More for less: adaptive labeling payments in online labor markets

Tomer Geva¹ · Maytal Saar-Tsechansky² · Harel Lustiger¹

Received: 15 December 2017 / Accepted: 16 April 2019 / Published online: 13 June 2019
© The Author(s) 2019

Abstract

In many predictive tasks where human intelligence is needed to label training instances, online crowdsourcing markets have emerged as promising platforms for large-scale, cost-effective labeling. However, these platforms also introduce significant challenges that must be addressed in order for these opportunities to materialize. In particular, it has been shown that different trade-offs between payment offered to labelers and the quality of labeling arise at different times, possibly as a result of different market conditions and even the nature of the tasks themselves. Because the underlying mechanism giving rise to different trade-offs is not well understood, for any given labeling task and at any given time, it is not known which labeling payments to offer in the market so as to produce accurate models cost-effectively. Importantly, because in these markets the acquired labels are not always correct, determining the expected effect of labels acquired at any given payment on the improvement in model performance is particularly challenging. Effective and robust methods for dealing with these challenges are essential to enable a growing reliance on these promising and increasingly popular labor markets for large-scale labeling. In this paper, we first present this new problem of Adaptive Labeling Payment (ALP): how to learn and sequentially adapt the payment offered to crowd labelers before they undertake a labeling task, so as to produce a given predictive performance cost-effectively. We then develop an ALP approach and discuss the key challenges it aims to address so as to yield consistently good performance. We evaluate our approach extensively over a wide variety of market conditions. Our results demonstrate that the ALP method we propose yields significant cost savings and robust performance across different settings. As such, our ALP approach can be used as a benchmark for future mechanisms to determine cost-effective selection of labeling payments.

Communicated by Dr. Johannes Fuernkranz.

The first two authors have contributed equally to this work and are listed in alphabetical order.

✉ Maytal Saar-Tsechansky
maytal@mail.utexas.edu

Extended author information available on the last page of the article

Keywords Machine learning · Supervised learning · Label acquisition · Crowdsourcing · Online labor markets · Adaptive labeling payments

1 Introduction

Predictive modeling has radically impacted a wide variety of industries, becoming integral to the operations and competitive strategies of firms and giving rise to entirely new business platforms. Supervised learning of predictive models requires labeled training data—namely, instances for which the dependent variable value (label) is known. However, in many important applications, costly human intelligence is also needed to determine the labels for training data. Such labeling tasks include, for example, image classification tasks (e.g., whether an image contains a car, for application of surveillance, autonomous driving, and scene understanding), and text classification problems (e.g., whether a text contains hate remarks, or is humorous or sarcastic). Many similar labeling tasks are relatively intuitive and simple for humans to perform, but may require a large number of training examples for supervised learning to yield good performance. For such tasks, online crowdsourcing marketplaces, such as Amazon Mechanical Turk (AMT), have emerged as promising platforms for large-scale labeling, offering unprecedented scalability and immediacy. Markets such as AMT offer substantial savings and agility by allowing employers (“requesters”) to offer simple micro-tasks to hundreds of thousands of online workers simultaneously, thereby allowing labels to be acquired cheaply and relatively quickly. In these markets, requesters typically post the task description and payment offered to all workers who meet certain criteria, and workers can review this information when choosing which tasks to undertake.

A fundamental challenge for the cost-effective use of such markets, however, is that for any given task and at any given time, it is not known what level of labeling quality can be acquired on the market for any given payment per label. Prior studies have found that different prevailing trade-offs arise between payment and quality for different tasks and at different times (e.g., Kazai 2011; Kazai et al. 2013; Mason and Watts 2010). Yet, the mechanisms giving rise to these different trade-offs are not well understood, and can be affected by a host of properties such as the labeling task itself and the market conditions—such as the competing tasks and the composition and availability of workers on the market at any given time.

Because the cumulative costs of routine use of crowdsourcing labor markets for labeling can be substantial, a robust, data-driven methodology for identifying advantageous payments for labeling that yield a desired predictive performance in a cost-effective manner is essential.

In this paper, we refer to this problem as Adaptive Labeling Payment (ALP). Specifically, given a budget for label acquisition, ALP refers to data-driven, sequential learning and adaptation of payments for labeling. At each step, i , labels for b training instances are acquired on the market, and a payment p^i for labeling each unlabeled instance is offered to crowd workers, before they undertake a labeling task. The objective for selecting payments $p^1, p^2, \dots, p^I$ is that the ultimate predictive model induced from the set L of labeled instances, acquired over I steps, yields the

best performance, without exceeding the acquisition budget (see the complete formal problem formulation in Sect. 3.1).

The *market* we consider here is an online crowdsourcing labor market, where a labeling task for a given payment is offered to all workers, and where no prior knowledge on an individual worker's performance for the particular task may be available (e.g., when a new worker is hired). In addition, we consider markets such as AMT, where the population of available workers and the competitive settings (e.g., alternative tasks and workers) may also vary over time.

Note that at the outset no data is available. Thus, payments are adapted sequentially, after each step, based on the data acquired thus far. Furthermore, because the cost-effectiveness of a given payment for labeling can change over time due to changes in the population of workers and competing tasks, the payment offered per label to workers can also be revised to adapt to these changes.

The ALP problem is related to active learning (Lewis and Gale 1994; Abe and Mamitsuka 1998; Kong and Saar-Tsechansky 2014), and to prior work about online labor markets (e.g., Raykar et al. 2010; Wang et al. 2017). However, as we discuss in detail below, both prior streams of work did not aim to adaptively identify advantageous payments per label to improve model performance cost-effectively. The most closely related prior work considered theoretical settings in which individual labelers produce a predetermined level of quality and require a predetermined price that is known to the requester (Yang and Carbonell 2012); however, such settings do not hold in many real-world crowdsourcing labor markets (e.g., Amazon Mechanical Turk) that we consider here, where new workers with no prior history are frequently encountered, workers' labeling quality for different payments are unknown a priori and can change over time, and where the tasks are offered to all workers on the market (or to all worker who meet a certain criteria).

The ALP problem presents several challenges, which we discuss in the remainder of the paper. Perhaps the most fundamental challenge pertains to the objective of identifying the labeling payments that cost-effectively improve predictive performance. The choice of payment per label that will yield the most cost-effective improvement in model performance is ultimately affected by a host of factors that impact model performance and its cost. These factors include the quality of the labels that can be acquired at different costs, the predictive task, the inductive modeling algorithm, and the labels purchased thus far as also reflected by the current position on the learning curve.¹

Identifying advantageous payments per label does not correspond merely to learning the effect of payment per label on the quality of the labels, so as to infer what payment(s) can produce a given labeling quality. This is because, as has been noted in prior work, acquiring labels of a given quality and cost, can yield different benefits to model learning under different settings (Lin and Weld 2014); consequently, to assess the benefits of acquiring labels at given quality and cost, one must consider the impact of these prospective acquisitions on model performance. To identify advantageous payments at any given time, in this work we propose to estimate the effect of any given

¹ The marginal improvement in predictive performance from acquiring the same set of labeled training instances is likely to be quite different if we already have a large number of labeled training data instances, than if the training set is very small. Similarly, acquiring highly noisy labels may have a different impact on learning when the training set is small, than if added to a large training set.

payment per label directly on the predictive performance of the model. This allows an ALP approach to pursue different payments and labeling qualities under different settings to improve the model's performance, cost-effectively. Specifically, as we will see later on, for some domains and market settings, such a data-driven approach with a focus on model performance allows an ALP approach to either acquire fewer and higher quality labels, or pursue a larger number of cheaper labels, if it estimates such acquisitions will yield comparable (or better) performance for a given cost.

In this paper, we develop an ALP algorithm that aims to select the payment per label that is estimated to yield the most cost-effective improvement in model performance. We then evaluate our approach's performance and robustness relative to alternatives under different settings. Our results demonstrate that the ALP method we propose often yields substantial cost savings compared to the existing benchmark, and that its performance is robust over a wide variety of settings. In addition, we find that when the underlying trade-off between payment and labeling quality changes over time, our ALP method effectively adapts and continues to maintain robust performance. Overall, our results show that the proposed ALP method constitutes both an effective and robust approach for adapting payments to labelers in dynamic crowdsourcing markets.

The contributions of this work are as follows. Our study is the first to introduce the problem of Adaptive Labeling Payment for online crowdsourcing labor markets. We propose a data-driven ALP method that both learns the benefits of alternative payments and continuously adapts the payment per label, with the goal of improving the model's predictive performance cost-effectively. Importantly, our ALP approach aims to directly estimate the expected benefits to predictive performance from labels acquired at different costs. To do this, our algorithm introduces a novel approach for estimating the expected change in model performance from future acquisitions of labels at different payments; thus, our approach uses previously acquired, noisy, labeled data, and it does not rely on the additional acquisition of costly "gold standard" data. The ALP method we develop here is also generic, and can be applied in order to cost-effectively acquire labeled training data for a given task, induction algorithm, population of workers, or set of market conditions. Finally, because our approach aims to identify advantageous payments for label acquisition, it can also be applied in conjunction with methods that address complementary problems, such as repeated labeling methods that aggregate multiple labels per instance to improve the labeling quality, but which do not address what payment per label to offer labelers. In the empirical evaluations, we demonstrate how such methods can be effectively combined. Finally, we conduct an extensive set of experiments to offer insights into our approach's performance and choices of acquisitions under different settings.

The remaining of the paper is organized as follows. We review related research in the prior work section. We then discuss the desired properties of an ALP method, followed by the development of our proposed ALP approach. In the empirical evaluation section, we discuss in detail the setting and procedures we use to evaluate our ALP method's performance compared to alternatives in different settings. We then report our results, followed by conclusions and a discussion of the implications of our work and directions for future research.

2 Prior work

Prior work did not consider how to determine and continuously adapt the payments offered to labelers before they undertook a task, so as to improve the model's predictive performance cost-effectively. Existing work about online labor markets in particular discussed a variety of mechanisms to improve the work quality, such as ways to screen workers and improve task design, and methods for acquiring multiple labels for the same instance so as to increase the probability of obtaining a correct label (e.g., Kazai 2011; Downs et al. 2010; Lee et al. 2013; Paolacci et al. 2010).

