A Framework for Quality Assurance in Crowdsourcing

Jing Wang*[1], Panagiotis G. Ipeirotis†[2], and Foster Provost‡[2]

[1]School of Business and Management, Hong Kong University of Science and Technology

[2]Leonard Stern School of Business, New York University

Abstract

The emergence of online *paid micro-crowdsourcing* platforms, such as Amazon Mechanical Turk (AMT), allows on-demand and at scale distribution of tasks to human workers around the world. In such settings, online workers come and complete small tasks posted by a company, working for as long or as little as they wish. Such temporary employer-employee relationships give rise to adverse selection, moral hazard, and many other challenges. How can we ensure that the submitted work is accurate, especially when the verification cost is comparable to the cost of performing the task? How can we estimate the exhibited quality of the workers? What pricing strategies should be used to induce the effort of workers with varying ability levels? We develop a comprehensive framework for managing the quality in such micro-crowdsourcing settings: First, we describe an algorithm for estimating the error rates of the participating workers, and show how to separate systematic worker biases from unrecoverable errors and generate an unbiased "worker quality" measurement. Next, we present a selective repeated-labeling algorithm that acquires labels in a way so that quality requirements can be met at minimum cost. Then, we propose a quality-adjusted pricing scheme that adjusts the payment level according to the contributed value by each worker. We test our compensation scheme in a principal-agent setting in which workers respond to incentives by varying their effort. Our simulation results demonstrate that the proposed pricing scheme is able to induce workers to exert higher levels of effort and yield larger profits for employers compared to the commonly adopted uniform pricing schemes. We also describe strategies that build on our quality control and pricing framework, to tackle

*jwang@ust.hk
†panos@stern.nyu.edu
‡fprovost@stern.nyu.edu

crowdsourced tasks of increasingly higher complexity, while still maintaining a tight quality control of the process.

Keywords: crowdsourcing, pricing/incentive mechanisms, workflows, quality assurance, agency theory, service level agreement, simulation

1 Introduction

Crowdsourcing has emerged over the last few years as an important new labor pool for a variety of tasks (Malone et al., 2010), ranging from micro-tasks on Amazon Mechanical Turk (AMT) to big innovation contests conducted by Netflix and Innocentive. Mechanical Turk, in particular, dominates today the market for crowdsourcing "micro-tasks", which are easy for humans to accomplish, but remain challenging for computers (Ipeirotis, 2010). The employers on Mechanical Turk, who are called requesters in the Mechanical Turk parlance, can post a variety of small tasks, such as image tagging, language translation, text annotation, and so on. Workers complete these tasks and get compensated in the form of micro-payments, typically in the range of 5 to 20 cents per task. The immediate and elastic supply of labor makes it possible to complete these tasks with low latency and high throughput.

Despite the promise, significant challenges remain. Workers in crowdsourcing markets usually have different levels of expertise and experience, which cannot be communicated through traditional signals such as education level and work experience. Understandably, workers may also adjust their effort in response to incentive schemes, and exhibit heterogeneous quality in task execution. Unfortunately, verifying the quality of every submitted answer is an expensive operation and negates many advantages of crowdsourcing: the cost and time for verifying the correctness of the submitted answers (e.g., checking the answers for a question such as "Do you see any recognizable human face in the picture?") is typically comparable to the cost and time for performing the task itself. The difficulty of verification, combined with a uniform pricing scheme (i.e., paying all the workers the same price for completing the same type pf task), leads to both adverse selection and moral hazard: crowdsourced tasks are more appealing to workers who are less capable; and once hired, workers choose to exert an inefficient level of effort. The abundance of low-quality work (Wais et al., 2010) harms the reliability, scalability, and robustness of online labor markets.

Our main research objective is to develop a comprehensive framework for assuring the quality of the results of crowdsourcing processes in a cost-effective manner. Without loss of much generality, we focus on quality control for tasks that have answers consisting of a small set of discrete choices (e.g., "Does this photograph violate the terms of service? Yes or No."). While this might seem limiting, we show in Appendix 8.2 that many complex tasks can be broken down into a set of simpler operations for which a multiple choice task

serves as a key building block for quality assurance. Hence, our proposed scheme naturally fits into such workflows and provides a fundamental quality control mechanism for other more complicated operations. Such synergies lead to workflows that can accomplish complex tasks with guarantees of high-quality output, even when the underlying workforce has uncertain, varying, or even moderate-to-low quality.

Our first contribution is to use a decision-theoretic approach to create a "quality score" for each worker: The quality score is an unbiased estimate of the true uncertainty in the answers provided by the worker, after removing any systematic bias, and taking into account the costs of different types of errors.[1]

We then look at a "streaming" environment, where workers arrive over time while we are running the task, and so incoming workers can be assigned to different tasks dynamically. We introduce a novel selective repeated-labeling strategy which allocates more labels to tasks that are expected to incur higher misclassification costs, based on the estimated quality of the workers that have already worked on the tasks. We demonstrate significant savings in labor costs and execution time by using our dynamic resource allocation mechanism.

Having a reliable worker quality estimation method and a cost-effective selective repeated-labeling strategy, we next turn our attention to determining a fair and incentive-compatible pricing scheme for the workers. In particular, we consider a model with strategic workers of heterogeneous abilities, and propose an incentive mechanism that compensates workers according to their contribution towards achieving the required accuracy level. The contributed value of each worker is estimated based on the idea that multiple low-quality workers can work in tandem to generate high-quality data. Since workers' quality measurements are inherently uncertain, we also establish a payment scheme in which we pay workers based on the lower estimates of their quality, effectively withholding some payment for the workers that pass the test of time and prove themselves to really be reliable workers. As our quality estimates become more precise over time, we refund the "withheld" payment, ensuring that, in the limit, workers receive a payment that corresponds to their true quality, even in the presence of measurement uncertainties.

The remainder of the paper is organized as follows. Section 2 reviews related literature. Section 3 outlines the modeling assumptions and formalizes the problem. Section 4 describes the quality-estimation framework for the workers and the data. Section 5 presents the dynamic resource allocation mechanism, which saves labeling resources effectively, while still maintaining the required level of data quality. Section 6 proposes a pricing scheme, in a principal-agent setting, that rewards workers according to the value that they contribute as well as the competition in the market. The experimental results in Section 7 demonstrate a significant improvement over existing baselines, in terms of both data quality and workforce engagement. Section 8 concludes by describing the managerial implications, limitations, and directions for future research.

[1] For example, allowing a porn image to pass a moderation filter is often costlier compared to blocking incorrectly a legitimate image.

2 Literature Review

Our work lies at the intersection of computer science, information systems and economics. In this section, we survey the literature in the following streams of research: quality estimation and control, active information acquisition, agency problems (adverse selection and moral hazard), and payment schemes.

2.1 Quality Estimation and Control

One common approach to measure the quality of submitted answers is to use "gold" data: insert a small percentage of tasks for which the correct answers are known, and measure the performance against these tasks. The testing of worker quality using "gold" labels is related to, but distinct from, two lines of research: test theory in psychometrics and education (Crocker and Algina, 2006; DeMars, 2010), and acceptance sampling in operation management (Dodge, 1973; Wetherill and Chiu, 1975; Berger, 1982; Schilling, 1982). Existing test theory models are appropriate for accurate ability estimation. However, these models do not consider the additional costs that can be incurred: Each time we test a worker, we forfeit the opportunity to get some work done. This is analogous to the inspection cost in manufacturing process. Optimal acceptance sampling maximizes the profits of producers by striking the appropriate balance between quality assurance and total cost. A key difference is that in acceptance sampling, a production lot of items will get rejected if the number of defective items in a sample exceeds a threshold, whereas in crowdsourcing markets that deal with information goods, low-quality work can be combined to provide high-quality outcomes.

Another method to ensure quality is to rely on majority voting: ask multiple workers to complete the same task and use majority voting to identify the correct answer. In reality, most employers check agreement of workers with majority voting and dismiss workers systematically in disagreement with the majority. The disadvantages of this approach are: first, it does not account for heterogeneity in the exhibited quality of the workers; second, we have little knowledge about the correctness of the majority labels; and third, it suffers in the face of diligent and informative workers whose answers are biased.

Dawid and Skene (1979) propose an expectation maximization (EM) algorithm to estimate the diagnostic error rates of doctors when the patients' true diagnoses are not available. Variations of the algorithm were recently proposed by Carpenter (2008) and by Raykar et al. (2010). The algorithm iterates until convergence, following two steps: (1) estimates the true response for each patient, using records given by all the observers, accounting for the error rates of each observer; and (2) estimates the error rates of observers by comparing the submitted records with estimated true response. Welinder et al. (2010) proposed a generative Bayesian model in which each annotator is a multidimensional entity with variables representing competence, expertise and bias. One major objective across all these approaches is to estimate the error rates for each worker. In our work, we leverage the error rates of individual workers to obtain an unbiased quality measurement that

eliminates systematic worker biases.

2.2 Active Information Acquisition

Active information acquisition, which focuses on acquiring various types of information incrementally, so as to cost-effectively achieve different objectives, has been an important topic in machine learning and management literature. Moore and Whinston (1986, 1987) develop a theoretical decision-making framework in which the decision-maker gathers costly information optimally and sequentially to reduce the uncertainty associated with the final decisions. There have been a large number of papers (Cohn et al., 1994; Lewis and Gale, 1994; Roy and McCallum, 2001; Saar-Tsechansky and Provost, 2004) devoted to active learning, which aims to economize resources on training instances that are more likely to be informative for building classifiers. Another stream of papers (Lizotte et al., 2002; Zheng and Padmanabhan, 2006; Saar-Tsechansky et al., 2009) study the active feature-value acquisition problem where the values of features of the training data are costly to acquire.

In the context of repeated labeling using multiple noisy workers, Sheng et al. (2008) and Ipeirotis et al. (2014) have developed several different selective repeated-labeling strategies and shown that selective allocation of labeling resources can improve the overall labeling quality. But those strategies all suffer from the same drawbacks: all the workers are assumed to have equal level of quality when labeling each instance, and the same costs are incurred for different types of misclassification. We propose an expected-misclassification-cost-based selective repeated-labeling method which accounts for the heterogeneity in both worker quality and misclassification cost.

2.3 Agency Problems

Agency theory (Jensen and Meckling, 1976; Eisenhardt, 1989) maintains that the existence of information asymmetry between principals and agents can lead to both adverse selection (Akerlof, 1970) and moral hazard (Hölmstrom, 1979).

The pre-contractual problem of adverse selection arises since the agents possess private information of their true ability. In micro-task crowdsourcing platforms, because of the relative anonymity of the workers, the employer cannot readily assess the workers' qualifications to fulfill the tasks. The inability to differentiate between competent and incompetent workers leads to a situation where high-quality workers leave the market, and only low-quality workers remain. This might cause market failure: "it is quite possible to have the bad driving out the not-so-bad driving out the good in such a sequence of events that no market exists at all" (Akerlof, 1970). One mechanism to deal with this problem is reputation systems (Resnick et al., 2000; Dellarocas, 2003), which rely on the assumption that the past performance of workers reflects their true

ability. Reputation systems require a relatively large amount of historical data about the performance of each worker on each particular type of task, which is unfortunately largely missing in dynamic crowdsourcing settings. Our scheme can work either independently or in tandem with the existence of a reputation system. Reputation can be easily incorporated in our model as a prior belief about the true quality of the worker, which can be further updated as the worker engages in more tasks.

The post-contractual problem of moral hazard takes place when the agent's effort cannot be perfectly observed. Online paid crowdsourcing allows employers to reach distant workers but makes the monitoring of worker behavior challenging. Therefore, self-interested workers are more likely to engage in shirking or other types of opportunistic behavior to maximize their own profits. Since effort is not observable, compensation is often contingent on output. However, output is in many cases not one-dimensional but multifaceted (Holmstrom and Milgrom, 1991). In our context, both the *quantity* and *quality* of output are important to the employers. The existing literature models the tradeoff between quantity and quality in two ways. One stream of papers takes a multi-dimension approach in which quantity and quality are treated as two independent decision variables, and the agent's total cost depends on the effort that she devotes to each duty (Holmstrom and Milgrom, 1991; Olmos and Martínez, 2010). Another stream of papers captures the trade-off using a single decision variable, i.e., the average time spent per task (Lu et al., 2009; Anand et al., 2011; Kostami and Rajagopalan, 2013). In this paper, we adopt the second approach because it highlights the intrinsic quantity-quality trade-off in the simplest manner: the longer a worker spends on each task, the higher typically the quality of the submitted answers, but at the expense of completing fewer tasks. Proper compensation methods might provide explicit incentives for workers to submit high-volume and high-quality work (Lazear, 1986).

