Leveraging Side Information to Improve Label Quality Control in Crowd-sourcing

Yuan Jin and **Mark Carman**
Faculty of Information Technology
Monash University
{yuan.jin, mark.carman}@monash.edu

Dongwoo Kim and **Lexing Xie**
College of Engineering and Computer Science
Australian National University
{dongwoo.kim, lexing.xie}@anu.edu.au

Abstract

We investigate the possibility of leveraging side information for improving quality control over crowd-sourced data. We extend the GLAD model, which governs the probability of correct labeling through a logistic function in which worker expertise counteracts item difficulty, by systematically encoding different types of side information, including worker information drawn from demographics and personality traits, item information drawn from item genres and content, and contextual information drawn from worker responses and labeling sessions. Modeling side information allows for better estimation of worker expertise and item difficulty in sparse data situations and accounts for worker biases, leading to better prediction of posterior true label probabilities. We demonstrate the efficacy of the proposed framework with overall improvements in both the true label prediction and the unseen worker response prediction based on different combinations of the various types of side information across three new crowd-sourcing datasets. In addition, we show the framework exhibits potential of identifying salient side information features for predicting the correctness of responses without the need of knowing any true label information.

Introduction

Crowd-sourcing, the process of outsourcing *human intelligence tasks* to an undefined, generally large group of people to seek answers via an open call (Sheng, Provost, and Ipeirotis 2008), has gained much popularity in machine learning communities in recent years for generating and collecting labeled data, thanks to the development of the corresponding online service providers, such as Amazon Merchanical Turk[1] and CrowdFlower[2]. These platforms facilitate large-scale online data labeling and collection processes in an inexpensive and timely manner. However, they are also constantly confronted by workers with various motives and abilities, who end up producing conflicting labels for the same items. Moreover, an ever-growing number of unlabeled items versus limited budgets in most crowd-sourcing projects often results in a small number of responses per item. Aggregating such small numbers of conflicting responses using majority vote to infer the true label of each item is often unreliable.

To overcome the above problem, the quality of responses must be controlled in a principled manner such that the influence of "high-quality" responses can outweigh that of "low-quality" ones when aggregated for the true labels. This activity is known as Quality Control for Crowd-sourcing (QCCs) (Lease 2011). The QCCs methods, largely based on statistical modeling and machine learning, consider the *abilities* of workers to govern the quality of the labels they produce with greater abilities indicating higher quality (Dawid and Skene 1979; Raykar et al. 2010a). Some of them also consider the *difficulty* of items that can counteract the worker abilities to undermine the label quality (Rasch 1960; Whitehill et al. 2009; Bachrach et al. 2012). These methods have overall achieved superior performance over the conventional majority vote, and the basic pre-task or in-task incompetent worker filtering using control questions. In fact, it is typical for the QCCs methods to follow the filtering in a pipline to enhance the quality control performance nowadays.

However, there exists one major pitfall of the current QCCs models which is that they are vulnerable to the *label sparsity* problem, which happens frequently in real-world crowd-sourcing scenarios where only few labels get collected for each item or from each worker for a task (Jung and Lease 2012). Consequently, parameter estimation for these models (e.g. estimates of worker abilities and item difficulty) becomes unreliable, causing the models' QCCs performance to deteriorate. As an example, due to the paucity of her provided labels, an expert worker can be considered inaccurate by the QCCs models if most of her labels happen to disagree with the majority, while a novice worker can be considered as accurate if her happens to have made some fortunate guesses. In this case, extra side information about the demographics of these workers, the questions they have answered, the time they have taken to respond, or even their situated environments can possibly help to improve the estimation of their parameters in the models when the labels they provide are scarce. Following the previous example, if we know that the demographics of the expert (e.g. her education, interests related to the particular crowd-sourcing task) is very similar to those of the other experts who have been labeling correctly (e.g. by agreeing with the majority), then

[1] https://www.mturk.com/

[2] https://www.crowdflower.com/

the belief of her being a novice due to her poor performance so far can be counteracted by her similarity with the other experts. As a result, her next label will be more trusted by the models.

In most crowd-sourcing tasks, there is always extra side information that is easy to collect (e.g. labeling prices, worker comments or feedback), or can be collected with some minor efforts (e.g. designing simple surveys to collect demographics or programming to collect them silently). The side information can be elicited from the items, the workers and the contexts in which worker-item interaction takes place during the crowd-sourcing process. Relevant work in crowd-sourcing thus far has focused on exploiting only very specific types of side information for improving the performance of QCCs. To our best knowledge, a study that develops a scalable framework able to integrate and utilize arbitrary types of side information for improving true label prediction is still missing.

The corresponding contributions of this paper are summarized as follows:

- We have developed a probabilistic framework, by extending the basic GLAD model (Whitehill et al. 2009), that seamlessly integrates various types of side information while modeling the interaction between the worker expertise and the item difficulty;
- Overall improvements in both the true label prediction and the unseen (held-out) worker response prediction have been found in our experiments over three new crowd-sourcing datasets all of which experience the label sparsity problem for their items. The experiments involve (1) the comparison between the different instantiations of our framework with incremental amounts of side information and three baseline methods in terms of the above two prediction tasks, and (2) their comparison in terms of the same prediction tasks by learning from small random response subsets sampled from the workers of the three datasets where they are exposed to escalated sparsity across both the workers and the items.
- We show the framework has the potential of automatically identifying side information features important for predicting the correctness of responses in an unsupervised manner.

Related Work

Research that considers differences in worker abilities while inferring the true labels for items dates back to the work of Dawid and Skene (1979), who integrated parameters modeling workers' abilities and biases into a single confusion matrix accommodating conditional probabilities of all possible response labels given all possible true labels. Since then, many models have been proposed to prevent the learning of the confusion matrix for each worker from over-fitting the corresponding labeled data. They have succeeded via either simplifying the confusion matrix setting by making it symmetric (Whitehill et al. 2009; Raykar et al. 2010b; Liu, Peng, and Ihler 2012; Wauthier and Jordan 2011), or grouping workers with similar confusion matrices together to smooth the worker-specific confusion matrices with the ones learned at the group-level (Venanzi et al. 2014; Moreno et al. 2014). Sometimes modeling worker abilities alone is not enough for accurate estimation of label quality, which might also be affected by variations in labeling accuracy across items. Accordingly, there has been research (Whitehill et al. 2009; Bachrach et al. 2012) taking into account another set of model parameters known as *item difficulty*, with others further considering the *multi-dimensional* interactions between the two entities (Welinder et al. 2010; Ruvolo, Whitehill, and Movellan 2013).

