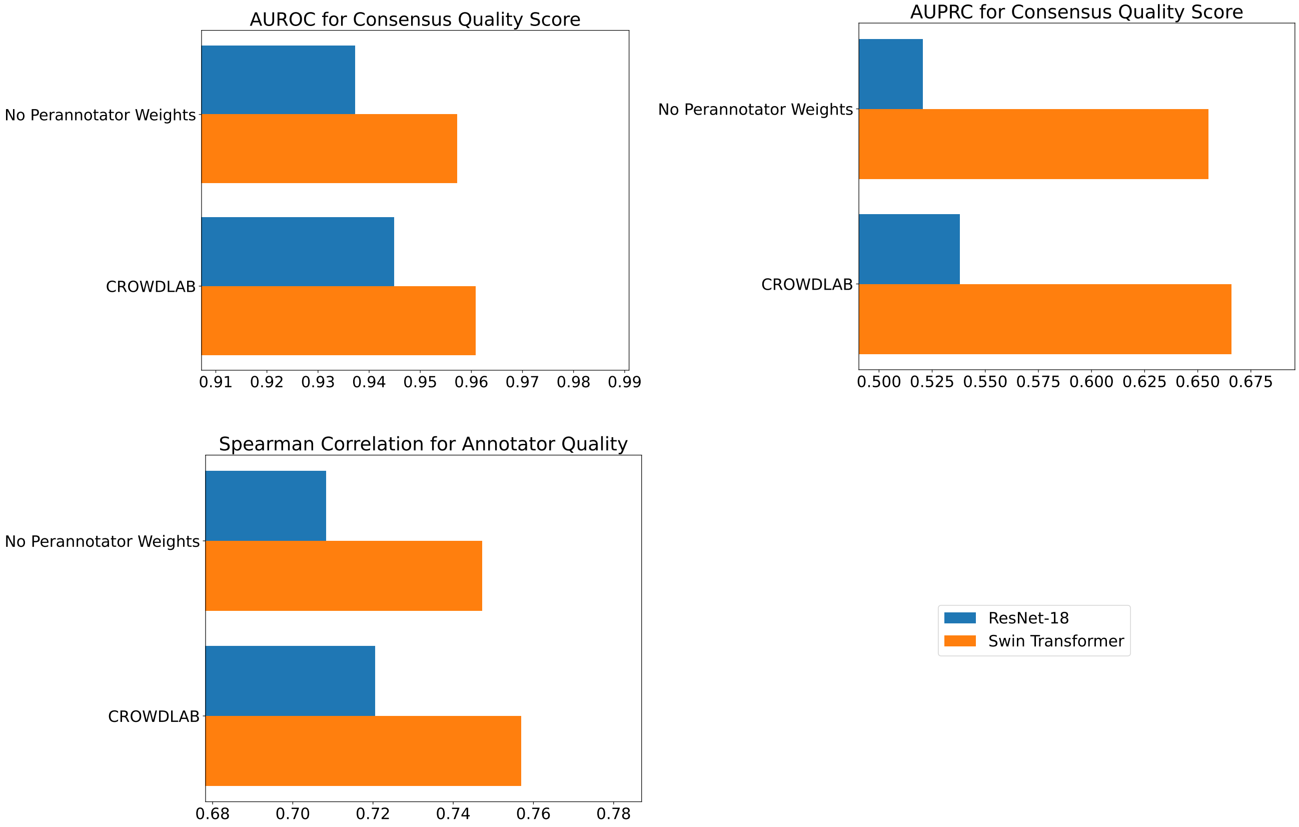


Figure S4: Benchmarking CROWDLAB with/without per annotator weights on the *Hardest* dataset.

Model	**Quality Method**	**Lift @ 10**	**Lift @ 50**	**Lift @ 100**	**Lift @ 300**	**Lift @ 500**
ResNet-18	No Perannotator Weights	24.33	21.9	18.25	13.95	11.92
ResNet-18	CROWDLAB	24.33	22.38	17.76	14.27	11.82
Swin	No Perannotator Weights	25.13	23.62	20.85	17.84	14.02
Swin	CROWDLAB	25.13	24.62	20.85	17.76	13.82

Table S7: Evaluating the precision of CROWDLAB consensus quality scores with/without per annotator weights on the *Hardest* dataset. Lift@T is directly proportional to Precision@T, and reports what fraction of the top-T ranked consensus labels are actually incorrect normalized by the fraction of incorrect consensus labels expected for a random set of examples.