2.4 Payment Schemes

The choice of payment scheme has been a central topic of research in economics and management. A variety of payment schemes have been proposed and used in practice, such as fixed-wage, piece-rate, quota (Bonner et al., 2000), tournament (Lazear and Rosen, 1979) and piece-rate with monitoring (Nagin et al., 2002). Previous work has attempted to assess the relative effectiveness of various payment schemes (Lazear, 2000; Bonner et al., 2000; Agranov and Tergiman, 2013) either empirically or experimentally, but most of these papers only focus on one dimension of worker performance (i.e., quantity), leaving the testing about the effect on quality untouched. There are a few exceptions: for example, Paarsch and Shearer (2000) uses a structural model to analyze the payroll records of a tree-planting firm and show that workers under piece-rate contracts have higher productivity in terms of quantity but lower quality; Nagin et al. (2002) examine the data collected from an experiment in a call center and find that a reduction in monitoring rate actually

increases the likelihood of making bad calls.

Fortunately, the effect of compensation schemes on quantity and quality has gained growing interest in crowdsourcing settings. This is partly because the employer-employee relationships that develop in online marketplaces are short term, thereby increasing the agency risks towards employers. Mason and Watts (2010) found that a quota scheme performed better than a piece-rate scheme in motivating high-quality work. Harris (2011) showed that introducing both a bonus (on matches) and a penalty (on misses) into a piece-rate scheme improves the quality of the submitted work.

Our work differs from previous papers in several important respects. First, we have a nearly costless monitoring because the testing of worker quality in our setting is achieved by comparing one worker's labels with those given by a group of others.[2] Second, labels provided by low-quality workers are valuable because they can be aggregated to generate data that meet the prescribed level of quality. These allow us to develop a novel quality-adjusted piece-rate payment scheme which rewards workers for both quantity and quality.

To the best of our knowledge, this paper is to date the first to study the design of a comprehensive framework for quality assurance in a crowdsourcing setting where workers with heterogenous ability levels act strategically to maximize their expected profits by varying the effort they devote to each task. From a methodological standpoint, this paper integrates elements from quality control, active information acquisition and agency theory and brings both managerial and technical perspectives to crowdsourcing research.

3 Modeling Framework

In this section, we describe our modeling assumptions and formalize the problem. For model simplicity, we only consider the case of one type of task, one client, one service provider, and a pool of crowdsourced workers with varying ability levels. Table 1 summarizes the key notations used in the paper.

Task

In our labeling task, each object o is associated with a *latent* true class label $T^{(o)}$, picked from one of L different labels. The true class label $T^{(o)}$ is unknown and the task is to identify the true label for the object o.

Client

The client is the owner of the unlabeled objects, and wants the objects labeled with correct categories. To quantify the quality of labeling, the client provides a set of misclassification costs $\mathbf{c}$: the cost c_{ij} is incurred when an object with true label i is classified into category j. The client requires a service level agreement

[2]The monitoring accuracy depends on both the number and the quality of labels devoted to each task.

Notation	Definition
O	The set of objects that need to be labeled
L	The set of possible labels for the objects in O
$T^{(o)}$	True class of object (o)
$\boldsymbol{\pi}$	Vector with prior probabilities for object classes
π_i	Prior probability for class i
$\mathbf{p}^{(o)}$	Vector with probability estimates for the true label of object (o)
$p_i^{(o)}$	Probability that the true label of object (o) is i
$K^{(o)}$	Set of workers that assign labels to object (o)
$O^{(k)}$	Set of objects labeled by worker (k)
$\pi_j^{(k)}$	Probability that worker (k) assigns label j
$l_{(o)}^{(k)}$	Label that worker (k) assigns to object (o)
$I(l_{(o)}^{(k)} = i)$	Indicator function for the event $l_{(o)}^{(k)} = i$
$\mathbf{e}^{(k)}$	Confusion matrix for worker (k)
$e_{ij}^{(k)}$	Probability that worker (k) will classify an object with true category i into category j
$\boldsymbol{\theta}^{(k)}$	Dirichlet parameters for error rate distributions of worker (k)
$\mathbf{c}$	Matrix with the misclassification costs
c_{ij}	Cost incurred when an object with true label i is classified into category j
τ_c	Cost threshold specified in service level agreement (SLA)
S	Fixed price charged to the client for objects with average misclassification cost below τ_c
$\boldsymbol{\phi}^{(k)}$	Latent ability matrix of worker (k)
$w^{(k)}$	Reservation wage per unit time of worker (k)
$h^{(k)}$	Lifetime of worker (k)

Table 1: Key Notations Used

(SLA), with the guarantee that the average misclassification cost of the labeling will not exceed a threshold τ_c.[3] The client offers to the service provider an exogenously defined, fixed piece-rate price S for labeled objects[4] with average misclassification cost not exceeding τ_c.

Service Provider

The service provider[5] receives, from the outside client, the stream of jobs that need to be completed, together with the quality/cost requirement. The received tasks are posted on the crowdsourcing market for workers to work on. The service provider announces a price scheme and pays each worker according to the worker's quality, on a piecemeal (i.e., per task) basis. The service provider acts as an intermediary between the client and workers by ensuring a particular level of data quality for the client, and monitoring the performance of crowdsourced workers. The goal of the service provider is to maximize its own rate of profit.

[3]The cost can be determined post hoc, for example, using acceptance sampling (Schilling, 1982), and the decision on whether the promised labeling quality is met would be made accordingly.

[4]Although we assume that the price is exogenously defined, the price can also be defined by the service provider in response to competitive pressures. The only assumption that we need is the existence of a piece-wise price S.

[5]E.g., https://crowdflower.com/

Workers

Workers in crowdsourcing markets come to work on the available tasks. Each worker (k) is associated with: (1) a *latent* ability matrix $\boldsymbol{\phi}^{(k)}$, with $\phi_{ij}^{(k)}$ being the probability that worker (k) will classify an object with true category i into category j when the worker invests infinite time in the labeling;[6] (2) a reservation wage $w^{(k)}$, that is, the lowest wage per unit time at which the worker will accept the task; and (3) a lifetime $h^{(k)}$, which represents the total amount of time available to the worker. The distribution of ability, reservation wage, and lifetime $f_{\boldsymbol{\Phi},W,H}(\boldsymbol{\phi}, w, h)$ is common knowledge. However, the individual values of $\boldsymbol{\phi}^{(k)}$, $w^{(k)}$, and $h^{(k)}$ for each worker are all private knowledge of the workers and not known apriori to the service provider.

Following Lu et al. (2009), we characterize the intrinsic tradeoff between the quality and the productivity of a worker in a simple form: invested time per task increases the exhibited quality of the worker but reduces the number of tasks that the worker can work on during her lifetime. Specifically, for a worker (k) who spends time t on each task, the exhibited quality matrix is given by $g(\boldsymbol{\phi}^{(k)}, t)$, where g is a nondecreasing and concave function of time, indicating a diminishing marginal improvement in quality from an increase in time. Since the lifetime is allocated equally across all the tasks, the productivity of this worker is $h^{(k)}/t$ (i.e., number of tasks worker (k) can complete throughout her lifetime). Each worker is a profit-maximizer: given the payment scheme announced by the service provider, the worker chooses the amount of time spent on each task to maximize her expected profits per unit time.[7] (Note that, of course, she always has the option to not participate if her reservation wage is higher than the maximal attainable profits.)

4 Quality Estimation

In reality, the exhibited quality of a worker is jointly determined by her underlying true ability and the time she devotes to each task, and cannot be directly observed. The challenge facing the service provider is to come up with an effective strategy to estimate worker quality. Towards this, in Section 4.1, we describe a scheme that uses redundancy to generate estimates of the type and prevalence of the errors committed by workers in their tasks. Then, in Section 4.2, we investigate some problems of using error rates as a measure of quality, and describe how to generate an unbiased quality estimator, using a decision theoretic framework.

4.1 Worker Quality Estimation

An early paper by Dawid and Skene (1979) described how the diagnostic error rates of doctors can be estimated when the correct answers for the diagnoses are unknown. The basic idea is to rely on redundancy,

[6] This matrix captures the limit of worker's ability. As will be discussed shortly, a decrease in working time will induce a decline in worker's exhibited quality.

[7] We assume that the workers are risk-neutral.

that is, to obtain multiple opinions about the diagnosis. We rephrase their algorithm into our problem setting, in which workers assign labels to objects. The algorithm iterates until convergence, following two steps: (1) estimate the true class for each object, using labels provided by the workers, accounting for the error rates of each worker; and (2) estimate the error rates of workers by comparing the submitted labels with estimated correct class for each object. The final outputs of this expectation-maximization algorithm are the class probability distribution $\mathbf{p}^{(o)}$ for each object (o) and the estimated error rates for each worker (k) represented by a "confusion matrix" $\mathbf{e}^{(k)}$.[8] The algorithm performs well when each worker submits a sufficient number of labels. Unfortunately, participation in crowdsourcing environments follows a very skewed distribution (Stewart et al., 2010; Nov et al., 2011) with only a few workers contributing a lot, while the majority submit only a few tasks. In such a setting, maximum likelihood approaches result in over-confident estimates of the error rates of the workers.

Following Raykar et al. (2010), we move from maximum likelihood estimates to Bayesian ones. If the true class of an object is i, we model the error rates of the worker (k) as a Dirichlet distribution with parameter vector $\boldsymbol{\theta}_i^{(k)}$. The value of $\theta_{ij}^{(k)}$ is given by $\theta_{ij}^{(k)} = \alpha_{ij}^{(k)} + n_{ij}^{(k)}$, where $n_{ij}^{(k)}$ represents the number of times that the worker classified objects of class i into class j and $\alpha_{ij}^{(k)}$ captures the prior belief. If we start with an uninformative prior, then $\theta_{ij}^{(k)} = 1 + n_{ij}^{(k)}$. Using this strategy, the error rates of a worker can be fully captured by a set of Dirichlet distributions (which reduce to Beta distributions for the binary case). Algorithm 1 presents a sketch of the process, where $\boldsymbol{\theta}^{(k)}$ parameterizes the error rate distributions of worker (k) and $\mathbf{e}^{(k)}$ is defined by the expected values.[9]

4.2 Generating Unbiased Quality Measurements

The confusion matrix $\mathbf{e}^{(k)}$ for each worker (k) is not a scalar, and therefore cannot be used as a simple metric of the worker quality. A straightforward method is to simply sum up the non-diagonal entries of the matrix $\mathbf{e}^{(k)}$, weighting each error rate by the estimated prior of the corresponding class (i.e., how often the worker submits an incorrect label). Unfortunately, this approach would incorrectly reject biased but careful workers. Consider the following example:

Example 1 *Two workers are working on the task of classifying web sites into two groups:* porn *and* notporn. *Worker A is always incorrect: labels all* porn *web sites as* notporn *and vice versa. Worker B is lazy and classifies all web sites, irrespectively of their true class, as* porn. *Which of the two workers is better? A simple error analysis indicates that the error rate of worker A is 100%, while the error rate of worker B is*

[8] The element in the i-th row and j-th column of the confusion matrix $e_{ij}^{(k)}$ gives the probability that worker (k) classifies an object with true class i into class j.

[9] Since crowdsourcing workers tend to have heterogeneous levels of quality, we use uninformative priors in our estimation (i.e., $\alpha_{ij}^{(k)} = 1$).

Input: Set of Labels $\{l_{(o)}^{(k)}\}$

Output: Confusion matrix $\mathbf{e}^{(k)}$ for each worker (k), Class priors $\boldsymbol{\pi}$, Class probability estimates $\mathbf{p}^{(o)}$ for each object (o)

Initialize class probability estimates $\mathbf{p^{(o)}}$ for each obect (o):$p_i^{(o)} = (\sum_{(k)\in K^{(o)}} I(l_{(o)}^{(k)} = i))/(\left|K^{(o)}\right|)$;

while *not converged* **do**

  Estimate the $\boldsymbol{\theta}^{(k)}$: $\theta_{ij}^{(k)} = \alpha_{ij}^{(k)} + n_{ij}^{(k)} = \alpha_{ij}^{(k)} + \sum_{(o)\in O^{(k)}} p_i^{(o)} I(l_{(o)}^{(k)} = j)$;

  Estimate the confusion matrix $\mathbf{e}^{(k)}$: $e_{ij}^{(k)} = \theta_{ij}^{(k)}/(\sum_{m=1}^{L} \theta_{im}^{(k)})$;

  Estimate the class priors $\boldsymbol{\pi}$: $\pi_i = (\sum_{(o)} p_i^{(o)})/(|O|)$;

  Compute the object-class probabilities $\mathbf{p}^{(o)}$ for each object (o):

  $$p_i^{(o)} = \frac{\pi_i \prod_{(k)\in K^{(o)}} \prod_m (e_{im}^{(k)})^{I(l_{(o)}^{(k)}=m)}}{\sum_q \pi_q \prod_{(k)\in K^{(o)}} \prod_m (e_{qm}^{(k)})^{I(l_{(o)}^{(k)}=m)}};$$

end

return $\{\mathbf{e}^{(k)}\}$, *class priors* $\boldsymbol{\pi}$, $\{\mathbf{p}^{(o)}\}$

Algorithm 1: Bayesian expectation maximization algorithm for worker error rates estimation.

only 50%.[10] *However, it is not difficult to see that the errors of worker A are easily reversible, while the errors of worker B are irreversible. In fact, with this reversal, worker A can be utilized as a perfect worker, while worker B is a spammer.*

Naturally, a question arises: Given estimates of the confusion matrix $\mathbf{e}^{(k)}$ for each worker (k), how can we distinguish between low-quality workers and high-quality, but biased, workers? How can we separate systematic biases from the intrinsic, non-recoverable error rates?