Among all the prior work, research investigating the use of side information to improve QCCs is limited. Kamar, Kapoor, and Horvitz (2015) studied utilizing observed side information, in particular features of the items, to estimate item-side confusion matrices to account for task-dependent biases. Kajino, Tsuboi, and Kashima (2012) developed convex optimization techniques using worker-specific classifiers centered on a base classifier which takes in item features for inferring their true labels. Ruvolo, Whitehill, and Movellan (2013) built a multi-dimensional logit model for predicting the correct label probability based on observed worker features. Ma et al. (2015) took into account "bag-of-word" information for learning the topical expertise of individual workers. Venanzi et al. (2016) considered response delay information to better distinguish spammers from genuine workers. We can see that each of these relevant works has focused on exploiting only one specific type of side information for improving the label quality control performance.

Proposed Framework

We are interested in developing a unified framework that is able to predict item true labels based on both worker responses and various types of side information about workers, items and contexts. The framework should be scalable for incorporating new types of side information. Table 1 summarizes the inputs, parameters and hyper-parameters of the proposed framework.

Basic Framework

Our proposed framework extends the GLAD (Whitehill et al. 2009) model which is shown in **Figure 1a**. GLAD applies a *logistic* function to the product between worker expertise and item difficulty variables to calculate the probability of correct labeling. More precisely, GLAD defines the probability of a response r_{uv} being correct, (i.e. equal to the true label l_v), as follows:

$$p(r_{uv} = l_v) = f^{\mathcal{H}}_{uv} = \frac{1}{1 + \exp(-z^{\mathcal{H}}_{uv})}$$
$$z^{\mathcal{H}}_{uv} = f^{\mathcal{H}_u}_{u} f^{\mathcal{H}_v}_{v} \qquad f^{\mathcal{H}_u}_{u} = e_u \qquad f^{\mathcal{H}_v}_{v} = \exp(d_v) \tag{1}$$

where $\mathcal{H} = \{e_u, d_v\}_{u \in \mathcal{U}, v \in \mathcal{V}}$ denotes the set of model parameters excluding the latent true labels $\mathcal{L}$ of items $\mathcal{V}$. According to GLAD, $e_u \in \mathbb{R}$ models the expertise of worker u, and $1/\exp(d_v)$ with $d_v \in \mathbb{R}$ models the difficulty of item v. For clarity, we call d_v the easiness of item v for the rest of the paper. Moreover, GLAD assumes normal priors over e_u

and d_v with μ_v, σ_v^2 and μ_u, σ_u^2 being their prior means and variances respectively. Whitehill et al. (2009) further simplify the GLAD model so that instead of needing to infer a $|K| \times |K|$ matrix of parameters for each worker as done by Dawid and Skene (1979), the model makes use of the following conditional statements:

$$p(r_{uv}|l_v) = f_{uv}^{\mathcal{H}} \text{ if } r_{uv} = l_v; \text{otherwise } = \frac{1 - f_{uv}^{\mathcal{H}}}{|\mathcal{K}| - 1} \quad (2)$$

We choose GLAD as the basis of our framework because of its factorization nature which easily allows for linear combination of different factors encoding arbitrary information about workers, items and their dyadic relations.

Incorporating Worker Information

We start with encoding information about workers into the basic framework as shown in **Figure 1b**. In this case, the worker-side expression $f_u^{\mathcal{H}_u}$ is changed to be:

$$f_u^{\mathcal{H}_u} = e_u + \boldsymbol{x}_u^T \boldsymbol{\beta}^{\mathcal{U}} \quad (3)$$

Here the dot product between the multi-dimensional feature vector $\boldsymbol{x}_u$ of worker u and the weight vector $\boldsymbol{\beta}^{\mathcal{U}}$ forms a global regression across all the workers with $\boldsymbol{\beta}^{\mathcal{U}}$ learned to bring the expertise offsets of similar workers closer together. This helps to smooth the irregular expertise estimates that result from the sparse labels across workers. Moreover, we assume a normal prior over the m-th component of $\boldsymbol{\beta}^{\mathcal{U}}$ with mean $\mu_m^{\mathcal{U}}$ and standard deviation $\sigma_m^{\mathcal{U}}$.

Incorporating Item Information

The item information is incorporated into the basic framework as shown in **Figure 1c**. The item-side expression $f_v^{\mathcal{H}_v}$ now has the following form:

$$f_v^{\mathcal{H}_v} = \exp(d_v + \boldsymbol{x}_v^T \boldsymbol{\beta}^{\mathcal{V}}) \quad (4)$$

where the dot product between the multi-dimensional feature vector $\boldsymbol{x}_v$ of item v and weight vector $\boldsymbol{\beta}^{\mathcal{V}}$ forms a global regression over all the items. $\boldsymbol{\beta}^{\mathcal{V}}$ serves the same purpose as its worker-side counterpart. We assume a normal prior over the m-th component of $\boldsymbol{\beta}^{\mathcal{V}}$ with mean $\mu_m^{\mathcal{V}}$ and standard deviation $\sigma_m^{\mathcal{V}}$.

Incorporating Response and Session Information

We consider contextual information at both the response level and the session level. The former type of contextual information is specific to each response given by a worker to an item, encoded by features including response delay and order, while the latter is specific to each labeling session of a worker which we define to start when a task page is loaded and to end once the page is submitted with no time-out in between. In this case, session features can include labeling devices (e.g. a personal computer), rendering browsers and the time periods (e.g. the day of the week) of the labeling. The session features capture much more variation in the labeling across different workers than within each worker as most of the workers remain situated in the same environments throughout the entire crowd-sourcing tasks. In contrast, the response features should provide much more insight into the variation within each worker's labeling behavior.

Symbols	Description
Inputs	
$\mathcal{U}$	set of workers
$\mathcal{V}$	set of items
$\mathcal{K}$	set of label categories
$r_{uv} \in \mathcal{K}$	response of worker u for item v
$\mathcal{R}$	set of responses $\{r_{uv}\|(u \in \mathcal{U}) \wedge (v \in \mathcal{V})\}$
$\boldsymbol{x}_u$	feature vectors for worker u
$\boldsymbol{x}_v$	feature vectors for item v
$\boldsymbol{x}_{us}$	feature vectors for a labeling session s of worker u
$\boldsymbol{x}_{uv}$	feature vectors for the response given by worker u to item v
Parameters	
$\mathcal{L} = \{l_v\}_{v \in \mathcal{V}}$	set of latent true label variables for items in $\mathcal{V}$
$\mathcal{H}$	set of model parameters excluding $\mathcal{L}$
$\boldsymbol{\theta}$	probability vector over item true label categories
e_u	expertise variable $e_u \in (-\infty, +\infty)$ of worker u
d_v	"easiness" variable $d_v \in [0, +\infty)$ of item v
$\boldsymbol{\beta}^{\mathcal{U}}$	weight vector for worker features $\boldsymbol{x}_u$
$\boldsymbol{\beta}^{\mathcal{V}}$	weight vector for item features $\boldsymbol{x}_v$
$\boldsymbol{\beta}^{\mathcal{S}}$	weight vector for session features $\boldsymbol{x}_{us}$
$\boldsymbol{\alpha}_u$	weight vector for the responses given by worker u
Hyper-parameters	
$\mathcal{H}_0$	set of hyper-parameters for the model parameters
$\boldsymbol{\gamma}$	Dirichlet prior for item true label l_v
μ_u, σ_u	Normal prior for e_u
μ_v, σ_v	Normal prior for d_v
$\mu^{\mathcal{U}}, \sigma^{\mathcal{U}}$	Normal prior for each weight component of $\boldsymbol{\beta}^{\mathcal{U}}$
$\mu^{\mathcal{V}}, \sigma^{\mathcal{V}}$	Normal prior for each weight component of $\boldsymbol{\beta}^{\mathcal{V}}$
$\mu^{\mathcal{S}}, \sigma^{\mathcal{S}}$	Normal prior for each weight component of $\boldsymbol{\beta}^{\mathcal{S}}$
$\mu_\alpha, \sigma_\alpha$	Normal prior for each weight component of $\boldsymbol{\alpha}_u$