One stream of research focuses on repeated acquisition of multiple labels for the same instance, assuming that the payment per label is fixed and pre-determined (e.g., Ipeirotis et al. 2014, Dai et al. 2013, Lin et al. 2012, Zhang et al. 2016). Such repeated acquisitions can be used, for example, to infer the most likely label from multiple noisy ones. In contrast to the study we present here, these works did not address the problem of determining the payment per label to offer workers. Within this stream of research, some studies suggest methods to infer the likely label so as to learn better models using data instances that undergo repeated labeling (Dalvi et al. 2013; Kumar and Lease 2011; Raykar et al. 2010; Rodrigues et al. 2013; Zhang et al. 2015; Zhou et al. 2012). Some repeated labeling methods have also been applied together with active learning methods, in order to reduce the number of instances for which multiple labels are acquired for a pre-determined payment (Ipeirotis et al. 2014; Karger et al. 2011, 2014; Sheng et al. 2008; Wauthier and Jordan 2011; Lin et al. 2016). However, none of these methods identify which payment per label is advantageous to offer labelers and assume that the payment and subsequent quality of each label is predetermined and remains fixed.

Our work differs from this stream of research in several important ways. As noted earlier, the key difference is that prior work did not aim to identify advantageous payment per label to be offered to crowd workers so as to cost-effectively yield a given predictive performance. Some prior work proposed means to assess work quality retrospectively, *after* workers have performed the task (e.g., Raykar et al. 2010; Wang et al. 2017). In this work, we aim to identify advantageous payments offered to workers per label *before* they decide whether or not to take on the task. We also do not assume the availability of individual worker-performance history. Therefore, the method we propose is suitable for recommending payments in popular crowdsourcing labor-market settings such as Amazon Mechanical Turk, where employers continuously encounter new workers, and where tasks and the corresponding payment are offered to all workers who meet certain criteria (e.g., workers are from a certain country). We also consider settings where the work quality that can be obtained for a given payment may not remain the same indefinitely, but rather that market conditions, and consequently the labeling quality that can be obtained for different payment levels, may vary over time. To remain cost effective over time, our approach continuously adapts the payment so as to yield cost-effective improvements in model performance. Finally, in contrast to prior work that aimed to improve label quality (e.g., Wang et al. 2017), we aim to *cost-effectively* improve the *predictive performance* of the model induced from the acquired labels. Indeed, as we discuss in more detail below, the same quality of labels may have different implications for model performance across

modeling tasks, induction algorithms, and across different points along the learning curve. Hence, our approach proposes to assess directly and in a data-driven manner the cost-effectiveness of different label acquisitions toward the model's performance. Consequently, in some settings our approach may recommend buying fewer labels of higher quality and in other settings it may recommend buying more labels for the same or lower quality.

The ALP problem we consider here is also related to active learning (e.g., Lewis and Gale 1994; Abe and Mamitsuka 1998; Saar-Tsechansky and Provost 2004; Kong and Saar-Tsechansky 2014). Active learning methods typically assume that labels for unlabeled instances can be acquired at some fixed, pre-determined cost, and that the acquired labels are also correct; consequently, active learning methods aim to identify the training instances for which to acquire labels, so as to produce the best model performance for any given *number* of acquisitions. Therefore, while ALP aims to determine how much to offer workers per label to produce cost-effective models, active learning addresses a complementary problem, where the payment per label is assumed to be predetermined and fixed, and thus the goal is to identify the instances for which to acquire labels. Importantly for the challenge we address here, most active learning methods also assume that the acquired labels are correct, thus, active learning methods do not typically consider the challenge of inferring the benefit of alternative acquisitions, when the labels acquired are noisy.

Our study is perhaps most closely related to work by Yang and Carbonell (2012), but they make assumptions that do not hold in real online labor markets. Specifically, Yang and Carbonell (2012) do not consider settings which we consider here where: new crowd workers with no prior history are encountered continuously, workers labeling quality for different payments are unknown a priori and can change over time, and where the tasks are offered to all workers on the market (or to all worker who meet a certain criteria); rather, they assume theoretical settings in which labelers' label quality for a given price is predetermined and known to the requester.

In what follows, we discuss the desired properties of an Adaptive Labeling Payment approach in our setting, followed by our proposed ALP approach from which we derive a specific ALP method.

3 The adaptive labeling payment (ALP) approach

In this section, we outline the ALP problem and develop our approach to address it.

3.1 ALP problem formalization

We consider online crowdsourcing labor markets, where a labeling task is offered to all workers for a given payment per label, where no prior knowledge on an individual worker's performance for the particular task may be available, and where the population of workers available at any given time as well as the competitive settings (e.g., the number and properties of other tasks available on the market) may vary over time. Given the market above, a set of unlabeled instances UL for a given classification

task, a *budget* for label acquisition, a model inducer of choice M , and a model performance measure of choice, *Performance* (e.g., AUC), the ALP problem refers to sequential decisions, where at each step, i , a batch of b labeled instances are acquired and added to the set L of labeled training instances. At each step i , one has to decide what payment level $p^i \in C$ from a set of payment options $C = \{c_1, c_2, \dots, c_{po}\}$ to offer workers for labeling a single training instance, before they undertake the labeling task. The goal of selecting payments $p^1, p^2, \dots, p^I$ is that the model $M(L)$, induced from the set of labeled instances L , yields the best model *Performance* for a given budget.² Specifically, our goal is to select payments $p^1, p^2, \dots, p^I$ such that:

$$\arg \max_{\{p^1, p^2, \dots, p^I\}} \text{Performance}(M(L)), \text{ subject to } \sum_{i=1}^I p^i \cdot b \leq \text{budget}.$$

A key property of an ALP problem is that it is data-driven, and an ALP method should effectively learn from the set of instances labeled thus far and continuously adapt to identify advantageous payments in a data-driven manner. In light of prior research findings that different trade-offs between labeling payment and quality arise in different settings, one cannot assume any given trade-off between payment per label and labeling quality for either an arbitrary task or for market conditions. Therefore, a data-driven approach should identify cost-effective payments without making a priori assumptions on the prevailing trade-off. Furthermore, because market conditions, and, consequently, the relationship between pay and quality per label may vary over time, to remain cost effective an ALP approach should also allow to adapt the payment per label to adapt to changing market conditions.

It is useful to note that while a model's generalization performance can be estimated by empirically measuring its performance on a correctly labeled test set, the labels available in our setting are acquired via online crowdsourcing markets, and are thus inherently noisy. An alternative is to acquire "gold standard" data; however, such data are costly and may become obsolete when the underlying concept being learned changes over time. Therefore, in this paper, we aim to develop an ALP method that assesses generalization performance directly from noisy labels acquired via online crowdsourcing markets, without the acquisition of additional data for model evaluation.

Finally, because ALP methods aim to determine advantageous labeling payments that improve model performance cost-effectively, they can be applied alongside methods addressing complementary problems, such as the acquisition of multiple labels for the same instance, as we demonstrate in our empirical evaluations, or with active learning for selecting advantageous instances to label.

3.2 Adaptive labeling payment algorithm

In this section, we develop an ALP algorithm to address the ALP problem and to adaptively select the payment at which labels are acquired. Recall that the ALP problem objective is to identify advantageous payments per label so as to yield the best

² Note that the number of acquisition phases, I , is an outcome of the selected payments and the budget. Thus, if lower payments are selected for labeling, more instances can be acquired over a larger number of acquisition phases.

Table 1 Key notations

Notation	Description
$C = \{c_k\}, k = 1, \dots, p_o$	Set of alternative payment levels per label
n	Number of labeled instances acquired
p^i	Selected payment per label at phase i
T_c	Total cost incurred for labeling payments
S_n	Set of labeled instances so far
b	Number of instances labeled at each acquisition phase
$B_{c_k}^j \subset S_n$	j -th random draw of b instances from S_n , labeled previously at payment C_k
m	Number of subsets $B_{c_k}^j$
h	Number of recent acquisitions used to determine payment
$M(\cdot)$	Model induced from training set $(\cdot)$
budget	Labeling acquisition budget
init_batch	Number of batches of labeled instances required for initialization

model performance for a given budget. Because the benefits to learning from different payments per label are unknown at the outset and because such benefits may also change over time, our approach employs a sequential, myopic heuristic. Specifically, at each acquisition phase, a set of labels is procured on the market, and our approach aims to identify the payment per label to offer labelers that is likely to yield the largest marginal improvement in performance per unit cost.

The approach we propose is iterative, where at each phase $i = 1, 2, \dots, I$, the labels of b instances are acquired on the market. At each phase, the labels for b instances are acquired at a payment level p^i per label, where p^i is selected from a fixed and predetermined set of payment level options $C = \{c_1, \dots, c_{p_o}\}$, such that it is estimated to yield the largest marginal improvement in the model's generalization performance per unit cost. The acquisition of labels proceeds sequentially, until either the budget is exhausted or the model reaches a desired level of performance. Table 1 lists the key notations we use throughout the paper.

3.2.1 Initialization

To compile an initial dataset from which learning proceeds, our ALP algorithm is first initialized by acquiring labels at pay rates that are uniformly drawn from the set of payment level $\{c_1, \dots, c_{p_o}\}$, so as to yield a representative set of instances labeled at different levels of payments. As described below, following the initialization phase, subsequent acquisitions will be made iteratively, such that at each acquisition phase, the next batch of b labels are acquired at the payment level that is estimated to yield the best improvement in performance per unit cost.

3.2.2 Estimating model performance improvement (EPI)

Because our aim is to improve the model's predictive accuracy in a cost-effective manner, our ALP approach aims first to directly estimate the expected impact on *model*

performance of a batch of labels acquired at different possible payments $\{c_1, \dots, c_{po}\}$, per label. Specifically, ALP aims to estimate the expected model performance if the next batch of labels is acquired at each alternative payment level $\{c_1, \dots, c_{po}\}$. The next batch of labels are then acquired at the payment expected to yield the greatest improvement per unit cost.

To estimate the expected effect on model performance of acquiring labeled instances at a given payment per label, our approach *omits* instances labeled previously for the corresponding payment. Specifically, our approach estimates the expected effect on model performance from acquiring the next batch of b labels at a pay $c_k \in \{c_1, \dots, c_{po}\}$, at the current point along the learning curve, by the change in model performance resulting from *removing* from the current training data a set of b instances previously labeled at the corresponding pay c_k . The motivation underlying our omission-based approach is that if labeled instances acquired at pay c_k are more beneficial for induction, their omission from the training data would result in a greater drop in model performance, at the current point along the learning curve.