We start with the following observation: Each worker assigns a "hard" label to each object. Using the error rates for this worker, we can transform this assigned label into a "soft" label (i.e., posterior estimate), which is the best possible estimate that we have for the true class. If we have L possible classes and the worker assigns class j as a label to an object, we can transform this "hard" assigned label into the "soft", posterior label: $\left\langle \pi_1 \cdot e_{1j}^{(k)}, \ldots, \pi_L \cdot e_{Lj}^{(k)} \right\rangle$, where π_i is the prior that the object belongs to class i and $e_{ij}^{(k)}$ is the probability that worker (k) classifies into class j an object that in reality belongs to class i. Of course, the quantities above need to be normalized by dividing them with $\pi_j^{(k)} = \sum_{i=1}^{L} \pi_i \cdot e_{ij}^{(k)}$, where $\pi_j^{(k)}$ denotes the probability that worker (k) assigns label j.

Now, we can proceed to estimate the cost of labeling. To estimate the expected cost of each soft label, we need to consider the costs associated with all possible classification errors. In the simplest case, we have a cost of 1 when an object is misclassified, and 0 otherwise. In a more general case, we have a cost c_{ij} when an object of class i is classified into category j.

[10] Assume, for simplicity, equal priors for the two classes.

Input: Confusion matrix $\mathbf{e}^{(k)}$, Misclassification cost matrix $\mathbf{c}$, Class prior vector $\boldsymbol{\pi}$
Output: Expected cost $cost^{(k)}$ for each worker (k)

foreach *worker* (k) **do**
  Estimate $\pi_l^{(k)}$ (how often the worker (k) assigns label l);
  $cost^{(k)} = 0$;
  foreach *label* l*, assigned with probability* $\pi_l^{(k)}$ **do**
    Compute the posterior probability $\mathbf{soft}^{(k)}(l)$ that corresponds to label l assigned by worker (k);
    Using Proposition 2, compute $ExpCost(\mathbf{soft}^{(k)}(l))$ for the soft label;
    $cost^{(k)}$ += $ExpCost(\mathbf{soft}^{(k)}(l)) \cdot \pi_l^{(k)}$;
  end
end
return $cost^{(k)}$ *for each worker* (k)

Algorithm 2: Estimating the Expected Cost of each Worker

Proposition 2 *Given the classification costs* $\mathbf{c}$ *and a soft label* $\mathbf{p} = \langle p_1, p_2, \ldots, p_L \rangle$*, the expected cost of the soft label* $\mathbf{p}$ *is* $ExpCost\,(\mathbf{p}) = \min_{1 \le j \le L} \sum_{i=1}^{L} p_i \cdot c_{ij}$.

The proof is straightforward. The expected classification cost if we report j as the true class is equal to the posterior probability of the object belonging to class i (namely, p_i), multiplied with the associated cost of classifying an object of class i into class i (namely, c_{ij}). The best decision is to report the category j with the minimum expected classification cost across all classes. The expected cost can help us make the best classification decision in the case where we receive only a single label per object. Given that we know how to compute the expected cost of each label, we can now easily estimate the expected cost of each worker (k). Algorithm 2 illustrates the process.

Example 3 *Consider the costs for the workers A and B from the previous example. Assuming equal priors across classes, and* $c_{ij} = 1$*, if* $i \neq j$ *and* $c_{ij} = 0$*, if* $i = j$*, we have the following: The cost of worker A is 0, as the soft labels generated by A are* $\langle 0, 1 \rangle$ *and* $\langle 1, 0 \rangle$*. For worker B, the cost is 0.5 (the maximum possible) as the soft labels generated by B are all* $\langle 0.5, 0.5 \rangle$ *(i.e., highly uncertain).*

It turns out that workers with confusion matrices that generate posterior labels with probability mass concentrated into a single class (i.e., confident posterior labels) will tend to have low estimated cost, as the minimum sum in Proposition 2 will be close to 0. On the contrary, workers that generate posterior labels with probabilities widely spreaded across classes (i.e., uncertain posterior labels) will tend to have high misclassification costs. Notice, as illustrated in the example above, that it is not necessary for a worker to return the correct answers in order to have low costs. Our quality metric based on expected misclassification costs resolves quite a few issues with online workers who exhibit systematic biases in their answers but also put a lot of effort into coming up with the answers. Prior approaches that rely on agreement generate a

significant number of rejections for such workers, which in turn discourages them from working for employers that heavily rely on agreement.

5 Selective Repeated Labeling in Data Quality Assurance

In the previous section, we focused on a static setting: we have all the data, perform the analysis, and infer data and worker quality. In reality, workers arrive to the market according to some exogenous traffic process, so labels are often obtained incrementally and dynamically. As mentioned in the modeling section, the payment of the service fee is contingent on the assurance of a particular level of data quality. Therefore, it is important to monitor efficiently the quality level of the delivered data, and to allocate worker resources appropriately. The key insight is that, given the set of labels assigned to an object, it is possible to estimate its class probability distribution and expected misclassification cost (Section 5.1), and based on the expected misclassification costs of the objects, the service provider can allocate labeling resources in a way that increases data quality in a cost-effective manner (Section 5.2).

5.1 Data Quality Estimation

In the labeling process, an important parameter for the service provider to determine is the number of workers to be assigned to each particular object. As more workers inspect and label an object, the confidence about the classification decision increases, in expectation: the more workers assigned to each object, the higher the integrated labeling quality. At the same time, the service provider wants to minimize the labor costs. Since the goal is to have an overall data quality higher than the quality promised in the SLA, it is optimal to assign to each object enough labels so that the expected misclassification cost of the labeled object is just below the one specified in the SLA.

How can we estimate the quality of the labeling? Assume that we have an object that has been labeled by m workers, and that these workers assigned a multiset of labels $\mathbf{j} = \{j_1, j_2, \cdots, j_m\}$, where j_s is the label assigned by the s-th worker in the set. Using the results from Section 4, we assume that we have an estimate of the confusion matrix for each worker, which we denote as $\mathbf{e}^{(s)}$ (see Section 3). We also assume that given the true class of the object, the labels submitted by different workers are conditionally independent. Under the conditional independence assumption, the probability of seeing a particular label assignment $\mathbf{j} = \{j_1, j_2, \cdots, j_m\}$ for an object of true class l is given by:

$$P(\mathbf{j}|\mathbf{e}^{(1)}, \mathbf{e}^{(2)}, \cdots, \mathbf{e}^{(m)}, l) = \prod_{s=1}^{m} e^{(s)}_{lj_s} \tag{1}$$

Using Bayes' Rule, we have the posterior probability of the object belonging to class l, given by:

$$P(l|\mathbf{j}, \mathbf{e}^{(1)}, \mathbf{e}^{(2)}, \cdots, \mathbf{e}^{(m)}) \propto \pi_l \cdot P(\mathbf{j}|\mathbf{e}^{(1)}, \mathbf{e}^{(2)}, \cdots, \mathbf{e}^{(m)}, l) = \pi_l \cdot \prod_{s=1}^{m} e_{ljs}^{(s)} \tag{2}$$

This is the best possible estimate that we have for the true class of the object, similarly to the "soft" label we generate in Section 4.2, but now combining the labels from multiple workers. Using Proposition 2, we can estimate the expected misclassification cost of this object. As before, objects with posterior probabilities concentrated in one class have low expected misclassification costs while those with posterior probabilities spreaded across classes have high misclassification costs, and therefore, need more additional attention.

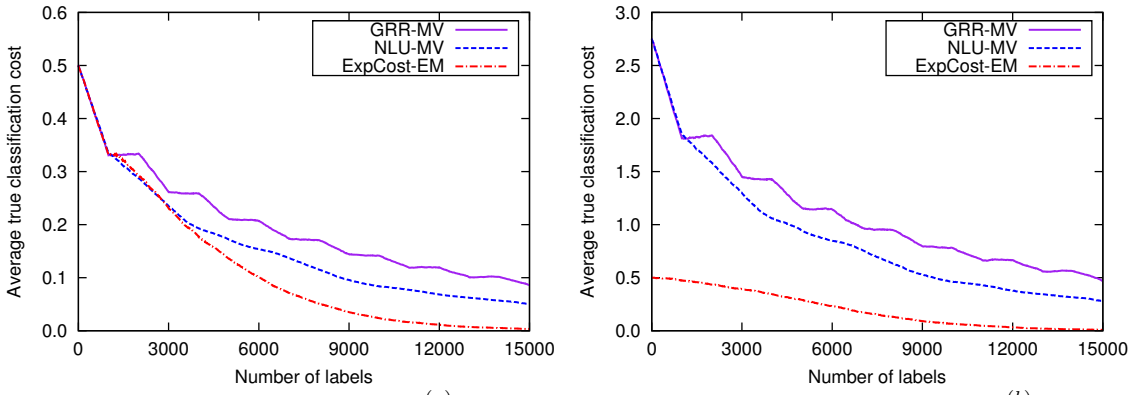
5.2 Selective Repeated Labeling Based on Expected Misclassification Cost

Given the quality estimate (i.e., expected classification cost) for each object, we can move to devise a labeling policy for the service provider to assign incoming workers to objects. The current state-of-the-art strategy for selective repeated labeling using only the information in the label multiset is the "new label uncertainty" (*NLU*) presented by Ipeirotis et al. (2014), in which the next object to (re-)label is the one with the highest label uncertainty score (without considering the different quality levels of individual workers). In this paper, we introduce a new expected-misclassification-cost-based selective repeated-labeling strategy (*ExpCost*) which allocates more labeling resources to the objects with high expected classification costs. When a worker arrives, *ExpCost* assigns her to label the object with the highest expected misclassification cost, as long as the object has an expected cost higher than the one promised in the SLA.[11] (If the cost is lower, the service provider is ready to deliver the objects with estimated labels to the client.) Our *ExpCost* method improves upon *NLU* by explicitly taking into account workers' heterogenous levels of quality and the different costs incurred by various types of misclassifications. We demonstrate the superior performance of *ExpCost* next, using a set of simulation experiments.

5.3 Effectiveness of *ExpCost* in Achieving Data Quality

We test the performance of the following repeated-labeling strategies: (1) *GRR* (generalized round-robin) which assigns the next worker to label the object with the fewest labels; (2) *NLU* which assigns the next worker to label the object with highest label uncertainty score (Ipeirotis et al., 2014); and (3) *ExpCost* which prioritizes objects with high expected misclassification costs. In both *GRR* and *NLU*, the final class is determined using simple majority voting (MV) because the methods are agnostic to differences in worker

[11] In the actual implementation, the service provider can either meet the SLA in expectation, or with a certain confidence level. A higher confidence level reduces the risk of failing to meet the standard but demands more labeling resources.



(a) symmetric cost matrix $c_{ij}^{(a)}$ (b) asymmetric cost matrix $c_{ij}^{(b)}$

Figure 1: Average true classification cost as a function of the number of labels acquired, for a round robin strategy (*GRR*), a selective labeling strategy based on label uncertainty (*NLU*), and our proposed strategy based on expected cost (*ExpCost*)

quality when labeling the same object. In contrast, in *ExpCost*, the final class is determined using weighted majority voting, as discussed in the context of the EM algorithm (see Section 4.1).

The simulation setup is as follows: we have 1000 objects, evenly assigned to two categories, and 200 workers. We draw the confusion matrix $\mathbf{e}$ of each worker from a set of two beta distributions: $\mathbf{Beta}(\mathbf{4},\mathbf{2})$ and $\mathbf{Beta}(\mathbf{2},\mathbf{4})$, each corresponding to a row of the confusion matrix $\mathbf{e}$. Each time, we draw a worker uniformly from the worker pool, and depending on the strategy used (*GRR*, *NLU*, and *ExpCost*), we assign the worker to the example with the highest priority. We test the performance of our proposed method under two settings: a symmetric cost matrix $\mathbf{c}^{(a)} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, and an asymmetric cost matrix $\mathbf{c}^{(b)} = \begin{pmatrix} 0 & 1 \\ 10 & 0 \end{pmatrix}$.