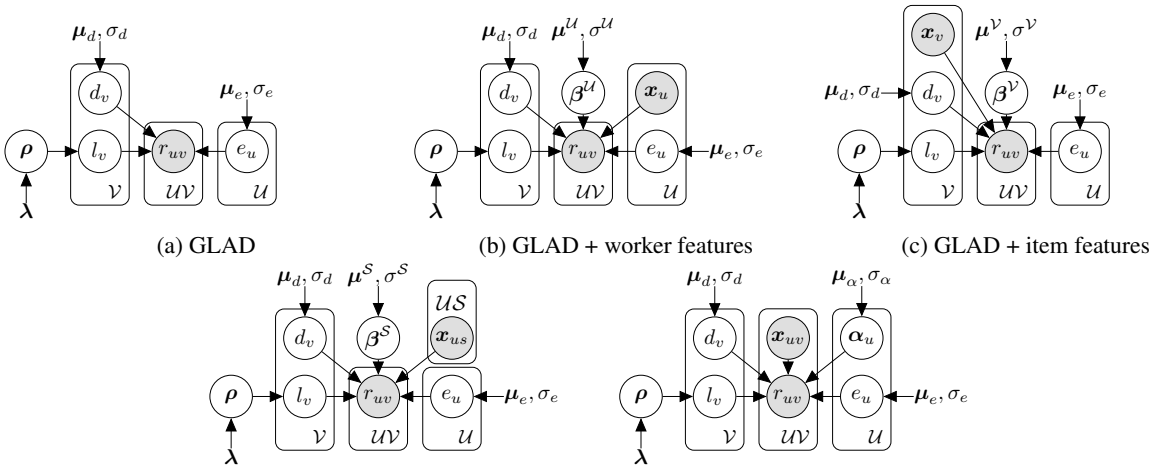
Table 1: List of Notation used.

To leverage the advantages of both types of contextual information, we incorporate them into the basic framework as shown in **Figures 1d** and **1e**, where $\mathcal{S}$ is the set of labeling sessions each corresponding to a task page containing a number of label-collecting questions.

The corresponding changes made to the worker-side expression and to the entire expression are respectively the following:

$$\begin{aligned} f_u^{\mathcal{H}_u} &= e_u + \boldsymbol{x}_{us}^T \boldsymbol{\beta}^{\mathcal{S}} \\ z_{uv}^{\mathcal{H}} &= f_u^{\mathcal{H}_u} f_v^{\mathcal{H}_v} + \boldsymbol{x}_{uv}^T \boldsymbol{\alpha}_u \end{aligned} \quad (5)$$

Here the dot product between the multi-dimensional feature vector $\boldsymbol{x}_{us}$ of worker u within session s and the weight vector $\boldsymbol{\beta}^{\mathcal{S}}$ forms a global regression over all the sessions of all the workers, while $\boldsymbol{\alpha}_u$ are specific to worker u, serving as the weight vector of a local linear regression over the feature vectors of all the responses made by worker u. Such local regressions aim at addressing worker-specific biases that GLAD fails to handle properly (Welinder et al. 2010). We again assume normal priors over the m-th component of both $\boldsymbol{\beta}^{\mathcal{S}}$ and $\boldsymbol{\alpha}_u$ with means $\mu_m^{\mathcal{S}}, \mu_{mu}$ and standard deviations $\sigma_m^{\mathcal{S}}, \sigma_{mu}$, respectively.



(a) GLAD (b) GLAD + worker features (c) GLAD + item features

(d) GLAD + labeling session features (e) GLAD + response features

Figure 1: (a), (b), (c), (d) and (e) show the GLAD model, and the GLAD model with the observed features about workers, items, sessions and responses respectively.

Inference

In this section, we describe a stochastic parameter estimator to obtain posterior probabilities of the latent true label variables $\mathcal{L}$. More specifically, in each iteration of the estimation, we alternate between the Collapsed Gibbs sampling (Griffiths and Steyvers 2004) for $\mathcal{L}$ given the current estimates of the model parameters $\mathcal{H}$, and the *one-step* gradient descent for updating $\mathcal{H}$ given the sampled $\mathcal{L}$.

Collapsed Gibbs Sampling for $\mathcal{L}$

At this stage, we employ a collapsed Gibbs sampler to obtain posterior samples for $\mathcal{L} = \{l_v\}_{v \in \mathcal{V}}$ given the current estimates of $\mathcal{H}$. In this case, the conditional probabilities of true label l_v is obtained by marginalizing out the multinomial probability vector $\boldsymbol{\theta}$, which ends up being:

$$P(l_v = k | \mathcal{L}_{\setminus v}, \mathcal{R}_v, \mathcal{H}, \boldsymbol{\gamma}) \propto \frac{N_{\setminus vk} + \gamma_k}{\sum_{j \in \mathcal{K}} (N_{\setminus vj} + \gamma_j)} \times \prod_{u \in \mathcal{U}_v} \left((f^{\mathcal{H}}_{uv})^{\mathbb{1}\{r_{uv} = l_v\}} \left(\frac{1 - f^{\mathcal{H}}_{uv}}{|\mathcal{K}| - 1} \right)^{\mathbb{1}\{r_{uv} \neq l_v\}} \right) \quad (6)$$

where $\mathcal{U}_v$ is the set of workers who responded item v with a set of responses $\mathcal{R}_v$, $\mathcal{L}_{\setminus v}$ is the set of current true label assignments to all the items excluding item v, and $N_{\setminus vk}$ is the number of items excluding v whose true labels are currently inferred to be k.