Formally, let s_n denote the set of all labeled training instances acquired so far (until phase i), let $s_n \setminus B_{c_k}$ denote the set of training instances after removing from s_n a subset B_{c_k} of b instances previously labeled at payment option c_k , and $M(\cdot)$ be a model induced from training set $(\cdot)$ via the induction technique M . ALP approximates the expected change in model performance from acquiring the next batch of b labeled instances at payment option c_k by the Expected Performance Improvement (EPI_{c_k}): the difference between the estimated performance of the current model, $M(s_n)$, and the estimated performance of a model, $M(s_n \setminus B_{c_k})$, induced after omitting the set B_{c_k} of b instances previously labeled at payment option c_k . Formally:

$$EPI_{c_k} = \text{Performance}(M(s_n)) - \text{Performance}(M(s_n \setminus B_{c_k})) \quad (1)$$

In Eq. 1, $\text{Performance}(M(\cdot))$ corresponds to any relevant measure of model performance, such as Area Under the ROC Curve exhibited by model $M(\cdot)$. The more advantageous a batch of labels acquired at payment option c_k is for subsequent learning, the greater the drop in performance between model $M(s_n)$, induced from all labeled data acquired so far, and model $M(s_n \setminus B_{c_k})$, induced after removing the subset B_{c_k} .³

Figure 1 illustrates our proposed approximation of the expected impact on performance of new instances labeled at payment options c_1 and c_2 . The learning curve shows the model's performance, measured here by the Area Under the ROC Curve (AUC), obtained at different acquisition phases. Point A corresponds to the AUC obtained by a model induced from all the data acquired so far, $\text{Performance}(M(s_n))$, and points 1 and 2 reflect $\text{Performance}(M(s_n \setminus B_{c_1}))$ and $\text{Performance}(M(s_n \setminus B_{c_2}))$ —namely, the AUC of the models induced after excluding a batch of labels acquired at payment c_1 and c_2 , respectively. As shown, omitting a set of labels acquired at payment option c_1 results in a more significant drop in performance. Using this approach, ALP assesses

³ Note that this notion holds for all learning curves, including those that are not strictly increasing. For example, in unusual cases when the learning curve decreases, our method can identify the least harmful payment option. In such cases, this notion can also be used to determine when to stop acquiring additional labels.

Fig. 1 Illustration of ALP's approximation of the Expected Impact on performance from acquiring labels at payment c_1 and c_2

the effect on modeling performance of acquiring new labeled instances at different payment options $c_k \in C$.

3.2.3 Selecting payment level

As mentioned above, our ALP method seeks to select the payment that yields the best improvement in model performance per unit cost. Thus, the particular measure we propose to capture the cost-effectiveness of a labeling payment, called “Maximum Total Ratio” (MTR), simply corresponds to the ratio between the (estimated) model's performance after acquiring b labels at payment c_k , and the total labeling cost incurred after the next batch of b labels is acquired. Formally, at each acquisition phase and for each payment option, c_k , MTR_{C_k} corresponds to the cost-effectiveness of acquiring a batch of b instances labeled at payment c_k , and is given by:

$$MTR_{C_k} = \frac{Performance(M(s_n)) + EPI_{c_k}}{Tc + (b \cdot c_k)}, \quad (2)$$

where T_c denotes the total labeling payments incurred thus far. At each phase i , our approach selects the payment option c_k , such that it yields the maximum MTR_{C_k} —namely, the selected payment p^i is given by $p^i = \underset{c_k}{argmax} \{MTR_{C_k}\}$.

Henceforth, we refer to our approach as ALP-MTR.

In what follows, we discuss the remaining elements of our ALP-MTR approach, address how it adapts to changing market conditions, and describe how we improve the estimation of model performance, $Performance(M(s_n))$, in the presence of noisy labels.

3.3 Estimating EPI with noisy data

A key challenge in estimating a model's performance, and, consequently, a challenge in estimating EPI_{c_k} (Eq. 1) as well, is the presence of noisy labels. Recall from our earlier

discussion, that we aim to use the noisy labels acquired via crowdsourcing, without relying on the availability of “gold standard” labels. Relying on labels acquired from crowd workers adversely affects the accuracy of the model performance’s estimation. In particular, errors in the labels contribute to a higher estimation variance, compared to when the estimation relies on correctly labeled training instances. To improve this estimation, our approach incorporates several elements that aim to reduce the variance in the model’s performance estimation.

Recall that our approach approximates the expected change in performance from acquiring b labels at a cost c_k by estimating $\text{Performance}(s_n \setminus B_{c_k})$; namely, by estimating the performance of a model induced after omitting from our training data b instances labeled at payment c_k . To reduce the estimation variance in estimating $\text{Performance}(s_n \setminus B_{c_k})$, we repeat the estimation multiple times by excluding random subsets of b instances acquired previously at payment c_k , and then estimating the model’s performance after this omission. $\text{Performance}(s_n \setminus B_{c_k})$ is then estimated by averaging the models’ performances measured over the multiple repetitions. Specifically, we randomly draw *with replacement* m subsets, $B_{c_k}^j, j \in 1 \dots m$, of b instances previously labeled at payment c_k . At each repetition, j , a different subset $B_{c_k}^j$ is removed from s_n and a measure of performance of the model $M(s_n \setminus B_{c_k}^j)$, induced from the reduced training set $s_n \setminus B_{c_k}^j$, is estimated via cross-validation. For now, we assume that the Area Under the ROC Curve, denoted $\text{AUC}(M(\cdot))$, is the relevant performance measure for a model M induced from training set $(\cdot)$; however, any other performance measure of interest can be used. The expected model’s performance after omitting a batch of b instances labeled at payment level c_k is estimated as the average over the m repeated experiments above:

$$\text{Performance}(M(s_n \setminus B_{c_k})) = \frac{\sum_{j=1}^m \text{AUC}(M(s_n \setminus B_{c_k}^j))}{m}.$$

Finally, to further reduce the error in the estimation of both $\text{AUC}(M(s_n))$ and $\text{AUC}(M(s_n \setminus B_{c_k}^j))$, we perform repeated applications (with different random seeds) of cross-validation, and estimate the model’s performance as the average performance over multiple applications of cross-validation. As noted earlier, cross-validation to evaluate $\text{AUC}(M(s_n))$ and $\text{AUC}(M(s_n \setminus B_{c_k}^j))$ is done using previously labeled training sets $(s_n \text{ and } s_n \setminus B_{c_k}^j, \text{ respectively})$.

3.4 Continuous adaptation of payments

Recall that, changes in the market settings over time, such as those due to changes in the population of workers and competitive settings, may give rise to different trade-offs between payment per label and labeling quality. To adapt labeling payments to the prevailing trade-off, ALP estimates the EPI (Eq. 1) at pay c_k , based on a subset of *recently* labeled instances. More specifically, recall that our ALP approach estimates the EPI from labels acquired at different payment levels, c_k , by estimating the average

change in performance between a model induced from the complete set of instances, $M(s_n)$, and a model, $M(s_n \setminus B_{c_k}^j)$, induced after omitting b instances labeled for a payment c_k . To facilitate adaptation, rather than draw subsets $B_{c_k}^j$ from all prior acquisitions at pay c_k , $B_{c_k}^j$ is drawn from a set D_{c_k} of the most *recent* h instances acquired at pay c_k ($B_{c_k}^j \subset D_{c_k}$), and we subsequently evaluate their average impact on performance. In the empirical evaluations that follow, we evaluate the benefits of this feature in settings where the trade-off between payment and quality per label changes over time, as well as when this trade-off remains the same.

Algorithm 1 outlines the pseudo code for our ALP approach, ALP-MTR. Note that our proposed approach estimates the cost-effectiveness of different payment alternatives by the Maximum Total Ratio measure (Eq. 2). However, other ALP methods may use alternative measures of cost-effectiveness.

Algorithm 1: ALP-MTR

Input:
 Stopping criterion: Exhausting acquisition budget
 Inducer: Classifier inducer of choice, that returns class probability estimates
 b: number of labeled instances acquired at each acquisition phase (batch size)
 h, m, C, init_batch

#Initialization:
 1. $T_c = 0$ # total cost so far
 2. Initial labeled set of instances: $S_0 = \{\}$
 3. Repeat init_batch times
 4. Randomly draw a payment level $c_k \in C$
 5. Purchase a batch of b labeled instances at payment c_k and add them to S_n
 6. $T_c \leftarrow T_c + b \cdot c_k$
 7. End repeat

Select payments
 8. While Stopping criterion is not met (Total_pay $\leq$ budget)
 9. Performance($M(s_n)$) = Compute AUC($M(s_n)$)
 10. **For** $k = 1, \dots, p_o$ # over all payment alternatives
 11. $D_{c_k} \leftarrow$ from S_n select h instances labeled at payment c_k most recently
 12. **For** $j = 1, \dots, m$
 13. $B_{c_k}^j \leftarrow$ randomly draw b instances from D_{c_k}
 14. Compute AUC($M(s_n \setminus B_{c_k}^j)$)
 15. **End For**
 16. Performance($M(s_n \setminus B_{c_k})$) = $\frac{\sum_{j=1}^m \text{AUC}(M(s_n \setminus B_{c_k}^j))}{m}$ # Calculate average performance
 17. $\text{EPI}_{c_k} = \text{Performance}(M(s_n)) - \text{Performance}(M(s_n \setminus B_{c_k}^j))$
 18. **End For**
 19. If ($\text{Max}\{\text{EPI}_{c_k}\} \geq 0$) : $p^i = \text{argmax}_{c_k} \left(\frac{\text{Performance}(M(s_n)) + \text{EPI}_{c_k}}{T_c + (b \cdot c_k)} \right)$ # p^i is the "best" payment option
 20. else : $p^i = \text{argmax}_{c_k} \{\text{EPI}_{c_k}\}$
 21. Purchase b labeled instances at pay p^i and add them to S_n
 22. $T_c \leftarrow T_c + b \cdot p^i$
 23. End while

4 Experimental setup

We evaluate our method's performance, compared to alternatives under a variety of simulated settings, corresponding to different labeling tasks and trade-offs between payment and quality that have been reported in prior work on crowdsourcing labor markets. In addition, because different trade-offs may arise at different times, we also evaluate our approach's performance under conditions in which the prevailing trade-off between payment and quality changes over time. Across multiple settings, we aim to identify whether a given method yields a robust performance. Specifically, a robust ALP method that can be relied on in practice ought to yield consistent performance across settings, and should not fail miserably in some settings.

Towards the evaluations we simulate different trade-offs between payment and quality identified in prior works. These include “Asymptotic”, “Concave”, and “Fixed” trade-offs illustrated in Fig. 2a–c. In Fig. 2 and throughout the paper, quality is characterized by the likelihood of error. Specifically, Kazai (2011) and later Kazai et al. (2013) have found an “asymptotic” trade-off between payment and quality, illustrated in Fig. 2a. A “concave” trade-off, illustrated in Fig. 2b, where quality improves initially with increasing payments, but then degrades beyond a certain payment level, was reported in several prior works, including by Kazai (2011) and Kazai et al. (2013), as well as by Feng et al. (2009). Kazai et al. (2013) conducted analyses to understand the possible causes for this trade-off; they have identified that while individual workers tend to produce better quality with a higher payment, higher payment levels also draw increasingly unethical and opportunistic workers, which can reduce the overall work quality. In fact, the two opposing effects may also explain the asymptotic trade-off function observed by Kazai (2011), and by Kazai et al. (2013). The third trade-off we consider here is a “Fixed” trade-off, documented by Mason and Watts (2010) and illustrated in Fig. 2c, where different payments yield the same quality.