Figure 1 shows the actual misclassification cost of the data as a function of the number of labels acquired for *GRR*, *NLU*, and *ExpCost*, under the two cost settings. The first observation is that the *ExpCost* method beats both *GRR* and *NLU* consistently. The advantage becomes even more substantial when classification costs are asymmetric. Second, in Figure 1(a), *NLU* and *ExpCost* have a similar performance during the early stage (when the number of labels acquired is fewer than 4000). This happens because the EM algorithm has not obtained good estimates for workers yet. The performance gap between *ExpCost* and *NLU* increases later on, showing that knowing the individual worker quality can help achieve better data quality.[12]

[12]For all the later experiments, we use the *ExpCost* method for label resource allocation.

5.4 Batch Processing

Two minor points limit the applicability of the labeling strategy described above in real-world large data environments. First, at each time point, we need to compute the expected classification costs of all the objects and choose the one with the highest cost, which is computationally expensive. Second, we tend to assign workers to objects for which we are less certain about first; however, an accurate estimation of worker quality relies on a good estimation of the labels for the objects that the worker has already worked on. This poses a disadvantage for the early-coming workers since they need to wait for a long time to get their expected cost correctly estimated. To avoid the computational complexity and latency in worker quality updates, we divide the full set of objects into a number of subsets $N = \{N_1, N_2, \cdots, N_n\}$, where each N_i only contains a relatively small number of objects.[13] We will start with the first subset N_1, and move to N_2 when the average expected cost of misclassification in N_1 is below the one specified in SLA, and so on.

6 Quality-Adjusted Pricing Scheme

The previous two sections deal with the crowdsourcing process, assuming workers' participation and effort decisions are given. However, in reality, workers are often rational agents and respond to external monetary incentives. Therefore, it is important for the service provider to design and implement an appropriate payment scheme to induce workers to behave in a desired manner. As a first step towards this objective, in Section 6.1, we discuss how to correctly assess the monetary value of workers to the service provider; and in Section 6.2, we propose a quality-adjusted piece-rate pricing scheme that ties payment to worker performance.

6.1 Monetary Value of Workers with Heterogeneous Quality

For ease of exposition, we first divide workers into two groups, qualified and not qualified: A worker is a *qualified* worker if the quality of the worker satisfies the SLA; otherwise, the worker is considered an *unqualified* worker. Since the client pays S for each successfully labeled object with labeling quality above the promised level, each label submitted by a qualified worker is worth S to the service provider. However, many workers in crowdsourcing markets fall into the category of unqualified workers whose quality does not meet the level promised by the service provider. In fact, there might be cases where *no* worker satisfies the desired quality.[14] Simply considering these workers as having zero value and disregarding their labels is short-sighted and renders the problem essentially intractable. Although each individual worker does not necessarily submit

[13] The number of objects within each batch can be decided by the service provider. Smaller batches save computation time at the cost of suboptimal resource allocation.

[14] Or, more commonly, it is not cost-effective to allow workers to be slow and careful in order to meet the SLA requirement. As specified in the modeling framework, the marginal improvement in quality from worker's additional effort is diminishing.

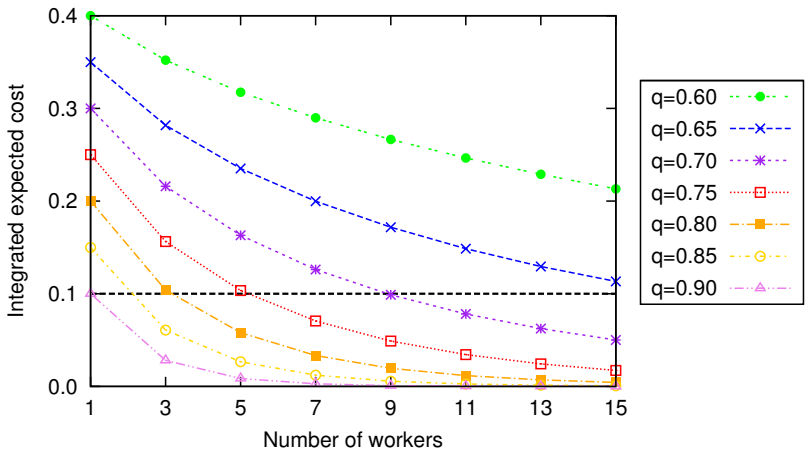


Figure 2: The relationship between the number of workers and integrated expected cost

high-quality labels, a group of them as a whole may be able to meet the SLA requirement. Several papers in the literature (Sheng et al., 2008; Snow et al., 2008; Welinder et al., 2010; Raykar et al., 2010; Ipeirotis et al., 2010; Bachrach et al., 2012) have shown that multiple, low-quality workers can be used to generate results that have high quality. The focus of this section is to examine the value of such "unqualified" workers.

The objective of the service provider is to get data labeled with classification cost lower than the level determined by the SLA. In Section 5, we described how we can improve data quality (and decrease expected cost of an object), by allocating multiple workers to label it. Therefore, a *set of unqualified workers that in tandem can generate labels of high quality* can be considered equivalent to a single, qualified worker. From this, we determine the value of unqualified workers according to the level of redundancy required to reach the required quality level.

Example 4 *Suppose that a client has a binary classification problem with equal priors, misclassification costs set to* 1*, a fixed price of* \$1 *offered to the service provider, and a SLA that requires a classification cost lower than* 0.1*. If we have workers with a confusion matrix of* $\mathbf{e} = \begin{pmatrix} q & 1-q \\ 1-q & q \end{pmatrix}$*, how many workers do we need to assign to each object, to achieve the SLA requirement? Figure 2 shows the relationship between the number of workers and the integrated expected cost with the value of* q *ranging from* 0.60 *to* 0.90 *with an interval of* 0.05*. The black dash line indicates the SLA-promised cost level. We can see that:*

1. *A worker with* $q = 0.9$ *is a qualified worker, and is worth* \$1 *to the service provider.*
2. *A worker with* $q = 0.8$ *is unqualified. However, a set of* 3 *workers with* $q = 0.8$ *generate labeling of SLA quality. Therefore a worker with* $q = 0.8$ *is worth* \$0.33*.*
3. *We need 9 workers with* $q = 0.7$ *to reach the SLA quality, therefore a worker with* $q = 0.7$ *is worth*

\$0.11.

The example above illustrates that the *value* of a worker is inversely proportional the number of workers with the same error rates required to achieve the acceptable level of cost. The example illustrates the process for a worker with a specific confusion matrix; next, we show the process for estimating the value of a worker with an arbitrary confusion matrix $\mathbf{e}$.

Definition 5 *The* value $v(\mathbf{e})$ *of a worker with a confusion matrix* $\mathbf{e}$ *is:* $v(\mathbf{e}) = \frac{S}{d(\mathbf{e})}$, *where* $d(\mathbf{e})$ *is the number of workers with confusion matrix* $\mathbf{e}$ *required to reach the SLA-defined classification cost of* τ_c, *and* S *is the price charged to a client for a unit of SLA-compliant work. For qualified workers* $d(\mathbf{e}) = 1$, *while for unqualified workers* $d(\mathbf{e}) > 1$.

Now the key challenge is to estimate the value $d(\mathbf{e})$ for an arbitrary confusion matrix $\mathbf{e}$. For this, we need to estimate the number of workers with identical confusion matrix $\mathbf{e}$ that are required to generate labeling of acceptable quality. Assume that we have m workers with identical confusion matrix $\mathbf{e}$ who assign labels to an object. This generates a label assignment $\mathbf{l} = \{l_1, \cdots, l_m\}$, which, because of the exchangeability of the labels, can be represented as a count of all the class labels $\mathbf{n} = \{n_1, \cdots, n_L\}$. When the true class label is i (which occurs with probability π_i), this label assignment happens with probability $Mult(\mathbf{n}|m, \mathbf{e}_{i\cdot}) = \binom{m}{n_1, \cdots, n_L} \cdot \prod_{j=1}^{L} (e_{ij})^{n_j}$, which is the probability mass function (pmf) of the multinomial distribution with parameters m (count of trials) and $\mathbf{e}_{i\cdot}$ (the row of the confusion matrix $\mathbf{e}$ that corresponds to the class i). Integrating this over all the classes, we get the overall probability of seeing $\mathbf{n}$ is:

$$P(\mathbf{n}) = \sum_{i=1}^{L} \pi_i \cdot Mult(\mathbf{n}|m, \mathbf{e}_{i\cdot}) = \binom{m}{n_1, \cdots, n_L} \sum_{i=1}^{L} \pi_i \cdot \prod_{j=1}^{L} (e_{ij})^{n_j} \tag{3}$$

Following the same procedure in Section 5.1, for each label assignment $\mathbf{n} = \{n_1, \cdots, n_L\}$, the "soft" label before normalization is proportional to:

$$\left\langle \pi_1 \cdot \prod_{j=1}^{L} (e_{1j})^{n_j}, \ldots, \pi_L \cdot \prod_{j=1}^{L} (e_{Lj})^{n_j} \right\rangle \tag{4}$$

The expected misclassification cost associated with the label assignment $\mathbf{n}$ is then estimated using Proposition 2. By repeating the process across all the possible label assignments and weighting the cost of each one by its occurrence probability, we get the average misclassification cost when using m workers with confusion matrix $\mathbf{e}$. Knowing how to compute the integrated expected cost, the value derivation becomes easier. Given a worker with specific confusion matrix $\mathbf{e}$, we simply find the minimum number of workers $d(\mathbf{e})$ we need to achieve the required cost level.

Input: Confusion matrix $\mathbf{e}$, Misclassification cost matrix $\mathbf{c}$, Class prior vector $\boldsymbol{\pi}$, Unit price for qualified objects S, Sample size N, Maximum number of workers D
Output: Value $v(\mathbf{e})$

for $x = 1$ *to* N **do**
  Generate object x with true class drawn from the prior probability distribution $\boldsymbol{\pi}$;
  Using Proposition 2, compute $ExpCost(x)$ for the prior probability vector;
end
$cnt = 0$;
while $cnt \leq D \cdot N$ **do**
  Pick the object y with the highest expected cost (i.e., $ExpCost(y) \geq ExpCost(x), \forall x$);
  Draw one label for the object y, following confusion matrix $\mathbf{e}$;
  $cnt = cnt + 1$;
  Using Equation (4), compute the posterior probability vector $\mathbf{p}(y)$ that corresponds to object y;
  Using Proposition 2, compute $ExpCost(y)$ for the posterior probability vector $\mathbf{p}(y)$;
  $SumCost = 0$;
  for $x = 1$ *to* N **do**
    $SumCost = SumCost + ExpCost(x)$;
  end
  $cost = \dfrac{SumCost}{N}$;
  if $cost \leq \tau_c$ **then**
    break;
  end
end
if $cnt \leq D \cdot N$ **then**
  $d(\mathbf{e}) = \dfrac{cnt}{N}$; $v(\mathbf{e}) = \dfrac{S}{d(\mathbf{e})}$;
else
  $v(\mathbf{e}) = 0$;
end
return $v(\mathbf{e})$

Algorithm 3: Estimating the value $v(\mathbf{e})$ of a worker with confusion matrix $\mathbf{e}$

Unfortunately, except for very simple cases, there is no closed form solution to this problem, and the computational complexity increases exponentially with the value of $d(\mathbf{e})$. In addition, the $d(e)$ generated above is likely to be an overestimate as we force each label assignment to have equal number of labels. As illustrated in the previous section, selective label acquisition can potentially reduce the amount of labels required to achieve quality level. Therefore, we resort to a Monte Carlo approach for estimating $d(\mathbf{e})$ in which labels are drawn incrementally and prioritized to objects with high expected misclassification costs, allowing some types of label assignments to have fewer labels than others. Algorithm 3 illustrates the overall process.[15]

[15]To save computation cost without sacrificing too much accuracy, we set $D = 30$ and $N = 1000$ in the actual implementation.

6.2 Quality-Adjusted Piece-Rate Pricing Mechanism

The objective of the service provider is to maximize profits, which depends on both the quantity and the quality of the labels submitted by workers. As illustrated in the previous analytical and experimental papers (Lazear, 1986; Paarsch and Shearer, 2000), simple piece-rate pricing schemes are likely to induce workers to favor quantity to the exclusion of quality and hurt the profitability of the service provider. Therefore, it is necessary to tie payment to quality and give workers an incentive to invest more effort on each task.