Gradient Descent for $\mathcal{H}$

The conditional probability distributions of the model parameters $\mathcal{H} = (\boldsymbol{\theta}, \boldsymbol{e}, \boldsymbol{d}, \boldsymbol{\beta}^{\mathcal{U}}, \boldsymbol{\beta}^{\mathcal{V}}, \boldsymbol{\beta}^{\mathcal{S}}, \boldsymbol{\alpha})$ are hard to compute analytically due to the presence of the logistic function. Instead, we run the gradient descent for one step with respect to $\mathcal{H}$ on the negative logarithm of its joint conditional probability distribution.

$$\mathcal{Q}(\mathcal{H}) = -\sum_{v \in \mathcal{V}} \sum_{u \in \mathcal{U}_v} \log \left((f^{\mathcal{H}}_{uv})^{\mathbb{1}\{r_{uv} = l_v\}} \times \left(\frac{1 - f^{\mathcal{H}}_{uv}}{|\mathcal{K}| - 1} \right)^{\mathbb{1}\{r_{uv} \neq l_v\}} \right) - \log p(\mathcal{H} | \mathcal{H}_0). \quad (7)$$

The first term of Equation 7 is the negative log-likelihood of the response data, and the second term is the log-prior with $\mathcal{H}_0$ denoting the set of hyper-parameters of $\mathcal{H}$. To minimize $\mathcal{Q}(\mathcal{H})$, we take partial derivative of Equation 7 with respect to each element in $\mathcal{H}$.

Estimate e_u and d_v In this case, the relevant prior terms in $\log p(\mathcal{H}|\mathcal{H}_0)$ are $\frac{(d_v - \mu_v)^2}{2\sigma_v^2} + \frac{(e_u - \mu_u)^2}{2\sigma_u^2}$. The gradients of e_u and d_v are thus:

$$\frac{\partial \mathcal{Q}}{\partial d_v} = -\sum_{u \in \mathcal{U}_v} \left(\delta_{uv} f^{\mathcal{H}_u}_u f^{\mathcal{H}_v}_v \right) + \frac{d_v - \mu_v}{\sigma_v^2} \quad (8)$$

$$\frac{\partial \mathcal{Q}}{\partial e_u} = -\sum_{v \in \mathcal{V}_u} \left(\delta_{uv} f^{\mathcal{H}_v}_v \right) + \frac{e_u - \mu_u}{\sigma_u^2}, \quad (9)$$

where $\delta_{uv} = [\mathbb{1}\{r_{uv} = l_v\}(1 - f^{\mathcal{H}}_{uv}) - \mathbb{1}\{r_{uv} \neq l_v\} f^{\mathcal{H}}_{uv}]$, and $\mathcal{V}_u$ is the set of items responded by worker u.

Estimate $\beta^{\mathcal{U}}_m$ and $\beta^{\mathcal{V}}_m$ Taking derivatives w.r.t. $\beta^{\mathcal{U}}_m$ and $\beta^{\mathcal{V}}_m$, the m-th components of $\boldsymbol{\beta}^{\mathcal{U}}$ and $\boldsymbol{\beta}^{\mathcal{V}}$ respectively, yields similar equations:

$$\frac{\partial \mathcal{Q}}{\partial \beta^{\mathcal{U}}_m} = -\sum_{v \in \mathcal{V}} \sum_{u \in \mathcal{U}_v} \left(\delta_{uv} f^{\mathcal{H}_v}_v x_{mu} \right) + \frac{\beta^{\mathcal{U}}_m - \mu^{\mathcal{U}}}{\sigma^{\mathcal{U}^2}} \quad (10)$$

$$\frac{\partial \mathcal{Q}}{\partial \beta^{\mathcal{V}}_m} = -\sum_{v \in \mathcal{V}} \sum_{u \in \mathcal{U}_v} \left(\delta_{uv} f^{\mathcal{H}_u}_u f^{\mathcal{H}_v}_v x_{mv} \right) + \frac{\beta^{\mathcal{V}}_m - \mu^{\mathcal{V}}}{\sigma^{\mathcal{V}^2}} \quad (11)$$

Equation 10 is also applied to $\frac{\partial \mathcal{Q}}{\partial \beta^{\mathcal{S}}_m}$ with x_{mus} and $\beta^{\mathcal{S}}_m$ respectively replacing x_{mu} and $\beta^{\mathcal{U}}_m$.

Estimate α_{mu} The gradient w.r.t. α_{mu}, the m-th component of $\boldsymbol{\alpha}_u$, is given by:

$$\frac{\partial \mathcal{Q}}{\partial \alpha_{mu}} = -\sum_{v \in \mathcal{V}_u} \left(\delta_{uv} x_{uv} \right) + \frac{\alpha_{mu} - \mu_\alpha}{\sigma_\alpha^2} \quad (12)$$

Experiments

We present experiments that study the performance of our framework on combining different types of side information to improve the quality control for crowd-sourcing on three real-world datasets.

Datasets

The three datasets were collected in separate crowd-sourcing tasks on CrowdFlower with three responses collected for each item. As a basic quality control measure, we filtered out workers who did not achieve 88% accuracy on pre-defined control questions. The qualified workers were also asked for additional information including demographics and personal traits[3]. Each qualified worker is allowed to label a certain number of items for each task and is free to quit labeling at any time. Nineteen items were randomly selected and shown to the a worker on each task page. Table 2 provides a summary of the three datasets.

Dataset	# Worker	# Item	# Response
Stack Overflow	505	14,021	42,063
Evergreen Webpage	434	7,336	22,008
TREC 2011	160	1826	5,478

Table 2: Dataset Summary.

TREC 2011 Crowd-sourcing Track Each CrowdFlower worker was asked to judge the relevance levels (i.e. *highly relevant*, *relevant* and *non-relevant*) of 38 Web-pages to their corresponding queries in the TREC 2011 Crowd-sourcing Track dataset[4]. **Figure 2a** shows a question for collecting the relevance judgments for a pair of query (i.e. "french lick resort and casino") and document.

Stack Overflow Post Status Judgement Each CrowdFlower worker was asked to judge the status of 95 archived questions from Stack Overflow[5]. The status of a question can be either *open*, meaning it is regarded suitable to stay active (i.e. visible, answerable and editable) on Stack Overflow, or *closed*, meaning the opposite for reasons including that it is *not a real question* (i.e. questions that are ambiguous, too broad or "show no efforts" in seeking answers), *not constructive* to the Website (i.e. questions that are subjective and have no correct answers) and *too localized* (i.e. questions that are not reproducible, thus useless to other workers in the future). **Figure 2b** shows a question for collecting the status judgments for a Stack Overflow post.

Evergreen Webpage Judgment Each CrowdFlower worker was asked to judge 57 Web-pages[6] on whether they think these Web-pages have a timeless quality or, in other words, will still be considered by average workers as valuable or relevant in the future.