Because different trade-offs arise in different market settings that can vary over time, it is also useful to examine the robustness of an ALP approach when the underlying trade-off between payment and quality shifts from one trade-off to another. In the evaluations reported below, we report ALP-MTR's performance when a given trade-off switches to another after 50% of the acquisition budget has been used.

Fig. 2 Different trade-offs between payment and quality observed in real experiments

Table 2 Datasets used for evaluation

Dataset	Task	Number of observations	Number of variables	Percent positive (%)
SPAM	Classifying email as spam or non-spam	4601	57	39.4
Pen digits	Classify whether an image displays a digit	10,991	16	9.6
Mushroom	Classify whether a mushroom is edible	8124	22	48.2

In the simulation studies we report below, once the payment to be offered to labelers for the next batch of labels is selected, the probability q that a label acquired at that payment is correct is determined by the prevailing trade-off. Thus, the correct label for an instance is assigned at probability q , and the label is reversed at probability $(1 - q)$. We also followed prior work for the payment range and the number of payment alternatives from which payment methods can select. Specifically, prior studies considered 2–4 payment levels, ranging from a minimum payment of \$0.01–\$0.03 to a maximum payment of \$0.1–\$0.25 (e.g., Feng et al. 2009; Kazai 2011; Kazai et al. 2013; Mason and Watts 2010; Rogstadius et al. 2011); thus, in the evaluations that follow, we considered three payment alternatives within this range of \$0.02, \$0.14, and \$0.25, reflecting low, mid, and high payment levels.

For each of the trade-offs outlined above, as well as for settings where the trade-off changes over time, we report experimental results for labeling tasks corresponding to three publicly available datasets. The first two, Pen Digits and SPAM (Lichman 2013), were selected as they reflect typical labeling tasks that are easy for humans to produce. The third dataset, Mushroom (Lichman 2013), is used for additional robustness and had been used in prior simulation-based research on labeling via crowdsourcing markets (Ipeirotis et al. 2014). Table 2 provides details regarding the different datasets. In “Appendix F” we also provides empirical results that show the number of instances required to reach various levels of model performance (measured by AUC) as a function of different levels of labeling quality.

4.1 Evaluation procedure and measures

Our evaluation procedure is illustrated in Fig. 3 and includes three modules: (a) Payment Method, (b) A module which simulates the Prevailing Trade-off in the market between payment and quality at any given time, and (c) Reporting and Measurement module.

The evaluation begins by randomly partitioning a labeled dataset into an Unlabeled dataset (70% of instances), for which the true labels are hidden and later can be acquired by different payment methods, and a Holdout dataset (the remaining 30%) containing instances with the correct labels; the correctly labeled holdout set is used exclusively for the evaluation of the models produced from labeled instances acquired by the alternative approaches (and thus not used by any approach to inform their payment

Fig. 3 Overview of the Evaluation Platform

selection). For control, the same holdout set is used to evaluate ALP-MTR and baseline methods. At each iteration of batch label acquisition, the payment for purchasing the labels of a batch of 10 unlabeled data instances from the Unlabeled set is selected by a payment method (e.g., ALP-MTR or baseline). Once the payment is determined, the Simulated Trade-off module uses the prevailing trade-off between payment and quality per label (asymptotic, fixed, or concave) to determine the labeling quality q . As described above, this probability is then used to produce the label, such that the correct label for each instance is assigned at probability q . Once the labels are determined, the labeled instances are added to the corresponding method's training set. Also, for control, at each iteration (acquisition phase), the labels for the same instances, drawn at random from the unlabeled set, are acquired by ALP-MTR and the baseline. Hence, any difference in cost-effectiveness can be attributed exclusively to the payment selected by each approach and the corresponding labeling quality produced as a result.

For each approach, once a new set of labeled instances is acquired and added to the corresponding approach's training set, a new model is induced from the augmented training set, and we report its performance over the correctly labeled holdout data set, so as to assess the improvement in model performance achieved by each approach from all of its label acquisitions.

We report the average performance, over 20 repetitions, of the learned model's Area Under the ROC Curves (AUC) as a function of cumulative labeling costs incurred by each approach to achieve this performance. AUC is evaluated over the correctly labeled holdout set.

Because the goal of developing ALP methods is to reduce the labeling cost, we report the labeling Cost Saving (CS) enabled by ALP-MTR to yield the highest AUC obtained by the baseline alternative. CS is illustrated using the stylized plot shown in Fig. 4: The benchmark method (B) incurs costs of \$130 to obtain its highest AUC

Fig. 4 Illustration of label acquisition costs incurred by two methods, A (red curve) and B (black curve), to achieve difference model performances. The horizontal line reflects the Cost Saving measure: the difference between the labeling costs incurred by B to yield its best performance, and the costs incurred by A to yield the same performance (Color figure online)

of 0.945, while the ALP method (A) achieves the same level of performance after incurring \$63, yielding cost savings of 52%.

We also report the statistical significance of the difference between the areas under the AUC curve as a function of the labeling costs generated by two competing methods. To this end, we use the BCA bootstrap method implemented in R, with 10,000 repetitions.

In the experiments reported here, we induced predictive models using the R implementation of the popular Random Forest algorithm (R Package: RandomForest). We used the default parameter setting with 100 base trees. We later replicate our main results for SVM and Bagging models. In the experiments reported below, ALP-MTR used eight-fold cross-validation repeated four times toward this estimation.⁴ For the history parameter h , we used 100 recently labeled instances to evaluate alternative payments. We later explore the robustness of ALP-MTR's performance when varying the values of these parameters.

4.2 Baseline payment policy

To the best of our knowledge, there are no existing solutions to the ALP problem. Hence, to identify a robust baseline policy we evaluated three possible alternative

⁴ These cross-validation parameters were selected for efficiency using 32 core machines.

baseline payment policies, two of which are optimal under some settings. The first policy, henceforth referred to as Minpay, always offers the lowest payment for labels. A second policy, Maxpay, always offers the highest payment. A third policy, henceforth referred to as UNIFORM, acquires labels at a representative set of payments—thus, in each batch of payments, Uniform draws uniformly at random the payment at which labels are acquired. In “[Appendix A](#)” we examine the robustness of the three alternative policies.

We find that both Minpay and Maxpay lack robustness and they each exhibit very poor performance under some settings. By contrast, the UNIFORM policy yields robust performance and can be relied on in practice not to fail miserably. Henceforth, we evaluate ALP-MTR’s performance relative to that of the robust Uniform policy.

5 Results

Table 3 summarizes the average cost savings achieved by ALP-MTR to yield the best performance exhibited by the UNIFORM policy for different data sets and trade-offs. As shown, ALP-MTR yields a significant reduction in the costs necessary to yield the best model performance achieved by the UNIFORM payment policy. Specifically, Table 3 shows that across different settings and labeling tasks, ALP-MTR yields a substantial improvement of 35.6% savings on average in labeling costs.

Figure 5a–i shows the performances obtained for different labeling acquisition costs incurred by ALP-MTR and the UNIFORM policy, under different labeling tasks and trade-offs between payment and quality. As shown, the ALP-MTR method exhibits consistent performance and is superior to the UNIFORM policy. For the learning curves presented in Fig. 5, Table 4 shows significance test results for the difference between the areas under the AUC curves produced by ALP-MTR and UNIFORM and shown in Fig. 5. For all

Table 3 Cost savings generated by ALP-MTR to achieve uniform’s best performance

Dataset	Trade-off function	Cost savings produced by ALP-MTR (%)
Mushroom	Asymptotic	39.9
	Concave	50.6
	Fixed	18.4
Spam	Asymptotic	39.7
	Concave	60.7
	Fixed	15.6
Pen Digits	Asymptotic	34.3
	Concave	38.2
	Fixed	23.0
Average savings		35.6

Fig. 5 Comparison of ALP-MTR and the UNIFORM payment policy. ALP-MTR is often significantly better and is otherwise comparable to that of the UNIFORM policy

datasets and underlying trade-offs, the difference between ALP-MTR and the UNIFORM policy is statistically significant ($p < 0.01$).⁵

Overall, as shown in Tables 3, 4, and Fig. 5, across settings, both the cost saving and the difference in performance between ALP-MTR and UNIFORM is statistically significant. Note that the absolute magnitude of the possible improvement in model performance is also a characteristic of the data domain, the underlying tradeoffs in the market between labeling payment and quality, and the induction algorithm—and not merely of the approach used to acquire labels. Specifically, for different

⁵ Figure 5 is adjusted to improve visibility of the relative performance of each approach. In addition, note that we aim to establish whether our method outperforms the baseline over multiple settings, rather than support a single hypothesis of improvement over the baseline. Hence, the significance tests across all settings should be examined (i.e., in what proportion of the settings the improvement over the baseline was significant). Consequently, individual tests do not include correction for False Discovery Rate.

Table 4 Significance testing for differences in area under the AUC curves

Dataset	Trade-off function	Figure	Significance of difference between ALP-MTR and UNIFORM (***) $p < 0.01$
Mushroom	Concave	6a	***
	Asymptotic	6b	***
	Fixed	6c	***
Spam	Concave	6d	***
	Asymptotic	6e	***
	Fixed	6f	***
Pen digits	Concave	6g	***
	Asymptotic	6h	***
	Fixed	6i	***

Bootstrap p values for significance test on the difference between AUC produced by ALP-MTR and UNIFORM, shown in Fig. 5. Bootstrap tests were performed using the bca method implemented in R. *** $p < 0.01$

data domains, tradeoff in the market, and induction algorithms, different improvements are feasible through different labeling payment selection strategies.

5.1 Changing trade-off between payment and quality per label

A key motivation for an Adaptive Labeling Payment policy is that, for any arbitrary labeling task and market settings, the trade-off between payment per label and labeling quality is unknown, and may also change over time (e.g., possibly due to changing market conditions). In what follows, we evaluate the proposed ALP-MTR's robustness in settings where the underlying trade-off between pay and label quality varies. These evaluations aim to establish whether under these conditions the ALP method continues to be advantageous and to yield robust performance. In the experiments reported here, the prevailing trade-off in the market changes once ALP incurs \$75 in labeling costs.

Table 5 summarizes the average cost savings achieved by ALP-MTR relative to the UNIFORM policy when the underlying trade-off changes from one trade-off to another. As shown, ALP-MTR exhibits robust behavior and is superior to the UNIFORM policy, yielding a significant savings of 32.3% in labeling costs, on average.