The service provider knows: (1) the distribution of worker's ability, reservation wage, and lifetime $f_{\mathbf{\Phi},W,H}(\boldsymbol{\phi}, w, h)$; and (2) the relationship between worker's ability, effort and exhibited quality level $g(\boldsymbol{\phi}, t)$. Although both ability $\boldsymbol{\phi}$ and effort t are not publicly observable and thus cannot be contracted upon, the exhibited quality of the workers can be estimated using the approach presented in Section 4. The service rewards each worker based on a simple sharing rule: a proportion of contributed value will be given back to the worker (i.e., $r(\mathbf{e}) = \alpha \cdot v(\mathbf{e})$, $0 < \alpha < 1$). The problem for the service provider is to find an optimal α that maximizes its own rate of profit. This is essentially a quality-adjusted piece-rate pricing mechanism that rewards for both quantity and quality.[16] The profit-sharing model encourages workers to behave in a manner that aligns with the objective of the service provider.

Suppose that the service provider announces a payment scheme $r(\mathbf{e}) = \alpha \cdot v(\mathbf{e})$ to the worker in the crowdsourcing pool. A given worker with ability matrix $\boldsymbol{\phi}$, reservation wage w, and lifetime h will participate and choose an effort t^* if and only if the following two constraints are satisfied: (i) incentive compatibility (IC): $t^* = \arg\max_t r(g(\boldsymbol{\phi}, t)) \cdot h/t = \arg\max_t \alpha \cdot v(g(\boldsymbol{\phi}, t))/t$; and (ii) individual rationality (IR): $\alpha \cdot v(g(\boldsymbol{\phi}, t^*))/t^* > w$. In this case, the net benefit that the service provider derives from this worker is $(1-\alpha) \cdot v(g(\boldsymbol{\phi}, t^*)) \cdot h/t^*$. If (ii) does not hold, the worker will not participate and the service provider receives no benefit. The expected net profit of the service provider under payment scheme $r(\mathbf{e}) = \alpha \cdot v(\mathbf{e})$ is given by integrating this over all possible values of ability matrix $\boldsymbol{\phi}$, reservation wage w, and lifetime h.

The optimal pricing scheme $r^*(\mathbf{e}) = \alpha^* \cdot v(\mathbf{e})$ solves the following maximization problem:

$$
\begin{aligned}
&\alpha^* = \arg\max_{\alpha} \int_0^{\infty} \int_0^{\infty} \int_{\boldsymbol{\phi}} (1-\alpha) \cdot v(g(\boldsymbol{\phi}, t^*)) \cdot h/t^* \cdot f_{\mathbf{\Phi},W,H}(\boldsymbol{\phi}, w, h) \mathrm{d}\boldsymbol{\phi} \mathrm{d}w \mathrm{d}h \\
&\text{subject to} \qquad t^* = \arg\max_{t} \frac{v(g(\boldsymbol{\phi}, t))}{t} \qquad \text{(IC)} \\
&\text{and} \qquad \frac{\alpha \cdot v(g(\boldsymbol{\phi}, t^*))}{t^*} > w \qquad \text{(IR)}
\end{aligned}
\tag{5}
$$

When distribution $f_{\mathbf{\Phi},W,H}(\boldsymbol{\phi}, w, h)$ is known, α^* can be computed through a variety of optimization methods, returning $r^*(\mathbf{e})$, the optimal piecemeal price to pay for a worker with exhibited quality matrix $\mathbf{e}$.

[16] Note that in principle the service provider could do better by using other forms of contracts. We choose this pricing mechanism because it is easy to implement, and yet very effective in inducing an efficient level of effort.

We should notice that the optimality condition above depends on the excess demand for labor (so that the service provider's dominant strategy is to acquire labels from each worker). This is realistic, since many companies nowadays have massive amounts of data and thus are experiencing a shortage of workers who can help them label these data. The model can be easily adapted to the setting in which supply exceeds demand by lowering the proportion of money transferred to workers until supply matches demand.

6.3 Real-Time Pricing under Imperfect Knowledge of Worker Quality

The estimates we get following the techniques in Section 4.1 are deterministic, but imperfect. Although our model assumes that the true quality of a worker $\mathbf{e}$ is fixed, our *estimate* of $\mathbf{e}$ is changing over time. Just based on sampling theory, the piece-rate payment of a worker is expected to fluctuate as the worker labels more examples, even if the payment is expected to converge towards the optimal price over time. Unfortunately, this fluctuation is not an acceptable part of a payment scheme. A worker would be negatively surprised if suddenly their wage plummets because of a single labeling mistake. How should we pay workers in this setting? Ideally, we want a payment scheme that:

1. Rewards workers with a payment as close as possible to their (unknown) optimal price.
2. Avoids payment fluctuations, resulting from expected measurement fluctuations, preferring a smooth payment evolution over time.
3. Avoids a decreasing payment slope, which can be interpreted as punishment, and prefer payment schemes that have either stable or increasing payment slopes.[17]

Condition 1 allows maximum worker engagement: Each worker has a reservation wage and a lifetime: if the average payment per unit time at the end of lifetime is lower than the reservation wage, the worker will not participate in the task. Of course, the more examples a worker labels, the closer the payment $\hat{r}(\mathbf{e})$ is to the optimal payment $r^*(\mathbf{e})$ under perfect knowledge of worker quality. Unfortunately, this scheme also leads to significant up and down fluctuations (violating condition 2), especially early on, leading to worker confusion. To avoid the sudden fluctuations, we can pay based on a moving average of worker quality, which softens the potential estimation fluctuations. Unfortunately, paying using a moving average can also lead to a decrease in payment over time, if the worker starts by giving a few correct answers before naturally reverting to the mean performance.

Our solution is a process that we call **payment with reimbursements**. Our scheme rewards workers over time by paying based on pessimistic estimates of worker quality (i.e., underpays initially) but compensates for the underpayment by *reimbursing* in later periods the payment withheld due to the uncertainty. To ensure

[17] Yin et al. (2013) found that an increasing payment slope improved worker quality while a decreasing payment slope hurt.

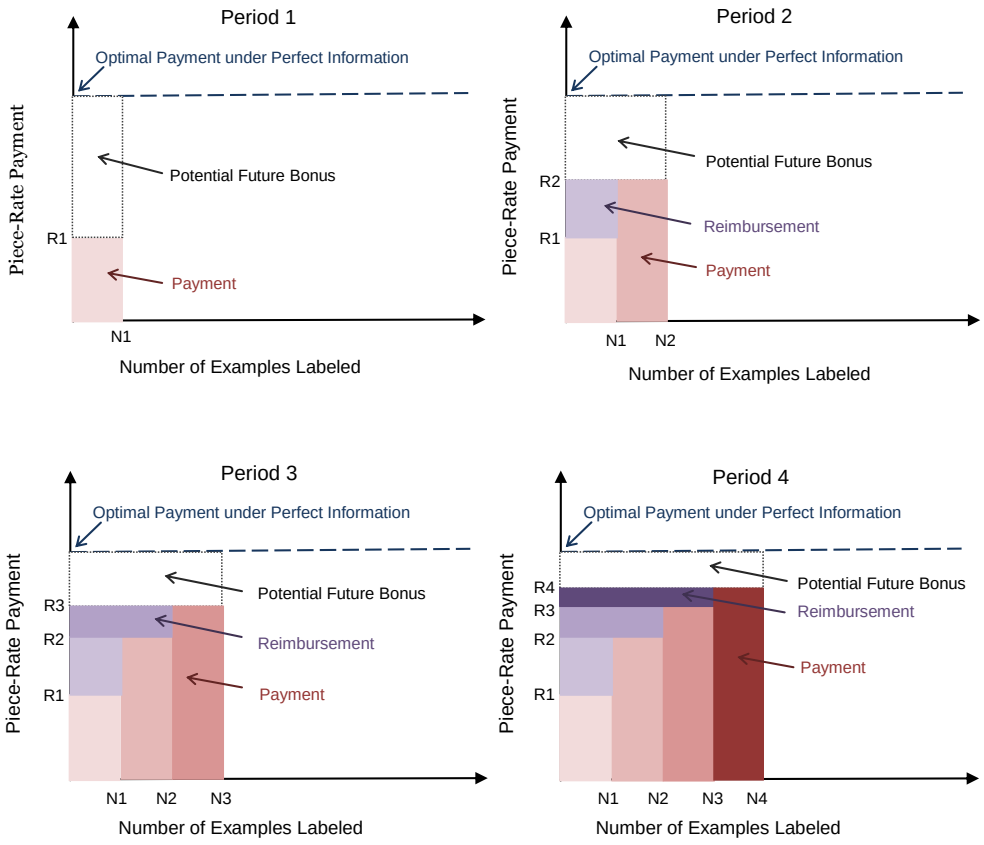


Figure 3: An illustration of Real-Time Payment

a pessimistic estimate of quality, we impose a low prior on the Bayesian estimation of the worker quality, assuming that the worker has an average quality that generates a very low payment. When a non-spammer worker submits answers, the distribution of quality increases, allowing the payment to increase over time. Then, as we get more data, we proceed with the payment estimations, reimbursing the workers for the underpayment in the prior periods. Given that payment over time is effectively a sum of random variables, Chernoff's bound applies in this case, guaranteeing that the uncertainty of payment decreases exponentially with the number of tasks submitted; therefore, our payment scheme converges into the real payment with exponentially low probability of overpaying.

Figure 3 illustrates the process: for a worker, we divide his lifetime into a set of small periods (for example, paying every 10 completed tasks). At the end of each period, we first pay the worker the deserved earnings in the current period, then reimburse the worker for the price difference between this period and the previous period for all the tasks completed before this period. Suppose that the piece-rate payment after the first 10

submissions is R_1, the worker gets paid $10 \cdot R_1$ at the end of Period 1. Now, the piece-rate payment after the second 10 submissions is R_2, we first pay the worker $10 \cdot R_2$ and then reimburse the "unpaid" part for the first 10 submissions by the price difference $10 \cdot (R_2 - R_1)$. Similarly, in the third period we examine if there are "unreimbursed" payments for the first and second periods, and do the same. We repeat the process until the worker reaches the end of the lifetime.

Notice that the strategy has the fortunate side-effect of incentivizing long-term participation: At any given time point, the worker improves payment by: (a) increasing the estimated pay rate $\hat{r}(\mathbf{e})$ (and bringing it closer to optimal payment $r(\mathbf{e})$), and (b) receiving a reimbursement payment (phrased as "bonus" to the worker) for all the underpayments in the prior periods. This strategy encourages good workers to work more, allowing us to understand better their quality. On the contrary, a worker that does not plan to work for long (therefore imposing to the service provider the risk of handling the unknown quality of the worker), receives a comparatively lower payment for the same amount and quality of work. So each incoming worker goes through a "reputation building" stage during which she is likely to be underpaid. However, as she completes more and more tasks, we will know better about her true quality and her payment will then increase.

7 Simulation Experiments

To test the performance of our proposed quality-adjusted piece-rate pricing strategy, we run a set of simulation experiments, where the workers are strategic actors who respond to changes in economic incentives. Simulation experiments are a powerful tool for modeling complicated market environments and conducting analyses under various parameter values (e.g., Chiang and Mookerjee, 2004; Adomavicius et al., 2009; Ketter et al., 2012). We describe below the setting for the simulations.

7.1 Market Simulation

We assume that the service provider receives a total of $N = 10,000$ binary labeling tasks from a client, who is willing to pay $S = 200$ for each successfully completed task. The SLA requirement sets the average misclassification cost at $\tau_c \leq 0.01$. The entire simulation process is described by the flowchart in Figure 4, which involves the decision-making of both the service provider and the crowdsourcing workers.

The Service Provider

The simulation process for the service provider is as follows:

S1) The service provider posts the job and announces a pricing scheme $r(\mathbf{e})$ to pay workers.