Side Info.	Dataset: Stack-Overflow	Dataset: Evergreen	Dataset: TREC
Worker	1. *age*, 2. *gender*, 3. *education*, 4. *personality traits*		
	self-appraisal about: 1. *programming experience* 2. *frequency of search/post/ edit/answer -ing questions* 3. *diversity of questions dealt with*	1. *mother tongue* **self-appraisal about:** 2. *frequency of online search* 3. *frequency of bookmarking Web-pages* 4. *frequency of revisiting Web-pages*	1. *mother tongue* **self-appraisal about:** 2. *frequency of online search* 3. *diversity of online search topics* 4. *average # SERPs checked*
Item	1. *content length*, 2. *item genre*		
	-	1. *Web-page features*	-
Response	1. *response time/delays*, 2. *response order*		
Session	1. *weekends or weekdays*, 2. *time of the day* 3. *labeling devices (e.g. PC, tablets, etc.)*		

Table 3: features encoding different types of information.

Feature Collection

We collected various types of side-information as summarized in Table 3.

Worker Feature The worker features include both the demographic and personality trait features which are common to all the three tasks, and the "self-appraisal" features which vary from task to task. The ages of workers (starting from 18) were discretized into 9 groups with the first 8 groups each having a 5-year gap onward and the last being of age 60 or over. The education backgrounds of workers were divided into 5 categories from "Less than high school" to "Master degree or above". To collect information about the personality traits of workers, we directly employed the 10 survey questions used by Kazai, Kamps, and Milic-Frayling (2012) based on the so called five personality trait dimensions (John, Naumann, and Soto 2008). As for the "self-appraisal" features, these were designed to capture the possible nuances in workers' expertise levels on different tasks from their own perspectives.

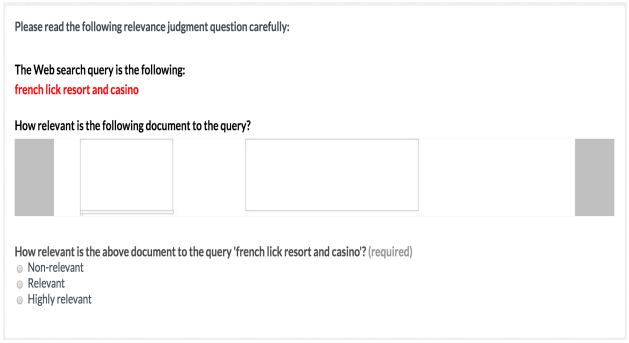
Item Feature The item features include both the common features which are the content length and item genres (already known for each item of each dataset), and those unique

[3]This was done by mixing the demographic survey questions with the control questions in the quiz for the workers to answer prior to starting the crowd-sourcing task. As a result, there exist a small number of missing values in the demographic data collected as not all the survey questions were chosen by CrowdFlower to appear on the quiz page.

[4]https://sites.google.com/site/treccrowd/2011

[5]The set of questions judged is a random subset of the training dataset used in the Kaggle competition "*Predict Closed Questions on Stack Overflow*" (https://www.kaggle.com/c/predict-closed-questions-on-stack-overflow).

[6]We used the training set from the Kaggle Competition "*StumbleUpon Evergreen Classification Challenge*" https://www.kaggle.com/c/stumbleupon.



(a) a question for relevance judgment

(b) a question for Stack Overflow post status judgment

to the Evergreen dataset which are the original Web-page features[7] provided in the Kaggle competition.

Response and Session Features The contextual features were collected at both the response and the session levels, and were the same across all the tasks. Response-level feature "response delay" records the amount of time each worker took to label each item. Its value was calculated by subtracting the click time of the previous item (or the page load time if it was more recent) from the click time of the particular item. We also computed the "response order" of each item by ordering items by their "last click time". As for the session feature "time of the day", we set its value, either "daytime", "night" or 'late night", corresponding to the periods [6am, 7pm), [7pm, 23pm), and [23pm, 6am), respectively.

Feature Normalization The pre-processing of the features involved binarizing the non-numeric features, and normalizing the numeric features using a Z-score transformation, (except for the numeric feature "response order" for which we used a min-max normalization as its values were always uniformly distributed). For numeric features with highly skewed empirical distributions, a log-transformation[8] was applied prior to normalizing with the Z-score transformation. Finally, we normalized the response-level feature "response delay" on a per-worker basis (i.e. using worker specific mean and standard deviation values), in order to facilitate local linear regression with the worker-specific weight vector α_u as specified in Equation 5.

Experiment Setup

Baselines We verify the efficacy of our model by comparing it with the following three baselines:

- *Majority Vote*: the predicted true label for an item is the response given by the majority of the workers.
- *GLAD*: the probability of a correct response is a logistic function over the product between the worker expertise and the item "easiness" variables.
- *Community-based Dawid-Skene* (DS) (Venanzi et al. 2014): to smooth out unreliable estimates of the confusion matrix entries due to label sparsity, the matrix is drawn (row-wise) from one that is shared by a *community* to which the worker is inferred to belong.

Evaluation Metrics We use the following metrics to evaluate the performance of the baseline methods and our proposed framework in terms of the item true label prediction:

- *Predictive accuracy (Accu)*:
  $\frac{1}{|\mathcal{V}|}\sum_{v\in\mathcal{V}} \mathbb{1}\{l_v = \hat{l}_v\}$
  where $\hat{l}_v = \arg\max_{l\in\mathcal{K}} P(l_v = l|\mathcal{R}, \textit{Model})$
- *Log-Loss (Log)*:
  $-\frac{1}{|\mathcal{V}|}\sum_{v\in\mathcal{V}}\sum_{l\in\mathcal{K}} \mathbb{1}\{l_v = l\}\log\left(P(l_v = l|\mathcal{R}, \textit{Model})\right)$

We use the following metric to evaluate the performance of the baselines nd our proposed framework with respect to the unseen worker response prediction.

- *Mean Absolute Error (MAE)*:
  $\frac{1}{|\mathcal{R}_h|}\sum_{r_{uv}\in\mathcal{R}_h} \mathbb{1}(r_{uv} \neq \hat{r}_{uv})$
  where $\hat{r}_{uv} = \arg\max_{l\in\mathcal{K}} P(r_{uv} = l|\mathcal{R}_{\setminus h}, \textit{Model})$

The first measure (Accu) tells us the performance in terms of the number of correctly predicted true labels. The second measure (Log) tells us how confident the model is in its predictions of the true labels. The second measure is often more sensitive than the first and can therefore be useful for comparing similarly performing models. The third measure (MAE) informs us of the performance with respect to the average number of correctly predicted held-out responses $\mathcal{R}_h$ from the workers based on the posterior mode $\hat{r}_{uv}$ given the training responses $\mathcal{R}_{\setminus h}$ and the specific model being tested. In the experiment using the entire responses, the proportion of the held-out test data $\mathcal{R}_h$ is set to be 30% and the accuracy of the unseen worker response prediction of a model is obtained by averaging its performance over 10 such held-out tests.