Figure 6 shows curves of the predictive performance obtained for different labeling acquisition costs. In the interest of space, we present results for a subset of the settings, and include in “Appendix B” results for all remaining settings. As shown, when the trade-off varies, ALP-MTR often achieves superior performance and is otherwise comparable to the UNIFORM approach across settings. Figure 6f shows an interesting result in which ALP-MTR's performance briefly deteriorates soon after the underlying trade-off changes; ALP-MTR then detects and adapts to the change, yielding cost-effective improvements in the model's performance. For all the results reported in Figs. 6 and 12 (“Appendix B”), Table 6 shows significance test results for the difference between the area under the AUC curves produced by ALP-MTR and UNIFORM. In almost all set-

tings, the difference is statistically significant. Importantly, in the single setting where ALP-MTR did not yield statistically significant superior performance, ALP-MTR yields performance comparable to that produced by the UNIFORM policy.

Table 5 Cost Savings achieved by ALP-MTR to yield UNIFORM's best performance

Dataset	Change in trade-off (From $\rightarrow$ to)	Cost savings produced by ALP-MTR (%)
Mushroom	Concave $\rightarrow$ asymptotic	38.5
	Concave $\rightarrow$ fixed	24.3
	Asymptotic $\rightarrow$ concave	46.3
	Asymptotic $\rightarrow$ fixed	29.1
	Fixed $\rightarrow$ asymptotic	26.0
	Fixed $\rightarrow$ concave	29.9
Spam	Concave $\rightarrow$ asymptotic	58.9
	Concave $\rightarrow$ fixed	48.1
	Asymptotic $\rightarrow$ concave	57.3
	Asymptotic $\rightarrow$ fixed	25.8
	Fixed $\rightarrow$ asymptotic	34.6
	Fixed $\rightarrow$ concave	10.4
Pen digits	Concave $\rightarrow$ asymptotic	29.4
	Concave $\rightarrow$ fixed	9.2
	Asymptotic $\rightarrow$ concave	38.0
	Asymptotic $\rightarrow$ fixed	6.2
	Fixed $\rightarrow$ asymptotic	31.0
	Fixed $\rightarrow$ concave	38.8
Average savings		32.3

Fig. 6 Performance of the ALP-MTR and the UNIFORM payment policy when the underlying trade-off between labeling payment and quality changes. As shown, ALP-MTR's performance is often significantly better than that of UNIFORM, and ALP-MTR is never worse than UNIFORM

Fig. 6 continued

Table 6 Significance testing for difference in area under the curves

Dataset	Change in trade-off functions (From → to)	Figure	Difference between ALP-MTR and uniform
Mushroom	Concave → asymptotic	7a	***
	Concave → fixed	12a	***
	Asymptotic → concave	12d	***
	Asymptotic → fixed	12e	***
	Fixed → asymptotic	7b	***
	Fixed → concave	12c	***
Spam	Concave → asymptotic	7c	***
	Concave → fixed	12g	***
	Asymptotic → concave	12j	***
	Asymptotic → fixed	12k	**
	Fixed → asymptotic	7d	
	Fixed → concave	12i	***
Pen Digits	Concave → asymptotic	7e	***
	Concave → fixed	12m	**
	Asymptotic → concave	12p	***
	Asymptotic → fixed	12q	**
	Fixed → asymptotic	7f	***
	Fixed → concave	12o	***

Bootstrap p -values for significance testing of the difference between the area under the AUC curves produced by ALP-MTR and UNIFORM, shown in Figs. 6 and 12 (in “Appendix B”). Bootstrap tests were performed using bca method implemented in R. * $p < 0.1$, ** $p < 0.05$, *** $p < 0.01$

5.2 Studies of ALP-MTR's performance, its elements, and parameter settings

In this section we evaluate the contribution of different elements of ALP-MTR to performance. In addition, we discuss and provide empirical evaluations to assess the effect of different ALP-MTR parameters on performance.

5.2.1 Generalization performance in the presence of noisy labels

A key element of ALP-MTR is the assessment of a model's generalization performance in the presence of noisy labels. To that end, ALP-MTR performs repeated omissions of sets of instances that were previously labeled at a given payment level, drawn at random with replacement, and the effect of these repeated omissions on predictive performance is averaged.

We empirically evaluate the benefits of using repeated omission to ALP-MTR's selection of cost-effective payments. Figure 7 shows the performance of ALP-MTR when $m = 10$ repeated omissions are done to estimate the cost-effectiveness of each payment alternative (default settings), and the performance of an ALP-MTR variant, ALP-MTR-SINGLE-OMISSION, in which a single subset of instances is omitted toward this estimation. In the interest of space, Fig. 7 shows results for the Pen Digits dataset, which are representative of the comparison across datasets. As shown, ALP-MTR's repeated sampling of subsets for omission benefits ALP-MTR's selection of cost-effective payments, thereby often producing better and otherwise comparable performance, compared to the effect of omitting a single set of instances.

Another element of ALP-MTR that aims to reduce variance in the estimation of the cost-effectiveness of different labeling payments is the repeated applications of a cross-validation for estimating the expected change in performance after omitting instances labeled at alternative payment levels. Figure 8 shows a comparison of the standard ALP-MTR with four repetitions of cross-validation (default setting) and ALP-MTR-SINGLE-CV, where only a single cross-validation is used in the estimation. In the interest of space, we present representative results for the Pen Digits dataset and these findings are consistent across datasets. As shown, repeated cross-validations benefits

Fig. 7 Comparison between ALP-MTR and ALP-MTR-SINGLE-OMISSION. ALP-MTR achieves superior or otherwise comparable performance to that of ALP-MTR-SINGLE-OMISSION

Fig. 8 Comparison between ALP-MTR and ALP-MTR-SINGLE-CV. ALP-MTR yields better or comparable performance to that of ALP-MTR-SINGLE-CV

the selection of cost-effective labeling payments, thereby often producing better and otherwise comparable performance as compared to when this procedure is not applied. Therefore, it is recommended to apply this feature and take the average measurement of the repeated cross-validation procedure. Yet, note that repeated cross-validation bears increased run-time or can be parallelized.

5.2.2 The history parameter settings

Recall that ALP-MTR evaluates alternative labeling payments by estimating the effect of omitting only labels acquired during the most recent h batches. The history parameter h aims to enable ALP-MTR to adapt to recent changes in the underlying trade-off between labeling payment and quality and ignore older patterns. A small value of h will result in ALP-MTR taking into account only recent market patterns, whereas, a large value for h takes will result in taking into account market patterns that may have occurred a long time ago.

Figure 9 shows a comparison of ALP-MTR (using the default value of $h = \text{last } 10 \text{ batches}$, per payment alternative) with an ALP-MTR variant, ALP-MTR-FULL-HISTORY, which evaluates the cost-effectiveness of alternative payments based on the entire purchase history. Figure 9a–c show results for settings in which the trade-off between labeling payment and quality remains constant, and Fig. 9d–i show learning curves for settings in which the trade-off changes. In Fig. 9, we show results for the Pen Digits dataset; results with the other datasets produced the same finding. As shown, across different trade-off settings, ALP-MTR (with a default value of $h = \text{last } 10 \text{ batches}$) yields either superior or comparable cost-effective acquisitions as compared to ALP-MTR-FULL-HISTORY, suggesting that consideration only of recent purchases when assessing alternative payments is beneficial.

Fig. 9 Comparison between ALP-MTR and ALP-MTR-FULL-HISTORY. ALP-MTR yields better or equivalent performance compared to ALP-MTR-FULL-HISTORY

5.2.3 Batch size

The batch size parameter, b , determines how many labels are acquired simultaneously at each acquisition phase. In “[Appendix E](#)” we present an evaluation of ALP-MTR’s robustness to different batch sizes (5, the default setting of 10, 15, and 20). We find that in most cases altering the batch size does not significantly affect performance, however, in a few cases a, larger batch size slightly improves performance as compared to a batch size of $b = 5$.

5.2.4 Initial random purchases

During the initialization phase, ALP-MTR compiles an initial dataset from which learning proceeds, and the number of batch acquisitions performed during initialization is controlled by the *init_batch* parameter. Setting a larger *init_batch* value can have

a contrasting effect on performance. On the one hand, a larger *init_batch* value can help produce a more representative initial sample, but because these acquisitions are uninformed, this will also delay the production of informed, cost-effective, payment decisions. In “[Appendix G](#)”, we present experimental results exploring the effect on performance when the number of initial batch purchases is either increased or decreased by 30%. We find that performance is not significantly affected by this change in the *init_batch* parameter.

5.2.5 Number of payment alternatives

ALP-MTR considers a set of payment levels $\{c_1, \dots, c_{p_0}\}$ and select advantageous payments in each acquisition phase. In the experiments presented here we considered three payment levels produced by evenly partitioning the range payments per label into three payment alternatives. However, such partitioning can indeed be made more refined to explore a larger number of alternatives. In “[Appendix H](#)” we evaluated the effect of refining this partition and considering a larger number of payment options (while maintaining the same number of instances in the initialization phase). We find that ALP-MTR achieves better or comparable performance across settings when considering a larger number of alternative payment levels.

5.2.6 Induction algorithms

In “[Appendix C](#)”, we replicate our main results using two alternative classification algorithms, Support Vector Machine and Bagging, to induce predictive models from the labels acquired by ALP-MTR. In both cases, ALP-MTR yields significant cost reduction as compared to the baseline UNIFORM policy. These results suggest that our method is generic and offers benefits for inducing different classification models.

5.3 Additional insights and analyses

In this section we discuss additional analyses of the ALP-MTR approach, including insights into its choices of acquisitions, its runtime complexity, and evaluation of a new variant. We also demonstrate how our approach can be used to determine payment per label in other settings, when repeated labeling is used.

5.3.1 Payment regulation

Because labels are noisy in our settings, having more acquisitions for a given payment can improve ALP-MTR’s estimation of the contribution to model induction of labels acquired for the corresponding payment. However, it is possible that due to noise, cost-effective payments should be deemed undesirable. Further, because such a payment is not selected, the estimation is not improved and the advantageous payment will continue to be ignored. This problem may be particularly significant when the trade-off between labeling payment and quality varies over time. In “[Appendix D](#)”, we consider an ALP-MTR variant, ALP-MTR-PAYMENT-REG, which uses ALP-MTR’s selection of payments as before; however, it also acquires labels at a given payment level if it has not been selected in the recent t consecutive batch acquisitions. Our results suggest that

initiating acquisitions at different payment levels, as done by ALP-MTR-PAYMENT-REG, can improve performance in some cases, but often yields inferior performance to that achieved by the standard ALP-MTR.