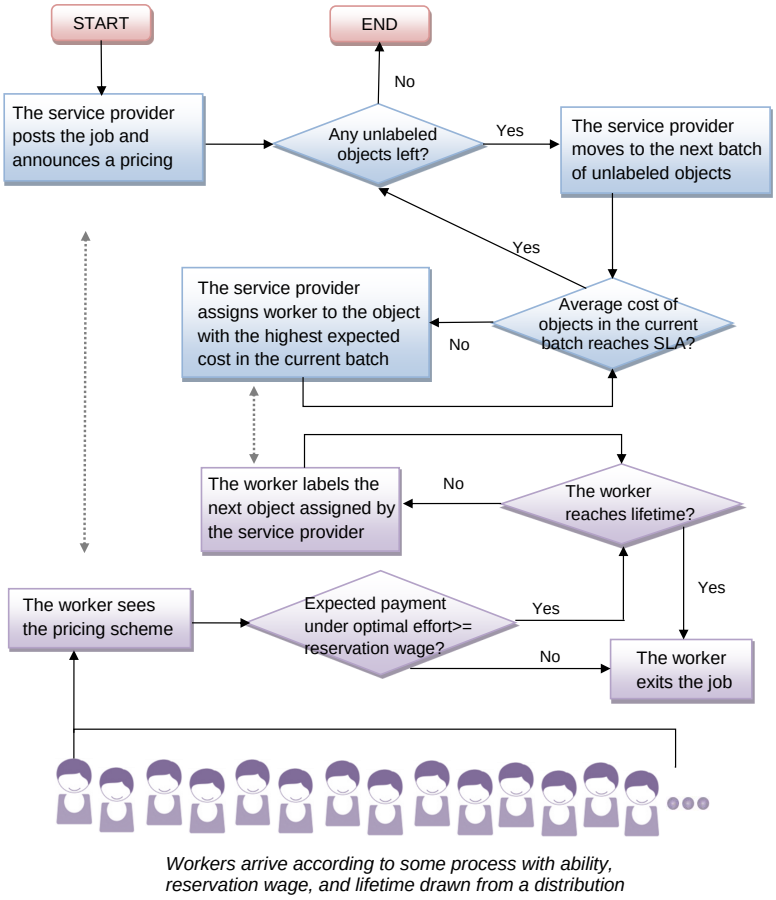


Figure 4: The entire simulation process

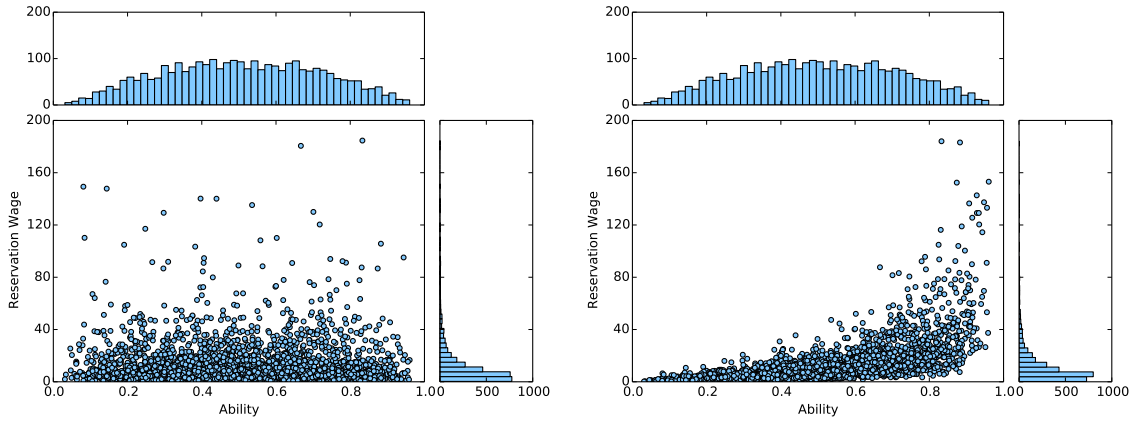
S2) If all the objects are successfully labeled, the service provider withdraws the job from the marketplace and releases the data to the client. We denote this time by T_F. Otherwise, the service provider moves to the next batch of size $N_b = 500$ of unlabeled objects.

S3) If the average cost of the objects in the current batch does not exceed τ_c, go to Step S2. Otherwise, the service provider assigns the next incoming worker to the object with the highest expected cost in the current batch.

The Workers

Every $t_a = 10$ time units, a new worker comes to the marketplace.[18] The ability a, reservation wage w, and lifetime h of the worker are generated as follows: (a) Draw v_a, v_w and v_h from a trivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, and then (b) Transform v_a, v_w, v_h to a, w, h by setting: $a = \text{logit}^{-1}(v_a)$, $w = \exp(v_w)$,

[18] For ease of exposition, we assume that worker's arrival follows a uniform process.



(a) $\rho_{12} = 0.0$ (b) $\rho_{12} = 0.8$

Figure 5: Histograms of Parameter Values

$h = \exp(v_h)$.[19] The parameters for the trivariate normal distribution are given below.

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix} = \begin{pmatrix} 0.0 \\ 2.0 \\ 5.0 \end{pmatrix} \quad ; \quad \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 & \rho_{13}\sigma_1\sigma_3 \\ \rho_{12}\sigma_1\sigma_2 & \sigma_2^2 & \rho_{23}\sigma_2\sigma_3 \\ \rho_{13}\sigma_1\sigma_3 & \rho_{23}\sigma_2\sigma_3 & \sigma_3^2 \end{pmatrix} = \begin{pmatrix} 1.0 & \rho_{12} & 0.0 \\ \rho_{12} & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$$

We test our strategy under two values of ρ_{12}, varying the correlation between worker reservation wage and worker ability: (a) $\rho_{12} = 0.0$, i.e., the quality of the workers and their reservation wages have no correlation, and (b) $\rho_{12} = 0.8$, i.e., quality of the workers and their reservation wages are positively correlated. Figures 5(a) and (b) show the scatter plot and the two histograms along x-axis and y-axis for $\rho_{12} = 0.0$ and $\rho_{12} = 0.8$, respectively. As we can see from the plots, when $\rho_{12} = 0.0$ the reservation wage is independent of worker's ability while when $\rho_{12} = 0.8$ the reservation wage tends to be higher as the ability of workers increases.

The simulation process for each worker after the drawing is presented below:

W1) The worker sees the pricing scheme $r(\mathbf{e})$ announced by the service provider and estimates her optimal effort which yields the maximum expected payment throughout her lifetime.

W2) If the expected payment per unit time under optimal effort is larger than worker's reservation wage, go to Step W3. Otherwise, the worker exits the job.

W3) If the worker reaches her lifetime, she stops working. Otherwise, she labels the next object assigned by the service provider.

[19] Following previous literature (Van den Berg, 1994; Bloemen and Stancanelli, 2001; Pannenberg, 2010; Wang et al., 2011), we assume that both worker's reservation wage and lifetime follow log-normal distributions.

7.2 Evaluation Criteria

The service provider is interested in optimizing its profit per unit time: $\frac{N \cdot S - \sum_{(k)} r(\mathbf{e}^{(k)}, n^{(k)})}{T_F}$, where $r(\mathbf{e}^{(k)}, n^{(k)})$ refers to the reward given to worker (k) who labels $n^{(k)}$ objects with exhibited error rates captured by $\mathbf{e}^{(k)}$, and T_F is the total time used to complete the job. Here we assume the labeling tasks all arrive at the same time but workers come continuously so that demand always exceeds supply. The profit per unit time captures the expected profit that the service provider can gain from each worker since workers' arrival process is exogenous and uniform.

7.3 Profit Maximizing Under Different Pricing Mechanisms

In Section 3, we assumed that additional effort put by the workers can increase the exhibited quality of submitted work but this increase is diminishing. We did not impose any explicit functional form on the relationship. To illustrate how our algorithm works, we set a specific quality function here, based on the ability value drawn previously: $e_{00} = e_{11} = \hat{g}(a,t) = 0.5 + 0.5a(1 - \exp(-\xi(t - \underline{t})^+)$. Here, ξ reflects how demanding the task is in terms of effort, and is set as $\xi = 3$. We use $(\cdot)^+$ to denote the maximum of 0 and a given value. Therefore, $\underline{t}$ is the minimum effort that can be exerted. We set $\underline{t} = 0.5$ in our simulation. This minimum effort assumption[20] is realistic in the sense that workers need to spend some time in reading the actual content of the question before making any sensible choice, and prevents the extreme case where the workers invest zero effort on each task and submit an infinite number of labels.

The rationale behind this function specification hinges on the following two key considerations. First, workers differ in their inherent ability to perform specific tasks, which imposes an upper bound on their exhibited quality levels. Second, workers can vary the amount of effort they put into each task, and thus the quality of their submissions. The value of function $\hat{g}(\cdot)$ is bounded between 0.5 and 1 ($0 \leq a \leq 1$).

We compare our proposed pricing scheme with two other pricing schemes commonly adopted in crowdsourcing platforms, namely piece-rate pricing with block, and piece-rate pricing with block and penalty. Next, we explain what each pricing scheme means, and analyze the strategic behavior of workers and profit-maximization decision of the service provider under each one.

7.3.1 Piece-Rate Pricing with Block (PR-B)

Piece-rate pricing with block is widely used in the today's crowdsourcing platforms. The way it works is that the service provider announces a pre-specified error tolerance level ε so that any worker with an error rate on her submitted labels exceeding ε will be blocked from participating and receive no payment. Under this scheme, as long as the worker meets the error threshold, all of his submitted labels will get paid. In reality, the

[20] Lu et al. (2009) imposed the minimum effort assumption in their paper.

service provider can also vary the value of error tolerance level ε to obtain a desirable performance. Suppose that the service provider decides to employ this pricing scheme and compensate all the workers who are not blocked, the optimal piece-rate price r_b^* and error threshold ε_b^* are given by the following maximization problem:

$$\begin{aligned}
&(r_b^*, \varepsilon_b^*) = \arg\max_{r,\varepsilon} \int_0^\infty \int_0^\infty \int_{\hat{g}(a,\infty)\leq\varepsilon} (v(\hat{g}(a,t^*)) - r) \cdot h/t^* \cdot f_{A,W,H}(a,w,h)\mathrm{d}a\mathrm{d}w\mathrm{d}h \\
&\text{subject to} \qquad t^* = \arg\min_t \hat{g}(a,t) \leq \varepsilon \qquad \text{(IC)} \\
&\text{and} \qquad \frac{r}{t^*} > w \qquad \text{(IR)}
\end{aligned} \tag{6}$$

7.3.2 Piece Rate Pricing with Block and Penalty (PR-BP)

Besides the option of blocking badly performed workers, many crowdsourcing platforms also allow the service provider to impose a penalty for every piece of work detected as incorrect. A natural way to enforce this is to only pay the rate for those correct answers. Under this scheme, the optimal piece rate r_{bp}^* and error threshold ε_{bp}^* are given by the following:

$$\begin{aligned}
&(r_{bp}^*, \varepsilon_{bp}^*) = \arg\max_{r,\varepsilon} \int_0^\infty \int_0^\infty \int_{\hat{g}(a,\infty)\leq\varepsilon} (v(\hat{g}(a,t^*)) - r \cdot \hat{g}(a,t^*)) \cdot h/t^* \cdot f_{A,W,H}(a,w,h)\mathrm{d}a\mathrm{d}w\mathrm{d}h \\
&\text{subject to} \qquad t^* = \arg\max_{\hat{g}(a,t)\leq\varepsilon} \frac{\hat{g}(a,t)}{t} \qquad \text{(IC)} \\
&\text{and} \qquad \frac{r \cdot \hat{g}(a,t^*)}{t^*} > w \qquad \text{(IR)}
\end{aligned} \tag{7}$$

7.3.3 Quality-Adjusted Piece-Rate Pricing (QA-PR)

Our proposed quality-adjusted piece-rate pricing mechanism discourages low-quality workers by offering them low wage rates instead of a harsh block or rejection. The optimal pricing scheme $r_q^*(\mathbf{e}) = \alpha^* \cdot v(\mathbf{e})$ is given by Equation (5) in Section 6.2. We can rephrase the maximization problem for this specific simulation setting as follows:

$$\begin{aligned}
&\alpha^* = \arg\max_{\alpha} \int_0^\infty \int_0^\infty \int_a (1-\alpha) \cdot v(\hat{g}(a,t^*)) \cdot h/t^* \cdot f_{A,W,H}(a,w,h)\mathrm{d}a\mathrm{d}w\mathrm{d}h \\
&\text{subject to} \qquad t^* = \arg\max_t \frac{v(\hat{g}(a,t))}{t} \qquad \text{(IC)} \\
&\text{and} \qquad \frac{\alpha \cdot v(\hat{g}(a,t^*))}{t^*} > w \qquad \text{(IR)}
\end{aligned} \tag{8}$$

7.4 Simulation Results

To ensure the robustness of the results, each experiment is replicated 20 times and the results are averaged over all replicas. Note that the comparison is made under the same quality estimation and and label allocation strategy, so any difference in performance would directly reflect the effect of employing different pricing

Correlation	Scheme	Parameter Values	Time Used	Total Payment	Unit Time Profit
$\rho_{12} = 0.0$	PR-B	$r_b^* = 10.5, \varepsilon_b^* = 0.23$	19046	521925	77.84
	PR-BP	$r_{bp}^* = 14.3, \varepsilon_{bp}^* = 0.23$	17528	564050	82.06
	QA-PR	$\alpha^* = 0.365$	11629	765920	106.36
$\rho_{12} = 0.8$	PR-B	$r_b^* = 14.9, \varepsilon_b^* = 0.24$	37519	817977	31.57
	PR-BP	$r_{bp}^* = 21.5, \varepsilon_{bp}^* = 0.24$	30879	884508	36.24
	QA-PR	$\alpha^* = 0.545$	13464	1174680	61.50

Table 2: Simulation results for the three pricing schemes

schemes.