Hyper-parameter Setup For GLAD, Whitehill et al. (2009) imposes normal priors over both e_u and d_v with means and standard deviations both set to be 1. This setting assumes that *a priori* approximately 75% of workers are reliable and 75% of items are relatively easy for average workers. Although straightforward, we believe that the

[7] https://www.kaggle.com/c/stumbleupon/data

[8] $\log(c + x)$ where $c = min(0.1, min(x))$

above hyper-parameter settings for GLAD is not particularly effective for modeling common crowd-sourcing scenarios for two main reasons.

First, crowd-sourcing tasks conducted on CrowdFlower can only be attempted, by default, by *leveled* [9] (in other words, experienced and high-quality) workers. We adopted this default setting for all our tasks. Moreover, control questions are very often used to vet and remove low-performing workers before and during the tasks. Thus, we believe that *a priori* no workers should be considered unreliable (that is e_u below or close to 0). Instead, all workers should be expected to have abilities close to the prior mean.

Secondly, it is common to collect only a small number of responses for each item during crowd-sourcing. Such few responses can hardly provide sufficient information for GLAD to reliably estimate d_v. Moreover, GLAD applies an exponential transformation on d_v to ensure its non-negativity, which is likely to further "inflate" its inaccurate estimation. As an example, when $d_v = 3$ (i.e. two positive standard deviations from its prior mean), $\exp(d_v) = 20.1$, much larger than $e_u = 3$ which is also at two positive standard deviations. As a result, the log-odds of correct labeling, modeled as their product in GLAD, is likely to be dominated by the potentially inaccurate estimates of d_v. Thus, we suggest a stronger regularization penalty for d_v to suppress these problems. Based on the above analysis, we set the prior means for e_u and d_v to be 2 and 0 respectively. Note that assigning the latter to zero imposes an uninformative setting for the prior of d_v. We set the prior variance for both to be 0.1 in accordance with the arguments above (to limit the variance of e_u and to increase the regularization on d_v).

Each component of the Dirichlet prior vector γ is set to be 1. For each regression weight vector (e.g. $\boldsymbol{\beta}^{\mathcal{U}}$), we set the prior means of its components to be 0, and the prior standard deviations to be (0.01 / #features) so that the influence brought by each global/local regression is comparable to that brought by its affecting factor (e.g. e_u). As for the gradient descent step size, we define a default step size of $\eta = 0.001$ and calculate a parameter-specific size based on the number of data instances available for estimating the parameter, that is $(\eta/\#datapoints)$. Discounting η is necessary as parameters are estimated at different (global or local) levels in our framework. Except for the Dirichlet prior for the class proportions to be fixed all at 1, the other hyper-parameters of the community-based DS, including the number of communities, is set through 10-fold cross-validaton repeated and averaged over 5 times, evaluated upon the likelihood of the validation responses.

Prediction with Subsampled Responses While the default evaluation task is to check the efficacy of our framework under item-side label sparsity as only 3 responses were collected for each item, we would like to further investigate whether the framework can handle even greater degrees of label sparsity which happens not only on the item side but also on the worker side. To do this, we randomly subsampled the same number of responses from each worker. By merging all the subsampled responses from each worker, we obtained a data subset with far fewer items for each of the three datasets. We varied the number of responses subsampled per worker from 1 to 12 (after which we actually observed very marginal differences in the model performence), and ran all the models for the true label prediction as well as the unseen (held-out) worker response prediction at each subsampling point. The unseen response prediction is evaluated on the remaining responses. Both prediction tasks are evaluated using the same metrics before as used in the experiments with the full responses. The whole subsampling procedure was repeated 5 times before we obtained the average predictive accuracy of each model. The hyper-parameter setup in this case remained unchanged as we employed the same hyper-parameter setting for our framework and GLAD, and the 10-fold cross-validation for finding the optimal number of communities.

	Stackoverflow		Evergreen		TREC	
	Accu	Log	Accu	Log	Accu	Log
MV	0.6083	0.9748	0.7630	0.6635	0.4720	1.124
4-Community DS	0.6088	0.9292	0.7633	0.6248	0.4818	1.112
GLAD	0.6084	0.9323	0.7631	0.6385	0.4747	1.126
GLAD+I	0.6085	0.9306	0.7632	0.6280	0.4765	1.122
GLAD+L	0.6084	0.9311	0.7631	0.6296	0.4751	1.126
GLAD+R	0.6087	0.9294	0.7633	0.6271	0.4766	1.122
GLAD+S	0.6085	0.9306	0.7631	0.6292	0.4760	1.123
GLAD+I+L+R	0.6088	0.9294	0.7633	0.6256	0.4808	1.116
GLAD+I+L+S	0.6087	0.9301	0.7633	0.6252	0.4804	1.116
GLAD+I+R+S	**0.6090**	0.9289	**0.7634**	0.6245	0.4819	1.112
GLAD+L+R+S	0.6088	0.9298	0.7633	0.6258	0.4802	1.118
GLAD+I+L+R+S	**0.6090**	**0.9288**	**0.7634**	**0.6238**	**0.4820**	**1.108**

Table 4: True label predictive accuracy of the models across the three datasets. We denote the side information about the workers, the items, the sessions and the responses respectively with capital letters "L", "I", "S" and "R".

	Stackoverflow	Evergreen	TREC
	MAE	MAE	MAE
4-Community DS	0.2306	0.1858	0.5176
GLAD	0.2438	0.1901	0.5216
GLAD+I	0.2347	0.1874	0.5176
GLAD+L	0.2396	0.1898	0.5195
GLAD+R	0.2288	0.1854	0.5173
GLAD+S	0.2356	0.1901	0.5211
GLAD+I+L+R	0.2245	0.1801	0.5162
GLAD+I+L+S	0.2325	0.1854	0.5187
GLAD+I+R+S	0.2276	0.1812	0.5169
GLAD+L+R+S	0.2318	0.1860	0.5184
GLAD+I+L+R+S	**0.2226**	**0.1801**	**0.5160**

Table 5: Unseen (held-out) response predictive accuracy of the models across 30% held-out response data from the three datasets.