5.3.2 Insights into ALP-MTR's payment strategies

The motivation underlying the ALP problem and which we outline in the introduction, is that the choice of payment per label that will yield a cost-effective improvement in model performance is affected by a host of factors, including the prevailing trade-off in the market between payment and quality per label, the underlying predictive task (data domain), the inductive algorithm used, and the labels available thus far (the corresponding “position” along the learning curve). Consequently, our ALP-MTR method aims to estimate directly, and in a data-driven manner, the effect of any given payment per label at any given time on the model's predictive performance.

Fig. 10 Model performance, average label quality, and number of labels acquired

As such, we note in the introduction, how an effective ALP approach ought to be able to acquire labels at a lower cost if this is estimated to yield more cost-effective improvements in performance, while in other settings, the same approach may produce a given model performance at a lower cost by acquiring fewer, but higher quality labels. We now show results that demonstrate ALP-MTR's flexibility to pursue different labeling qualities and costs under different settings to yield cost-effective improvements in model performance.

Figure 10 shows how ALP-MTR indeed achieves cost-effective model performance via versatile acquisition choices under different data domains and trade-offs between labeling payment and quality. For different trade-offs between payment and quality per label in the market, Fig. 10 shows the model performance, average label quality and number of labels acquired as a function of acquisition cost, resulting from acquisitions made by ALP-MTR and the UNIFORM policies. Specifically, for the Spam data set, under an asymptotic trade-off between labeling payment and quality, Fig. 10a–c show that ALP-MTR initiates the acquisition of fewer and higher quality labels on average. For the same data set, Fig. 10d–f shows that when the underlying trade-off between labeling quality and cost is fixed (the same label quality is produced for any given payment per label), and improvements in cost-effectiveness can only be achieved by acquiring less costly labels, ALP-MTR purchases a larger number of cheaper labels. Finally, for a linear positive relationship between payment per label and labeling quality,⁶ Fig. 10g–i shows that ALP-MTR does not pursue the highest quality labels, but a larger number of lower quality labels. Overall, ALP-MTR is able to produce versatile label acquisition choices across different settings, which is fundamental to its ability to produce cost-effective improvements in model performance, under arbitrary settings.

5.3.3 Runtime complexity

ALP-MTR runtime complexity at each acquisition phase depends on the parameter choices reported above as well as the time required to train and evaluate a model when utilizing an induction algorithm of choice (e.g., Random Forest, or SVM). Specifically, given:

T_m —Time to train and evaluate a single model with the induction algorithm of choice.

po —Number of payment options

m —Number of subsets (batches) that are repeatedly omitted

$folds$ —Number of cross-validation folds

R_{cv} —the number of times the entire cross-validation procedure is repeated

At each acquisition phase, the runtime complexity is given by:

$$T_m * folds * (1 + po * m * R_{cv})$$

Yet, because all model inductions and evaluations can be done simultaneously, prior to each batch acquisition by ALP-MTR, these computations can be fully parallelized. This yields an effective run time of $\sim O(T_m)$. We timed running ALP-MTR

⁶ The linear relationship is given by $Quality = 0.8265 + 0.1739 \cdot Cost$, where $0.83 \leq Quality \leq 0.87$, and $0.02 \leq Cost \leq 0.25$.

using default settings and unoptimized code, on an Intel E5-2630 2.2 GHZ, 10 core machine.⁷ Average computation time at each acquisition phase was 15.86, 16.25, and 8.63 seconds for the Spam, Mushroom and Pendigits datasets, respectively.

5.3.4 Using ALP-MTR to select payment per label for repeated labeling

Repeated labeling refers to the acquisition of multiple noisy labels for the same instance so as to infer the most likely label and improve the accuracy of the labels. Importantly, methods for repeated labeling do not determine which payments to offer labelers per label, but assume that the payment per label is fixed and somehow pre-determined. Repeated labeling thus aims to reduce noise (error) in the data by acquiring multiple labels for the same instance at a fixed and pre-determined cost per label, and use the set of labels available for a given instance to infer the most likely label. For example, a popular approach is to infer the majority label for each instance (e.g., Lee et al. 2013; Mason and Suri 2012). Since ALP-MTR and repeated labeling address complementary tasks, and can thus be applied in conjunction—in “[Appendix J](#)” we demonstrate how ALP-MTR can be applied to identify advantageous payments per label when repeated labeling is used. In particular, we apply ALP-MTR first to identify the cost-effective payment per label at which to acquire the next batch of labels; multiple labels for the same instance are then purchased at the payment determined by ALP-MTR, and these labels are aggregated by repeated labeling. We find that applying ALP-MTR for advantageous payment selection and subsequently applying repeated labeling, yields significant savings.

5.3.5 Potential for future improvements

ALP-MTR labeling payment choices yield robust performance, but may not be optimal. For a given setting, in “[Appendix I](#)” we offer insights into the potential for future improvements to our approach via a comparison between the payment choices ALP-MTR makes and those of an oracle that knows the optimal payment.

6 Conclusions, limitations, and future work

As machine learning becomes integral to the routine operations of firms and the products and services they provide, the immediacy and accessibility of online labor markets present unprecedented opportunities for on-demand labeling to be brought to bear on machine-learning tasks. Yet for these opportunities to materialize, it is important to devise solutions for the novel challenges these markets present. Given the different trade-offs that can arise between labeling payment and work quality under different labeling tasks and market conditions, in this paper we first formulate the problem of Adaptive Labeling Payments (ALP), then develop an approach to address it, and study extensively the performance of our proposed approach. Specifically, we develop an ALP approach, ALP-MTR, designed to determine and continuously adapt the payment

⁷ Default settings include $p_0 = 3$, $m = 10$, $\text{folds} = 8$, $R_{cv} = 4$ indicating the potential for substantial run time improvement given the availability of additional computational resources compared to our 10 core machine.

offered to crowd workers before they undertake a labeling task, so as to improve the performance of a model induced from the data cost-effectively. Our ALP approach estimates the effect on induction from omitting training instances, previously acquired at different payments; our approach also incorporates elements that benefit this estimation, particularly in the presence of noisy labels and changing prevailing trade-offs between payment and quality.

We empirically evaluated the performance of ALP-MTR relative to that of a robust alternative under a variety of market scenarios, reflecting different labeling tasks and trade-offs between labeling payments and quality documented in prior work, including settings where the trade-off changes over time. Our results show that ALP-MTR yields robust performance across settings and that it offers meaningful and substantial cost savings, with an average savings of 33.4% across settings. We also demonstrate that the design elements of ALP-MTR, which aim to improve its estimations of the expected benefits from alternative payments in the presence of noisy labels, and its adaptation to changing market conditions, indeed benefit ALP-MTR's performance. Given our method's consistently robust benefits under different settings, it can be considered a benchmark for evaluating future mechanisms to determine labeling payments.

The practical implications of this research are important for enabling a growing reliance on instance labeling via crowdsourcing labor markets. Our ALP approach yields both robust performance and meaningful cost savings. Because crowdsourcing for labeling tasks is becoming increasingly popular in academic research as well, academic efforts can similarly benefit from our method's efficiencies.

Our proposed approach offers a wealth of opportunities on which future research can build, both to improve as well as to extend our work. One interesting direction would be an adaptation of our ALP approach to regression predictive tasks, in addition to the classification task that we consider here. As prescribed by our ALP approach, the effect of target (dependent variable) values acquired at different payments may, in principle, be evaluated by estimating the effect on the regression model's performance after omitting instances previously acquired at the corresponding payment, as the estimations can benefit from the same elements included in ALP-MTR to improve the estimation under noisy labels and changing market conditions. Similarly, while in this work we considered an important and popular measure of performance (namely, Area Under the ROC Curve) to evaluate a classification model's performance, the framework we develop could accommodate other performance measures as well.

Our approach is designed to adapt to changing market conditions. However, inherent to all predictive modeling approaches is the assumption that the environment remains stable for some time so that the learned patterns can be exploited. Different solutions may be required in chaotic settings, where market conditions change significantly and very frequently.

Any data-driven-learning-based approach incurs costs for *learning*. In our context, a practical reference to assess these costs corresponds to the non-data-driven and robust, UNIFORM, alternative. A useful question to consider is whether our approach remains beneficial even if only a small number (i.e., a few hundred) of labels are acquired on the market. Indeed, ALP-MTR includes an initialization phase, during which it acquires an initial set of labeled instances, and from which it begins learning; importantly, these initial label acquisitions are uninformed by an estimation of the expected

benefits and may thus be rather costly. However, during the initialization phase, ALP-MTR selects payments in the same manner by which payments are selected by the robust UNIFORM alternative. This is also demonstrated in our empirical results, where ALP-MTR does not incur higher costs relative to a non-data-driven alternative early on. Nevertheless, it would be beneficial for future work to further improve ALP-MTR's performance as well as reduce the cost of (shorten) the initialization phase.

ALP-MTR was designed for and evaluated in challenging settings where all the labels are potentially noisy and no gold standard labels are available. However, with trivial modifications, ALP-MTR can be adjusted to operate in more favorable conditions when correct labels ("gold standard") are available for some instances. To improve performance, such gold standard data can be used towards ALP-MTR's evaluation of EPI_{c_k} (estimation of expected performance improvement for a certain payment level) as an internal holdout set, instead of applying cross-validation over noisy instances. Future work can also suggest how to best use gold standard data, including how to partition such data for training and ALP-MTR's internal holdout sets.

Recall that ALP-MTR adapts the payment per label only in consecutive acquisition phases, and does not simultaneously offer different payments for the same task. Nevertheless, it is possible that adaptive labeling methods may introduce "second-order effects"—i.e., that the mere *change* in payment, rather than the payment level, offered at different times might impact worker behavior. Prior work (Chen and Horton 2016) has shown that in different settings than those we consider here, and where the same worker who works continuously on a task is faced with wage cuts, the worker is more likely to discontinue taking additional tasks. (Chen and Horton did not find evidence that this impacts the work quality, unless workers are solicited to continue working in an inadequate manner). Importantly, our approach's performance may not be effected by an individual worker's decision to discontinue work due to changes in pay. This is because payments are revised and offered to the entire market in consecutive phases, and payments are not revised for a given worker. Furthermore, because the market is often large (as in real markets such as Amazon Mechanical Turk), the potential effect of such adaptation may not be substantial. Nevertheless, it is possible that this or similar effects may arise in different market conditions. It would be therefore beneficial for future work to model and measure any effects from revising the payment offered on the market over consecutive phases. Lastly, because our approach corresponds to a sequential process, where labeling tasks are offered sequentially on the market, to alleviate such effects it may be beneficial to consider revising the framing of the labeling task in subsequent phases.