Table 2 shows the final simulation results. The relative superiority of the three pricing schemes is consistent across the two correlation conditions: QA-PR > PR-BP ≈ PR-B. Our QA-PR strategy outperforms the second-best strategy PR-BP by 29.6% when there is no correlation between worker ability and reservation age, and by 69.7% when workers' ability levels and their reservation wages are positively correlated (i.e., $\rho_{12} = 0.8$). There are several points worth noting in our results. First, the time taken to achieve the SLA requirement is much shorter under QA-PR scheme compared to the other two. By offering higher pay for better quality, QA-PR attracts more good workers and induces a higher level of effort exerted on each task. The improvement in the overall quality of the submitted work significantly reduces the number of labels required to meet the SLA. Although the total amount of money paid to workers is slightly greater, QA-PR is able to achieve the highest profit per unit time for the service provider by shortening the execution time. Second, the unit time profits under positive correlation are overall lower than under no correlation, across all the three pricing schemes. This is not too surprising, because when worker ability and reservation wage are independently distributed, the service provider is still able to get some high-ability workers with low reservation wages; however, this happens less often under the positive-correlation condition. It should be noted here that a positive correlation between workers' ability levels and reservation wages is more realistic since high-ability workers are likely to have more outside options. Third, the advantage of QA-PR is more pronounced under positive correlation. This is because both PR-B and PR-BP pay workers at a single price, so the service provider has to keep the price low to accommodate the cost of dealing with low-quality labels. When good workers have higher wage expectations, PR-B ends up with only low-ability workers remaining. However, QA-PR is able to keep the high-ability workers because it provides performance bonuses to those who submit results of higher quality.

8 Discussion and Conclusions

To the best of our knowledge, this study is the first to propose a comprehensive framework for managing and paying crowdsourced workers. Our approach integrates viewpoints from both managerial and technical perspectives to explore a typical crowdsourcing setting: strategic workers with heterogeneous levels of ability come into marketplace and decide whether to participate and how much effort to put forth according to the payment scheme; and the service provider aims to achieve a certain level of quality assurance while minimizing the labor costs. In this study, we consider the entire decision process of a service provider, from quality estimation to resource allocation and to pricing scheme design. Our work fills the gap between the machine learning literature, which focuses on the information acquisition components of the crowdsourcing process without paying attention to the strategic behaviors of workers, and the management and economics literature, which emphasizes the economic incentives of workers but fails to consider the informational aspects of the process. The contribution of this paper is threefold: First, we present a novel strategy to separate the systematic biases from unrecoverable errors workers exhibit, allowing us to better evaluate the quality of the workers. Second, we introduce a dynamic resource allocation strategy that prioritizes labels on objects with high expected misclassification costs. The selective labeling approach is able to achieve the same level of data quality using fewer label acquisitions. Third, we propose a quality-adjusted piece-rate pricing scheme that accommodates both adverse selection and moral hazard of workers, and conduct simulated experiments to demonstrate its superior performance over two commonly used piece-rate pricing schemes. As illustrated further in Appendix 8.2, our work serves as a fundamental quality control block for a variety of tasks, ensuring that the outcome of crowdsourced production reaches the quality levels desired by the employers.

8.1 Practical Implications

Crowdsourcing is rapidly becoming a commonly used tool across many firms, from Fortune-500 companies to startups. Amazon has been using paid crowdsourcing for more than 10 years now to de-duplicate products in the catalogs uploaded to their platform by merchants. Metaweb, acquired by Google in 2010, has been using paid crowdsourcing to create Freebase (Kochhar et al., 2010), which is the base for the Google Knowledge Graph. Microsoft has built the Microsoft Universal Human Relevance System (UHRS)[21] to evaluate and improve the results in Bing, their search engine. Facebook is using crowdsourcing for content moderation, and Twitter is using Amazon Mechanical Turk[22] to improve their real-time event detection. Many other companies use crowdsourcing either directly or through an intermediary (cf., the participants in the CrowdConf conference). Firms are attracted to crowdsourcing because of the dynamic nature of hiring,

[21] http://www.signalprocessingsociety.org/technical-committees/list/sl-tc/spl-nl/2013-05/interview-crowdsourcing/
[22] http://engineering.twitter.com/2013/01/improving-twitter-search-with-real-time.html

allowing quick scaling up and quick downsizing of the workforce, according to the needs of the company, with reaction times within hours or even minutes. Crowdsourcing also provides a great way to help unemployed and underemployed people. For example, Samasource, a nonprofit organization, divides projects from clients into "microwork" for workers in developing regions (Gino and Staats, 2012).

Unfortunately, quality control remains an issue and most existing solutions simply attempt to screen workers through multiple gold tests, and reject unqualified workers. The use of more fine-grained strategies contributes to better utilization of crowdsourcing. The wider adoption of crowdsourcing can also lower the barrier-to-entry for workers with no prior experience and reputation. Since there is no interview stage, and workers can join the workforce at will, it becomes easier for unemployed people to find work and prove their skills while working. Our approach can automatically provide a performance measurement for each worker, and help build an honest reputation feedback mechanism that can facilitate the creation of a healthy, well-operating crowdsourcing marketplace. Our pricing scheme further ensures that workers are paid according to the value they contribute, incentivizing employers to open more of their tasks to "crowd workers". Moreover, since our pricing schemes ensures an (eventually) fair payment policy, good workers are also encouraged to keep working for long periods of time, which reduces the churning of good workers—one of the major problems for any employer, and a particularly acute one in crowdsourcing.

8.2 Limitations and Future Work

This study has several limitations and opens up opportunities for further research. In our study, we assume that the ability of workers does not change over time. However, for many types of tasks in practice, there might be either learning effects or tiredness effects, which lead to possible fluctuations in the ability and the exhibited quality of workers. To account for this, we can apply a particle filtering method to track the changes in worker quality (Crisan and Doucet, 2002; Donmez et al., 2010) and choose the size of window for aggregation appropriately (Aperjis and Johari, 2010). We also assume zero knowledge about the worker quality and start with an uninformative (and conservative) prior for quality estimation. It is possible that sometimes the service provider has access to the past performance of workers on other tasks. The service provider can utilize the inter-category reputation of workers (Kokkodis and Ipeirotis, 2013) by adjusting the prior belief of workers' quality in the quality estimation framework, which could potentially reduce the inefficiencies caused by the inaccurate quality estimates especially during the early stage.

In our problem setting, there is only one service provider with monopolistic power. In real-life scenarios, the service provider faces competition from other providers or employers. The competitiveness of the labor market can be somewhat captured by workers' reservation wages: the more competitive the market, the more outside options workers have, and the higher their wage expectations. We also assume that the service

provider knows or can estimate relatively well the joint distribution of worker abilities, reservation wage, and lifetimes. In reality, the service provider needs to learn this distribution, especially in an environment where workers arrive and leave the market freely and continuously. This estimation task should be studied carefully across real labor marketplaces in future work. Another assumption we make is that the crowdsourcing workers are purely selfish and only respond to extrinsic motivations. Previous studies (e.g., Rogstadius et al., 2011) show that workers may also be driven by intrinsic motivation and put in effort even when monetary rewards are low or absent. Therefore, our quality-adjusted pricing scheme is more appropriate to tasks that are less interesting or have lower social value so that the extrinsic motivators dominate in workers' decision-making. Future studies should conduct experiments with workers recruited from crowdsourcing marketplaces to examine how they react to different quality control schemes and incentives, as well as the influence of task characteristics on the effectiveness of the quality-adjusted pricing.

Despite these limitations, we believe that our current work provides a solid foundation on which future work can build. Furthermore, our work can be used immediately by interested parties, allowing easier management of crowdsourced workers, and therefore the development of further interesting applications, enabled by crowdsourcing.

References

Adomavicius, Gediminas, Alok Gupta, Dmitry Zhdanov. 2009. Designing intelligent software agents for auctions with limited information feedback. *Information Systems Research* **20**(4) 507–526.

Agranov, Marina, Chloe Tergiman. 2013. Incentives and compensation schemes: An experimental study. *International Journal of Industrial Organization* **31**(3) 238–247.

Akerlof, G. A. 1970. The market for lemons: Quality uncertainty and the market mechanism. *Quarterly Journal of Economics* **84**(3) 488–500.

Anand, Krishnan S, M Fazil Pac, Senthil Veeraraghavan. 2011. Quality-speed conundrum: trade-offs in customer-intensive services. *Management Science* **57**(1) 40–56.

Aperjis, Christina, Ramesh Johari. 2010. Optimal windows for aggregating ratings in electronic marketplaces. *Management Science* **56**(5) 864–880.

Bachrach, Yoram, Thore Graepel, Tom Minka, John Guiver. 2012. How to grade a test without knowing the answers—a bayesian graphical model for adaptive crowdsourcing and aptitude testing. *arXiv preprint arXiv:1206.6386* .

Berger, Roger L. 1982. Multiparameter hypothesis testing and acceptance sampling. *Technometrics* **24**(4) 295–300.

Bernstein, M. S., G. Little, R. C. Miller, B. Hartmann, M. S. Ackerman, D. R. Karger, D. Crowell, K. Panovich. 2010. Soylent: A word processor with a crowd inside. *Proceedings of the 23th annual ACM Symposium on User Interface Software and Technology*. 313–322.

Bloemen, Hans G, Elena GF Stancanelli. 2001. Individual wealth, reservation wages, and transitions into employment. *Journal of Labor Economics* **19**(2) 400–439.

Bonner, Sarah E, Reid Hastie, Geoffrey B Sprinkle, S Mark Young. 2000. A review of the effects of financial incentives on performance in laboratory tasks: Implications for management accounting. *Journal of Management Accounting Research* **12**(1) 19–64.

Carpenter, B. 2008. Multilevel Bayesian models of categorical data annotation. Available at http://lingpipe-blog.com/lingpipe-white-papers/.

Chiang, I Robert, Vijay S Mookerjee. 2004. A fault threshold policy to manage software development projects. *Information Systems Research* **15**(1) 3–21.

Cohn, David, Les Atlas, Richard Ladner. 1994. Improving generalization with active learning. *Machine learning* **15**(2) 201–221.

Crisan, Dan, Arnaud Doucet. 2002. A survey of convergence results on particle filtering methods for practitioners. *Signal Processing, IEEE Transactions on* **50**(3) 736–746.

Crocker, Linda, James Algina. 2006. *Introduction to Classical and Modern Test Theory*. Wadsworth.

Dawid, A. P., A. M. Skene. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics* **28**(1) 20–28.

Dellarocas, Chrysanthos. 2003. The digitization of word of mouth: Promise and challenges of online feedback mechanisms. *Management science* **49**(10) 1407–1424.

DeMars, Christine. 2010. *Item Response Theory*. Oxford University Press.

Dodge, Harold F. 1973. *Notes on the Evolution of Acceptance Sampling*. American Society for Quality Control.

Donmez, Pinar, Jaime Carbonell, Jeff Schneider. 2010. A probabilistic framework to learn from multiple annotators with time-varying accuracy. *SIAM International Conference on Data Mining (SDM)*. 826–837.

Eisenhardt, Kathleen M. 1989. Agency theory: An assessment and review. *Academy of Management Review* **14**(1) 57–74.

Gino, Francesca, Bradley R Staats. 2012. The microwork solution. *Harvard Business Review* **90**(12) 92–96.

Harris, Christopher. 2011. You're hired! an examination of crowdsourcing incentive models in human resource tasks. *WSDM Workshop on Crowdsourcing for Search and Data Mining (CSDM)*. 15–18.

Hölmstrom, Bengt. 1979. Moral hazard and observability. *The Bell Journal of Economics* 74–91.

Holmstrom, Bengt, Paul Milgrom. 1991. Multitask principal-agent analyses: Incentive contracts, asset ownership, and job design. *Journal of Law, Economics, & Organization* 24–52.

Ipeirotis, Panagiotis G. 2010. Analyzing the amazon mechanical turk marketplace. *XRDS: Crossroads, The ACM Magazine for Students* **17**(2) 16–21.

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

Ipeirotis, Panagiotis G, Foster Provost, Jing Wang. 2010. Quality management on amazon mechanical turk. *Proceedings of the ACM SIGKDD Workshop on Human Computation*. ACM, 64–67.

Jensen, Michael C, William H Meckling. 1976. Theory of the firm: Managerial behavior, agency costs and ownership structure. *Journal of Financial Economics* **3**(4) 305–360.