[9] http://crowdflowercommunity.tumblr.com/post/80598014542/introducing-contributor-performance-levels

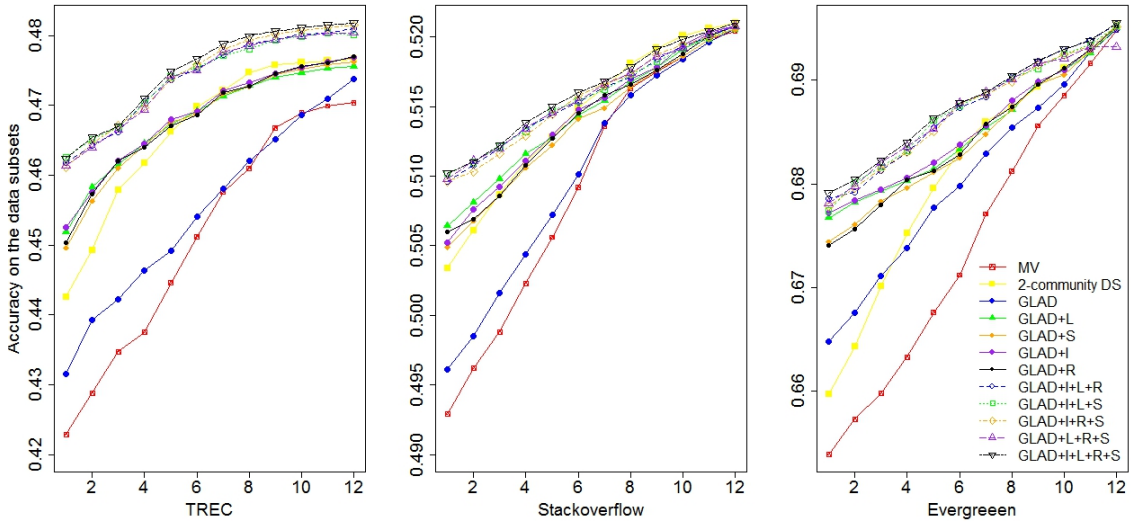


Figure 3: Changes of the true label predictive accuracy by varying the number of responses subsampled from each worker across the three datasets.

Results

The results of the experiment using the entire response data are summarized in Tables 4 and 5. For clarity, the side information of the workers, the items, the sessions and the responses is abbreviated to "L", "I", "S" and "R" respectively. From Table 4, our framework with all the features outperforms the three baselines (with the comunity-based DS optimized at 4 communities) on all the datasets. The largest improvement in predictive accuracy is seen over Majority Vote (MV) on the TREC dataset (i.e. by 1%). Marginal improvements in accuracy have been observed over the three baselines on the Stackoverflow and the Evergreen datasets. We believe the reason for observing only marginal improvements on these datasets is that all workers have exhibited similar ability for these tasks, producing labels of similar quality across the items.

We note from Table 4 that incorporating the observed features about items appears to produce a larger reduction in the log-loss than that is achieved by adding worker or session features. This is in line with our expectation of the first experiment which is that the item features help to reduce the uncertainty in the item easiness d_v, which likely has suffered from label sparsity across the three crowd-sourcing tasks with only three labels collected for each item . In contrast, reduction in the uncertainty of worker expertise e_u, attributed to the addition of the worker and the session features into our framework, appears far less beneficial given that there is already abundant response data available for the estimation. Moreover, it appears that incorporating response information brings systematic improvements in both accuracy and log-loss. The result confirms that our framework is able to consistently utilize such information to mitigate the bias specific to each worker.

From Table 5, we can see that when equipped with all the side information features, our framework again defeats the baseline models GLAD and community-based DS (the majority vote intrinsically not suitable for predicting worker responses) with the minimum mean absolute error (highlighted in bold font) for the 30% held-out response data across all the datasets.

The results of the experiment using the randomly subsampled response data from workers are summarized in Figures 3 and 4. We observe from Figure 3 that when the number of responses subsampled per worker is below 6, our framework has significantly outperformed GLAD and Majority Vote aross all the three datasets by leveraging only one type of side information. When the number is 12, our framework with combined types of side information still distinctly exceeds the performance of the two baselines over the TREC dataset. The community-based DS, whose optimal number of community is 2 in this case, is clearly beaten by our framework with (1) a single side-information type over the Evergreen dataset when the number of subsampled responses is below 4, and (2) the combined types over the TREC and the Stackoverflow datasets when the number is below 3.

When applied to the task of predicting unseen (held-out) responses of workers as shown in Figure 4, our framework still holds clear advantages over the GLAD model by leveraging only single types of side information, and over the community-based DS model (with its optimal number of communities being 2) by leveraging multiple (in our experiment at least 3) types of side information.

Overall, our framework has made much more significant improvements over the baselines when the items are far fewer and the response data are far more scarce, compared to its performance in the previous experiments based on the entire responses.

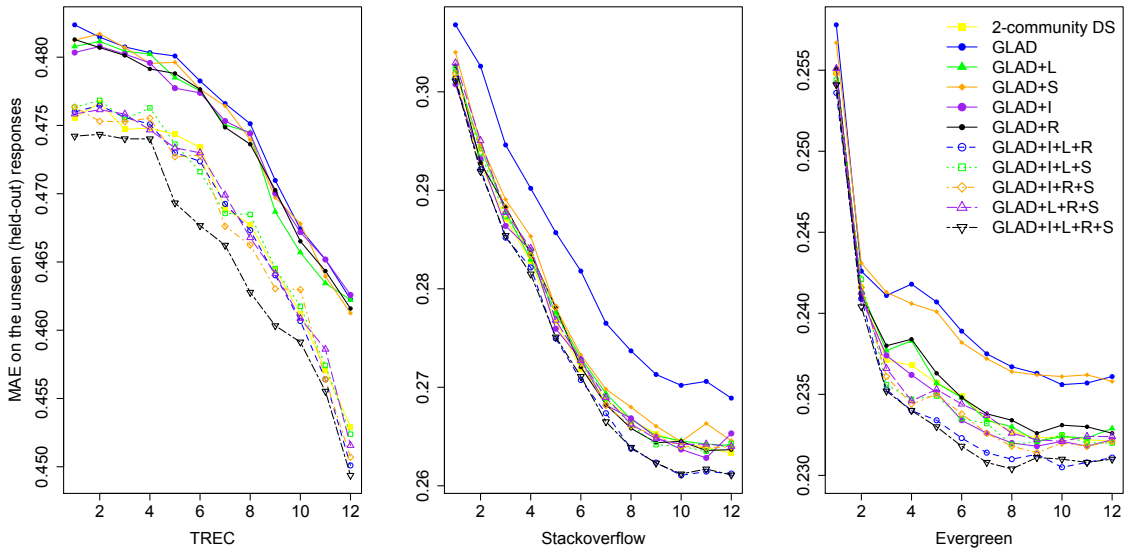


Figure 4: Changes of the unseen (held-out) response predictive accuracy by varying the number of responses subsampled from each worker across the three datasets.

Statistical Analysis of Feature Importance

To get a clear idea of which features might be important for predicting the accuracy of the responses, we conducted one-way ANOVA where the resulting P-values indicate the significance of the correlation between the features and the correct responses (Kazai, Kamps, and Milic-Frayling 2013; Li, Zhao, and Fuxman 2014). We compared the P-values with the feature weight estimates from our framework with all the types of side information. We listed Top-5 features with respect to the P-values and the absolute feature weight values inferred from the Evergreen dataset in Table 6. We also obtained similar results with Stack-Overflow and TREC datasets. Although our framework works in a fully unsupervised manner whereas one-way ANOVA is supervised, the results show that our framework is equally capable of identifying salient features for predicting the accuracy of each response.