Our approach does not consider the properties of individual instances, such as instance level properties that might yield differential difficulty for labelers to label them correctly. If a given instance is less likely than other instances to be labeled correctly, it may be beneficial for future work to develop methods that identify advantageous payments for labeling individual instances, so as to cost-effectively improve model performance.

Another possible venue for extending the ALP problem is to adapt it to domains in which it is possible to compartmentalize the feature sets into different subsets, each of which can be used to produce different models, and for which a different trade-off between labeling payment and quality might arise. For example, in text classification

tasks, some of the workers may be presented just with the text headline and receive a lower payment, whereas, other workers may be requested to classify an entire news article and receive a higher level of payment.

Finally, it would be interesting for future research to explore different ways of extending our approach. In particular, it would be beneficial for future work to explore the possibility of removing instances labeled at certain payments if this could improve the model's generalization performance. Similarly, it would be interesting to explore an extension of our approach that studies possible effective stopping criteria for acquisitions at a given time. Such stopping criteria could be based, for example, on the EPI measure outlined in Eq. 1 to assess whether acquiring labels at a given payment might undermine induction.

Acknowledgements The authors are grateful for insightful comments and suggestions by the Associate Editor and three reviewers. The paper has also greatly benefited from comments and discussions with seminar participants at Cornell University's Operation Technology and Information Management (OTIM) Workshop, Boston University, The University of Iowa, Temple University, as well as participants of the Statistical Challenges in E-Commerce Research (SCECR) workshop and of the Winter Conference on Business Intelligence (WCBA). The authors are grateful for financial support from the Jeremy Collier Foundation, Blavatnik ICRC at Tel Aviv University, and the Henry Crown Center for Business Research.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix A: Selecting a (non-data-driven) benchmark

In this Appendix we outline the evaluations of three non-data-driven payment policies with the goal of identifying a robust alternative. Per the discussion in the body of the paper, a robust policy is one which yields a consistent performance and which does not perform poorly under any setting. We consider three alternatives, two of which are optimal under some settings. Specifically, the first policy, henceforth referred to as *Minpay*, always offers the lowest payment for labels. A second policy, *Maxpay*, always offers the highest payment. The third policy, henceforth referred to as *UNIFORM*, acquires labels at a representative set of payments—specifically, in each batch of payments, *UNIFORM* draws uniformly at random the payment at which labels are acquired. In some settings, the *Minpay* and *Maxpay* policies have the optimal bias, and will yield the best model performance for a given payment. For example, *Minpay* is the most cost-effective policy in a setting where the prevailing trade-off in the market between payment and quality is “fixed”—namely, when the least costly labels yield the same labeling quality at any pay rate. However, both the *Minpay* and *Maxpay* policies are not robust and yield very poor performance under some settings.

Figure 11 shows the performance of the *Maxpay*, *Minpay*, and *UNIFORM* policies for different labeling tasks when the trade-offs between payment and quality reflect the Fixed, Concave, and Asymptotic trade-offs that have been reported in prior work to arise in online labor markets. As shown, both *Minpay* and *Maxpay* exhibited good performance in some settings, but they also yielded particularly poor performance in other settings. For example, in Fig. 11a, d, g it is evident that neither *Minpay* nor

Fig. 11 Performances of Minpay, Maxpay, and Uniform for different labeling tasks and trade-offs

Maxpay is optimal under a concave trade-off between payment and labeling quality. In this setting, in Fig. 11d, g Minpay policy actually yields particularly poor performance. By contrast, Fig. 11c, f, i shows that when labeling quality is fixed across payments, Minpay yields the best performance, and Maxpay yields the worst performance. When the trade-off between payment and label quality is asymptotic, as shown in Fig. 11b, e, h, Maxpay and Uniform produce good performances, whereas Minpay performs miserably. Overall, both Minpay and Maxpay lack robustness and they each exhibit

very poor performance under some settings. By contrast, the Uniform policy yields the most robust performance and can be relied on in practice not to fail miserably.

Appendix B: Changing trade-off between payment and quality

Figure 12 shows the performance of ALP-MTR and the UNIFORM policy under market settings in which the trade-off between payment and quality changes from one trade-off between payment and quality to another. As shown, ALP-MTR's performance is often significantly better and is otherwise comparable to that of the UNIFORM policy. Overall, ALP-MTR is preferable to UNIFORM in these settings.

Fig. 12 Comparison of ALP-MTR to UNIFORM under changing trade-off between payment and quality

Fig. 12 continued

Appendix C: Additional robustness evaluations

We evaluate ALP-MTR using two different popular classification algorithms to derive models from the acquired labels. Specifically, in the experiments below we use Support Vector Machine (SVM) and Bagging rather than Random Forest as the model inducer. For SVM we employed the SVM classifier in R package *e1071*, using the default settings. For Bagging, we used the bagging classifier, in R package *ipred*, with default settings.

For different data sets and trade-offs, Table 7 lists the average cost savings for inducing SVM models achieved by ALP-MTR to yield the best SVM model performance

Table 7 Cost savings produced by ALP-MTR relative to the uniform policy when models are induced by an SVM inducer

Dataset	Trade-off	Cost savings enabled by ALP-MTR (%)
Mushroom	Asymptotic	11.9
	Concave	17.1
	Fixed	15.4
Spam	Asymptotic	43.8
	Concave	73.8
	Fixed	27.1
Pendigits	Asymptotic	15.0
	Concave	29.8
	Fixed	22.1
Average savings		28.4

Table 8 Cost savings produced by ALP-MTR relative to the uniform policy when models are induced by a Bagging inducer

Dataset	Trade-off	Cost savings enabled by ALP-MTR (%)
Mushroom	Asymptotic	55.6
	Concave	58.9
	Fixed	33.8
Spam	Asymptotic	37.3
	Concave	51.4
	Fixed	14.3
Pendigits	Asymptotic	51.4
	Concave	32.0
	Fixed	21.9
Average savings		39.6

exhibited by the UNIFORM policy. As shown, ALP-MTR yields a significant reduction in the costs necessary to yield the best model performance achieved by the UNIFORM payment policy. Specifically, Table 7 shows that, on average, ALP-MTR yields 28.4% savings in labeling costs, across different settings and labeling tasks. Figure 13 shows the AUC performance curves achieved by each approach for different labeling costs. Overall, ALP-MTR is a more advantageous payment policy and it never yields worse models for a given cost than can be achieved by the uniform policy.

For different data sets and trade-offs, Table 8 lists the average cost savings for inducing Bagging models achieved by ALP-MTR to yield the best Bagging model performance exhibited by the UNIFORM policy. Again, as shown, ALP-MTR yields a significant reduction in the costs necessary to yield the best model performance. In average, ALP-MTR yields 39.6% savings in labeling costs, across different settings

Fig. 13 Performance of ALP-MTR compared to the UNIFORM baseline while using the SVM induction algorithm

and labeling tasks. Figure 14 shows the AUC performance curves achieved by each approach for different labeling costs. Similarly, as observed, ALP-MTR is a more advantageous payment policy and it never yields worse models for a given cost than can be achieved by the uniform policy.

Fig. 14 Performance of ALP-MTR compared to the UNIFORM baseline while using the Bagging induction algorithm

Appendix D: Extending ALP-MTR to include payment regulation

To identify payments that can yield the best model performance at the lowest cost, ALP-MTR uses training instances labeled previously for different payments. Because labels are noisy in our settings, having more acquisitions for a given payment can improve the estimation of the contribution to induction of labels acquired for the corresponding payment to producing a better model. However, it is possible that due to noise, cost-effective payments are deemed undesirable. Further, because such a payment is not selected, the estimation is not improved and the advantageous payment will continue to be ignored. This problem may be particularly significant when the trade-off between labeling payment and quality varies over time.

Fig. 15 Performance of ALP-MTR and ALP-MTR-PAYMENT-REG for different trade-offs between payment and quality

In this appendix, we extended ALP-MTR to include payment regulation. This ALP-MTR variant, ALP-MTR-PAYMENT-REG, uses ALP-MTR's selection of payments as before; however, it acquires labels at a given payment level if this payment has not been selected in the recent t consecutive batch acquisitions. While ALP-MTR-PAYMENT-REG aims to enhance the estimation of the benefits of different payment levels, it likely also incurs opportunity costs by offering payments on the market that are not cost-effective. We evaluated ALP-MTR-PAYMENT-REG for $t = 10$, $t = 20$ and, $t = 30$, all of which produced similar results. Figures 15 and 16 show the performance of ALP-

Fig. 16 Performance of ALP-MTR and ALP-MTR-PAYMENT-REG when the trade-off between payment and quality changes

Fig. 16 continued

MTR-PAYMENT-REG for $t = 10$, for different trade-offs (Fig. 15) and when the trade-off between payment and quality changes (Fig. 16). As shown, ALP-MTR-PAYMENT-REG produces similar performance to that of ALP-MTR, and in most cases it slightly underperforms ALP-MTR. Our results suggest that the standard ALP-MTR approach effectively captures the cost-effectiveness of different payments, and in settings where the trade-off between payment and quality varies, ALP-MTR effectively adapts to these changes. Thus, enforcing acquisitions at different payment levels, as done by ALP-MTR-PAYMENT-REG, can improve performance in some cases, but often yields inferior performance to that achieved by the standard ALP-MTR.

Appendix E: Evaluating the effect of different batch sizes

In this Section we consider the effect of the acquisition batch size on ALP-MTR's performance. Figure 17 shows the performances of ALP-MTR when at each phase batch sizes of 5, 10, 15, or 20 instances are acquired. As shown in most cases, altering the batch size generally does not have a significant effect on performance. However, in a few cases, increasing the batch size value slightly improved the results compared to running ALP-MTR with very low batch size value ($b = 5$).

Fig. 17 Performance of ALP-MTR using batch sizes of 5, 10 and 15 instances

Appendix F: Insights into label quality and learning

In this appendix we examine the effect on model learning of different label qualities (which can be acquired for different payments per label) and how this effect varies across domains. Figure 18 shows results for how different labeling qualities yield different learning curves; similarly, whether or not a larger number of instances labeled at a lower quality can yield the same performance enabled with a smaller set of higher quality labels, also varies across domains. Because different label qualities are acquired at different costs, these results underscore that different label acquisition strategies can be more advantageous under different settings. Thus, a data-driven

Fig. 18 AUC as a function of the number of instances and labeling quality (60–100%)

approach for selecting payment for labeling can produce versatile acquisition strategies across different settings.

Appendix G: Varying the number of instances in the initialization phase

During the initialization phase, ALP-MTR compiles an initial dataset from which learning proceeds, and the number of batch acquisitions performed during initialization is controlled by the *init_batch* parameter. Setting a larger *init_batch* value will result in a larger set that may provide a more representative sample, but also delays making informed purchase decisions.