Ketter, Wolfgang, John Collins, Maria Gini, Alok Gupta, Paul Schrater. 2012. Real-time tactical and strategic sales management for intelligent agents guided by economic regimes. *Information Systems Research* **23**(4) 1263–1283.

Kittur, A., B. Smus, S. Khamkar, R. E. Kraut. 2011. CrowdForge: Crowdsourcing complex work. *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*. 43–52.

Kochhar, Shailesh, Stefano Mazzocchi, Praveen Paritosh. 2010. The anatomy of a large-scale human computation engine. *Proceedings of the ACM SIGKDD Workshop on Human Computation*. ACM, 10–17.

Kokkodis, Marios, Panagiotis G Ipeirotis. 2013. Have you done anything like that? Predicting performance using inter-category reputation. *Proceedings of the ACM International Conference on Web Search and Data Mining*. ACM, 435–444.

Kostami, Vasiliki, Sampath Rajagopalan. 2013. Speed-quality trade-offs in a dynamic model. *Manufacturing & Service Operations Management* **16**(1) 104–118.

Kulkarni, Anand P., Matthew Can, Bjoern Hartmann. 2011. Turkomatic: Automatic recursive task and workflow design for mechanical turk. *Proceedings of the 2011 Annual Conference Extended Abstracts on Human Factors in Computing Systems*. 2053–2058.

Lazear, Edward P. 1986. Salaries and piece rates. *Journal of Business* 405–431.

Lazear, Edward P. 2000. Performance pay and productivity. *American Economic Review* **90**(5) 1346–1361.

Lazear, Edward P, Sherwin Rosen. 1979. Rank-order tournaments as optimum labor contracts. NBER Working Paper No. 401.

Lewis, David D, William A Gale. 1994. A sequential algorithm for training text classifiers. *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Springer-Verlag New York, Inc., 3–12.

Little, G., L. B. Chilton, M. Goldman, R. Miller. 2010. Turkit: Human computation algorithms on mechanical turk. *Proceedings of the 23th annual ACM Symposium on User Interface Software and Technology*. 57–66.

Lizotte, Daniel J, Omid Madani, Russell Greiner. 2002. Budgeted learning of nailve-bayes classifiers. *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 378–385.

Lu, Lauren Xiaoyuan, Jan A Van Mieghem, R Canan Savaskan. 2009. Incentives for quality through endogenous routing. *Manufacturing & Service Operations Management* **11**(2) 254–273.

Malone, T. W., R. Laubacher, C. Dellarocas. 2010. Harnessing crowds: Mapping the genome of collective intelligence. Available at http://ssrn.com/abstract=1381502.

Mason, Winter, Duncan J Watts. 2010. Financial incentives and the performance of crowds. *ACM SigKDD Explorations Newsletter* **11**(2) 100–108.

Moore, James C, Andrew B Whinston. 1986. A model of decision-making with sequential information-acquisition (part 1). *Decision Support Systems* **2**(4) 285–307.

Moore, James C, Andrew B Whinston. 1987. A model of decision-making with sequential information-acquisition (part 2). *Decision Support Systems* **3**(1) 47–72.

Nagin, Daniel, James Rebitzer, Seth Sanders, Lowell Taylor. 2002. Monitoring, motivation and management: The determinants of opportunistic behavior in a field experiment. Tech. rep., National Bureau of Economic Research.

Nov, Oded, Ofer Arazy, David Anderson. 2011. Dusting for science: motivation and participation of digital citizen science volunteers. *Proceedings of the 2011 iConference*. ACM, 68–74.

Olmos, Marta Fernández, Jorge Rosell Martínez. 2010. The quality-quantity trade-off in the principal-agent framework. *Agricultural Economics Review* **11**(1) 57–68.

Paarsch, Harry J, Bruce Shearer. 2000. Piece rates, fixed wages, and incentive effects: Statistical evidence from payroll records. *International Economic Review* **41**(1) 59–92.

Pannenberg, Markus. 2010. Risk attitudes and reservation wages of unemployed workers: evidence from panel data. *Economics Letters* **106**(3) 223–226.

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

Resnick, Paul, Ko Kuwabara, Richard Zeckhauser, Eric Friedman. 2000. Reputation systems. *Communications of the ACM* **43**(12) 45–48.

Rogstadius, Jakob, Vassilis Kostakos, Aniket Kittur, Boris Smus, Jim Laredo, Maja Vukovic. 2011. An assessment of intrinsic and extrinsic motivation on task performance in crowdsourcing markets. *Proceedings*

of the Fifth International AAAI Conference on Weblogs and Social Media. 321–328.

Roy, Nicholas, Andrew McCallum. 2001. Toward optimal active learning through sampling estimation of error reduction. *Proceedings of the Eighteenth International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc., 441–448.

Saar-Tsechansky, Maytal, Prem Melville, Foster Provost. 2009. Active feature-value acquisition. *Management Science* **55**(4) 664–684.

Saar-Tsechansky, Maytal, Foster Provost. 2004. Active sampling for class probability estimation and ranking. *Machine learning* **54**(2) 153–178.

Schilling, Edward G. 1982. *Acceptance Sampling in Quality Control*. CRC Press.

Sheng, V. S., F. Provost, P. G. Ipeirotis. 2008. Get another label? Improving data quality and data mining using multiple, noisy labelers. *Proceedings of the Fourteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2008)*. 614–622.

Snow, Rion, Brendan O'Connor, Daniel Jurafsky, Andrew Y Ng. 2008. Cheap and fast—but is it good? Evaluating non-expert annotations for natural language tasks. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 254–263.

Stewart, Osamuyimen, David Lubensky, Juan M. Huerta. 2010. Crowdsourcing participation inequality: a scout model for the enterprise domain. *Proceedings of the ACM SIGKDD Workshop on Human Computation*. ACM, 30–33.

Van den Berg, Gerard J. 1994. The effects of changes of the job offer arrival rate on the duration of unemployment. *Journal of Labor Economics* 478–498.

Wais, Paul, Shivaram Lingamneni, Duncan Cook, Jason Fennell, Benjamin Goldenberg, Daniel Lubarov, David Marin, Hari Simons. 2010. Towards building a high-quality workforce with Mechanical Turk. *Proceedings of Computational Social Science and the Wisdom of Crowds (NIPS)*. 1–5.

Wang, Jing, Siamak Faridani, P Ipeirotis. 2011. Estimating the completion time of crowdsourced tasks using survival analysis models. *Crowdsourcing for Search and Data Mining (CSDM 2011)*. 31–34.

Welinder, Peter, Steve Branson, Serge Belongie, Pietro Perona. 2010. The multidimensional wisdom of crowds. *Advances in Neural Information Processing Systems* **23** 2424–2432.

Wetherill, G. B., W. K. Chiu. 1975. A review of acceptance sampling schemes with emphasis on the economic aspect. *International Statistical Review/Revue Internationale de Statistique* 191–210.

Yin, Ming, Yiling Chen, Yu-An Sun. 2013. The effects of performance-contingent financial incentives in online labor markets. *Proceedings of the 27th Conference on Artificial Intelligence (AAAI)*. 1191–1197.

Zheng, Zhiqiang, Balaji Padmanabhan. 2006. Selectively acquiring customer information: A new data

acquisition problem and an active learning-based solution. *Management Science* **52**(5) 697–712.

Appendix: Importance of Quality Control for Multiple Choice Questions

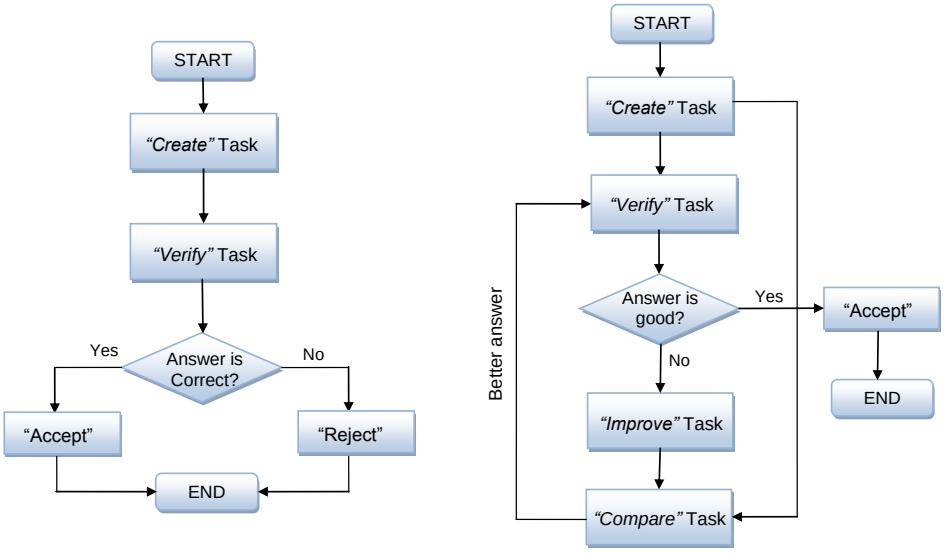
Our scheme can be directly applied to multiple choice questions, which already captures a large number of tasks that are crowdsourced today (e.g., image moderation, spam detection, restaurant rating, etc.). We would like to stress, though, that quality control mechanisms for multiple choice questions are at the heart of many other, more complex, tasks that are also executed in crowdsourcing platforms. Below we give some representative examples.

- **Open-ended questions with correct or incorrect answers**: Consider the task that asks workers to collect information about a given topic; for example, "collect URLs that discuss massive online education courses and their impact on MBA programs." For this type of task, it is usually difficult or feasible to enumerate all the correct answers, therefore it is not possible to control the quality of the task using quality control for multiple choice answers directly. However, once an answer is provided, we can easily check its correctness, *by instantiating another task*, asking a binary choice question: "Is this submitted URL about massive online education courses and their impact on MBA programs?" Thereby, one can break the task into two tasks: A *"Create"* task, in which one or more workers submit free-form answers, and a *"Verify'* task, in which another set of workers vets the submitted answers, and classifies them as either "correct" or "incorrect". Figure 6(a) illustrates the structure: the "Verify" task controls the quality of the "Create" task; the quality of the "Verify" task is then controlled using a quality control mechanism for multiple choice questions, similar to the one presented in this paper.
- **Varying degrees of correctness**: There are some tasks whose free-form answers are not right or wrong but have varying degrees of correctness or goodness (e.g., "generate a transcript from this manuscript," "describe and explain the image below in at least three sentences"). In such a setting, treating the submitted answers as "correct" or "incorrect" might be inefficient: a rejected answer would be completely discarded, although it is often possible to leverage the low-quality answers to get better results, by simply iterating. Past work (Little et al., 2010) has shown the superiority of the iterative paradigm by demonstrating that workers were able to create image descriptions of excellent quality, even though no single worker put any significant effort in the task. Figure 6(b) illustrates the iterative process. There are four subtasks: The *"Create"* task, in which free-form answers are submitted, the *"Improve"* task, in which workers are asked to improve an existing answer, the *"Compare"* task, in which workers are required to compare two answers and select the better one, and the *"Verify"* task, in which workers decide whether the quality of the answers[23] is satisfactory. In this case, the "Compare"

[23] "Verify" task either accepts input directly from the "Create" task or gets the better answer returned by "Compare" task.

and "Verify" are multiple choice tasks, and one can use the mechanisms presented in this paper to control the quality of the submitted answers (and of the participating workers). In turn, the "Create" and "Improve" tasks are controlled by the "Verify" and "Compare" tasks, as one can measure the probability that a worker submits an answer of high quality, or the probability that a worker is able to improve an existing answer.

- **Complex tasks using workflows**: Initial applications of paid crowdsourcing focused primarily on simple and routine tasks. However, many tasks in our daily life are much more complicated (e.g., "proofread the following paragraph from the draft of a student's essay," "write a travel guide about New York City") and recently, there is an increasing trend to accomplish such tasks by dividing complex tasks into a set of microtasks, using workflows. For example, Bernstein et al. (2010) introduced the *"Find-Fix-Verify pattern"* to split text editing tasks into three simple operations: find something that needs fixing, fix the problem if there is one, and verify the correctness of the fix. Again, this task ends up having quality control through a set of multiple choice tasks (verification of the fix, verification that something needs fixing). In other cases, Kittur et al. (2011) described a framework for parallelizing the execution of such workflows and Kulkarni et al. (2011) moved a step further by allowing workers themselves to design the workflow. As in the case of other tasks that are broken into workflows of micro-tasks, the quality of these complex tasks can be guaranteed by applying our quality control scheme to each single micro-task, following the paradigms described above.



(a) Correct or incorrect answers (b) Varying degrees of correctness

Figure 6: Workflows for two types of tasks