	Top 5 features ranked according to:	
	Supervised Prediction (P-value)	Unsupervised Prediction (weight)
I	1. num_alphaNumeric_chars 2. num_links 3. domain_business 4. content_length 5. frame_tag_ratio	1. num_links 2. num_alphaNumeric_chars 3. content_length 4. frame_tag_ratio 5. domain_business
L	1. searchOnline_sometimes 2. artist_agreeModerately 3. revisit_topic_diversity_level3 4. education_Bachelor 5. search_topic_diversity_level3	1. searchOnline_veryOften 2. lazy_slightlyAgree 3. revisit_topic_diversity_level3 4. thorough_stronglyAgree 5. age_45_to_49
S	1. use_mobilePhone, 2. use_PC 3. use_tablet, 4. weekends, 5. latenight	1. use_mobilePhone, 2. use_tablet 3. use_PC, 4. weekends, 5. latenight

Table 6: Comparison between top 5 most predictive features for supervised and unsupervised (actual) setting.

Conclusions and Future Work

In this paper we have developed a probabilistic framework for improving the quality control over crowd-sourced data by leveraging its associated side information from items, workers, labeling sessions and responses. The respective source features include item genres and content features, worker demographics and personality traits, labeling devices and labeling time periods, response delays and response orders. The efficacy of the framework has been demonstrated on three new crowd-sourcing datasets, where we have observed overall consistent improvements in predictive accuracy and log-loss, as well as in unseen (held-out) response prediction, when the response data is scarce across both workers and items. Moreover, response-level information was found particularly useful for helping the framework to account for worker-specific biases. In addition, our framework is found to be promising at identifying salient source features without having to know any true-label information.

Future work includes extending the current framework with the matrix factorization component to allow for domain-specific expertise to be estimated for each worker and the level of domain membership to be estimated for each item.

Acknowledgments

. The authors thank Professor Wray Buntine for useful discussions regarding this work. Mark Carman acknowledges research funding through a Collaborative Research Project award from CSIRO's Data61.

References

Bachrach, Y.; Graepel, T.; Minka, T.; and Guiver, J. 2012. How to grade a test without knowing the answers—a bayesian graphical model for adaptive crowdsourcing and aptitude testing. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, 1183–1190.

Dawid, A. P., and Skene, A. M. 1979. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 28(1):pp. 20–28.

Griffiths, T. L., and Steyvers, M. 2004. Finding scientific topics. *Proceedings of the National academy of Sciences* 101(suppl 1):5228–5235.

John, O. P.; Naumann, L. P.; and Soto, C. J. 2008. Paradigm shift to the integrative big five trait taxonomy. *Handbook of personality: Theory and research* 3:114–158.

Jung, H. J., and Lease, M. 2012. Improving quality of crowdsourced labels via probabilistic matrix factorization. In *Proceedings of the 4th Human Computation Workshop (HCOMP) at AAAI*, 101–106.

Kajino, H.; Tsuboi, Y.; and Kashima, H. 2012. A convex formulation for learning from crowds. In *AAAI*.

Kamar, E.; Kapoor, A.; and Horvitz, E. 2015. Identifying and accounting for task-dependent bias in crowdsourcing. In *Third AAAI Conference on Human Computation and Crowdsourcing*.

Kazai, G.; Kamps, J.; and Milic-Frayling, N. 2012. The face of quality in crowdsourcing relevance labels: Demographics, personality and labeling accuracy. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, 2583–2586. ACM.

Kazai, G.; Kamps, J.; and Milic-Frayling, N. 2013. An analysis of human factors and label accuracy in crowdsourcing relevance judgments. *Information retrieval* 16(2):138–178.

Lease, M. 2011. On quality control and machine learning in crowdsourcing.

Li, H.; Zhao, B.; and Fuxman, A. 2014. The wisdom of minority: discovering and targeting the right group of workers for crowdsourcing. In *Proceedings of the 23rd international conference on World wide web*, 165–176. ACM.

Liu, Q.; Peng, J.; and Ihler, A. T. 2012. Variational inference for crowdsourcing. In *Advances in Neural Information Processing Systems*, 692–700.

Ma, F.; Li, Y.; Li, Q.; Qiu, M.; Gao, J.; Zhi, S.; Su, L.; Zhao, B.; Ji, H.; and Han, J. 2015. Faitcrowd: Fine grained truth discovery for crowdsourced data aggregation. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 745–754. ACM.

Moreno, P. G.; Teh, Y. W.; Perez-Cruz, F.; and Artés-Rodríguez, A. 2014. Bayesian nonparametric crowdsourcing. *arXiv preprint arXiv:1407.5017*.

Rasch, G. 1960. *Probabilistic Models for Some Intelligence and Attainment Tests.* MESA Press.

Raykar, V. C.; Yu, S.; Zhao, L. H.; Valadez, G. H.; Florin, C.; Bogoni, L.; and Moy, L. 2010a. Learning from crowds. *Journal of Machine Learning Research* 11:1297–1322.

Raykar, V. C.; Yu, S.; Zhao, L. H.; Valadez, G. H.; Florin, C.; Bogoni, L.; and Moy, L. 2010b. Learning from crowds. *Journal of Machine Learning Research* 11(Apr):1297–1322.

Ruvolo, P.; Whitehill, J.; and Movellan, J. R. 2013. Exploiting commonality and interaction effects in crowdsourcing tasks using latent factor models. In *Neural Information Processing Systems. Workshop on Crowdsourcing: Theory, Algorithms and Applications*.

Sheng, V. S.; Provost, F.; and Ipeirotis, P. G. 2008. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, 614–622. New York, NY, USA: ACM.

Venanzi, M.; Guiver, J.; Kazai, G.; Kohli, P.; and Shokouhi, M. 2014. Community-based bayesian aggregation models for crowdsourcing. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, 155–164. New York, NY, USA: ACM.

Venanzi, M.; Guiver, J.; Kohli, P.; and Jennings, N. R. 2016. Time-sensitive bayesian information aggregation for crowdsourcing systems. *Journal of Artificial Intelligence Research* 56:517–545.

Wauthier, F. L., and Jordan, M. I. 2011. Bayesian bias mitigation for crowdsourcing. In *25th Annual Conference on Neural Information Processing Systems 2011*, 1800–1808.

Welinder, P.; Branson, S.; Perona, P.; and Belongie, S. J. 2010. The multidimensional wisdom of crowds. In *Advances in neural information processing systems*, 2424–2432.

Whitehill, J.; Ruvolo, P.; Wu, T.; Bergsma, J.; and Movellan, J. R. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *23rd Annual Conference on Neural Information Processing Systems, NIPS'09*, 2035–2043.