In this appendix, we evaluate the impact of this trade-off by both increasing and decreasing the number of initial batch purchases by 30%. As shown, performance can be often improved with a shorter initialization phase, but performance is not significantly affected by the *init_batch* parameter (Fig. 19).

Fig. 19 Performance of ALP-MTR using default initialization duration (300 instances) and when increasing or decreasing the initialization phase by $\pm 30\%$

Appendix H: Varying the number of payment options

ALP-MTR considers a set of payment levels $\{c_1, \dots, c_{p_0}\}$ and selects advantageous payments in each acquisition phase. In the experiments presented here, we considered three payment options. As we note in the paper, we followed prior work for the payment range and the number of payment alternatives from which payment methods can be selected. Specifically, prior studies considered 2–4 payment levels, ranging from a minimum payment of \$0.01–\$0.03 to a maximum payment of \$0.1–\$0.25 (e.g., Feng et al.

Fig. 20 Performance of ALP-MTR using 3 (default) and 5 payment options

2009; Kazai 2011; Kazai et al. 2013; Mason and Watts 2010; Rogstad et al. 2011). Given the minimum and maximum payment levels, the set of payment options was produced by evenly partitioning the range of payments per label into three equal-size ranges, and where the midpoint of each range corresponds to each of the payment alternatives. Such partitioning, however, can be more refined, resulting in a larger number of payment options. In this appendix, we evaluated the effect on performance of increasing the number of payment options from the default set of payment options we considered in the empirical evaluations (0.02, 0.14, and 0.25 USD per label) to 5 payment options (0.02, 0.08, 0.14, 0.19, and 0.25 USD per label), while maintaining the same duration of the initialization phase. As shown in Fig. 20, ALP-MTR achieves either comparable or better performance when increasing the number of possible payment levels.

Appendix I: The cost of learning

To offer insights into the potential for future improvements of our approach, we consider the settings where the trade-off between payment per label and quality is “fixed” (Mason and Watts 2010), and thus labeling quality does not vary for different payments. While identifying the optimal payment at each step in the general case involves identifying, at each step, the payment that would yield the best improvement in performance for a given budget, an assessment that in itself can be noisy, the optimal payment choice for the fixed trade-off, namely, the lowest payment, is both known and remains the same throughout the acquisition phases. The difference between the choices made by an oracle that knows the optimal payment level and a method that is data-driven offers an upper bound for future improvements to the approach.

Table 9 shows percentage of acquisition batches in which ALP-MTR selects the optimal, lowest payment option, second best decision (second-lowest payment) and worst payment option (highest payment). As shown, ALP-MTR selects the optimal payment level of individual purchase decisions compared to the UNIFORM policy which would evenly distribute the payment decisions. Furthermore, it is important to note that these results are likely to be a conservative estimate of ALP-MTR’s overall performance. This evaluation considers a setting in which a fixed trade-off between payment and quality exists. In such a setting, ALP-MTR can only benefit from detecting that a lower payment level results in similar improvement in AUC as a higher payment level. But it cannot affect model quality by a selected payment that leads to different model performance. Indeed we observe that our method obtains much better improvements (on the aggregate level) in other market conditions (see Table 3).

Finally, we use the fixed trade-off setting in which the optimal decision is known in advance (i.e., in this setting, it is always best to pay the minimal price) to gauge the savings that such an oracle-based payment scheme will obtain compared to ALP-MTR in a similar way as we calculated the ALP-MTR cost savings compared to the baseline method (see again the illustration in Fig. 4 in the main body of the paper). We find that such an oracle based payment scheme obtains an improvement in cost of 79.6%, 77.6%, and 79.3% for the SPAM, Mushroom and Pen Digits datasets, respectively.

Table 9 Batch level decision quality distribution (3 payment options)

Dataset	Best decision (%)	Second best decision (%)	First or second best decision (%)	Worst decision (%)
SPAM	54.5	29.1	83.60	16.4
Mushroom	51.3	31.5	82.80	17.2
Pen Digit	63.7	24.5	88.20	11.8
Average	56.5	28.4	84.90	15.1

Appendix J: Using ALP-MTR to select payment per label for repeated labeling

Repeated labeling refers to the acquisition of multiple noisy labels for the same instance so as to infer the most likely label and improve the accuracy of the labels. Importantly, methods for repeated labeling do not determine which payments to offer labelers per label, but assume that the payment per label is fixed and somehow pre-determined. Repeated labeling thus aims to reduce noise (error) in the data by acquiring multiple labels for the same instance at a fixed and pre-determined cost per label, and use the set of labels available for a given instance to infer the most likely label. For example, a popular approach is to infer the majority label for each instance (e.g., Lee et al. 2013; Mason and Suri 2012). Because repeated labeling does not determine what payment per label to offer labelers, in this appendix we demonstrate how ALP-MTR can be applied to identify advantageous payments per label when repeated labeling is used. Indeed, ALP-MTR and repeated labeling address complementary tasks, and can thus be applied in conjunction. In particular, we apply ALP-MTR first to identify the cost-effective payment per label at which to acquire the next batch of labels; multiple labels for the same instance are then purchased at the payment determined by ALP-MTR, and these labels are used by repeated labeling to improve the labeling accuracy.

We evaluated the performance of ALP-MTR with repeated labeling compared to the performance of repeated labeling when the UNIFORM policy (rather than ALP-MTR) was used to select the payment per label. In these experiments, repeated labeling acquires three labels per instance, and an instance is labeled by the majority label. Table 10 shows the average cost savings achieved by ALP-MTR, and Fig. 21 shows the predictive performance achieved for different label acquisition costs by each approach. As shown, when multiple labels are acquired for repeated labeling, ALP-MTR can be used effectively to identify advantageous labeling payments, and it yields an average savings of 23% in labeling costs. Overall, using ALP-MTR to select the payment per label for repeated labeling often yields superior model performance for a given labeling

Table 10 Cost savings by ALP-MTR when used to select payment per label for repeated labeling

Dataset	Trade-off	Cost savings enabled by ALP-MTR (%)
Mushroom	Asymptotic	31.0
	Concave	33.9
	Fixed	6.9
Spam	Asymptotic	37.7
	Concave	36.6
	Fixed	10.8
Pendigits	Asymptotic	27.8
	Concave	19.0
	Fixed	6.4
Average savings		23.3

Fig. 21 Performance of ALP-MTR and UNIFORM with repeated labeling. Applying ALP-MTR to determine the payments used by repeated labeling yields better models for a given cost as compared to when payments are determined by UNIFORM

cost and is otherwise comparable to when repeated labeling is applied naively, without ALP-MTR's selection of cost-effective payments.

References

- Abe N, Mamitsuka H (1998) Query learning strategies using boosting and bagging. In: Proceedings of the international conference on machine learning (ICML). Morgan Kaufmann, pp 1–9
- Chen DL, Horton JJ (2016) Research note—are online labor markets spot markets for tasks? A field experiment on the behavioral response to wage cuts. *Inf Syst Res* 27(2):403–423

- Dai P, Lin CH, Mausam M, Weld DS (2013) Pomdp-based control of workflows for crowdsourcing. *Artif Intell* 202:52–85
- Dalvi N, Dasgupta A, Kumar R, Rastogi V (2013) Aggregating crowdsourced binary ratings. In: *Proceedings of the 22nd international conference on world wide web*. ACM, New York, pp 285–294
- Downs JS, Holbrook MB, Sheng S, Cranor LF (2010) Are your participants gaming the system? Screening mechanical Turk workers. In: *Proceedings of the SIGCHI conference on human factors in computing systems*. New York: ACM, pp 2399–2402
- Feng D, Besana S, Zajac R (2009) Acquiring high quality non-expert knowledge from on-demand workforce. In: *Proceedings of the 2009 workshop on the people's web meets NLP: collaboratively constructed semantic resources*. Association for Computational Linguistics, Stroudsburg, pp 51–56
- Ipeirotis PG, Provost F, Sheng VS, Wang J (2014) Repeated labeling using multiple noisy labelers. *Data Min Knowl Disc* 28(2):402–441
- Karger DR, Oh S, Shah D (2011) Iterative learning for reliable crowdsourcing systems. In: *Proceedings of advances in neural information processing systems: 25th annual conference on neural information processing*, December 12–14
- Karger DR, Oh S, Shah D (2014) Budget-optimal task allocation for reliable crowdsourcing systems. *Oper Res* 62(1):1–24
- Kazai G (2011) In search of quality in crowdsourcing for search engine evaluation. In: Clough P et al (eds) *Advances in information retrieval*. Springer, Berlin Heidelberg, pp 165–176
- Kazai G, Kamps J, Milic-Frayling N (2013) An analysis of human factors and label accuracy in crowd-sourcing relevance judgments. *Inf Retrieval* 16(2):138–178
- Kong D, Saar-Tsechansky M (2014) Collaborative information acquisition for data-driven decisions. *Mach Learn* 95(1):71–86
- Kumar A, Lease M (2011) Modeling annotator accuracies for supervised learning. In: *Proceedings of the workshop on crowdsourcing for search and data mining (CSDM), at the fourth ACM international conference on web search and data mining (WSDM)*, pp 19–22
- Lee D, Hosanagar K, Nair H (2013) The effect of advertising content on consumer engagement: evidence from Facebook (working paper). Available at SSRN 2290802
- Lewis D, Gale W (1994) A sequential algorithm for training text classifiers. In: *Proceedings of the ACM SIGIR conference on research and development in information retrieval*. ACM/Springer, pp 3–12
- Lichman M (2013) UCI machine learning repository. University of California, School of Information and Computer Science, Irvine. <http://archive.ics.uci.edu/ml>
- Lin CH, Weld DS (2014) To re (label), or not to re (label). In: *Second AAAI conference on human computation and crowdsourcing*
- Lin CH, Mausam M, Weld DS (2012) Crowdsourcing control: moving beyond multiple choice. In: *UAI*
- Lin CH, Mausam M, Weld DS (2016) Re-active learning: active learning with relabeling. In: *AAAI*, pp 1845–1852
- Mason W, Suri S (2012) Conducting behavioral research on Amazon's mechanical Turk. *Behav Res Methods* 44(1):1–23
- Mason W, Watts DJ (2010) Financial incentives and the performance of crowds. *ACM SIGKDD Explor Newsl* 11(2):100–108
- Paolacci G, Chandler J, Ipeirotis PG (2010) Running experiments on amazon mechanical Turk. *Judgm Decis Mak* 5(5):411–419
- Raykar VC, Yu S, Zhao LH, Valadez GH, Florin C, Bogoni L, Moy L (2010) Learning from crowds. *J Mach Learn Res* 11(11):1297–1322
- Rodrigues F, Pereira F, Ribeiro B (2013) Learning from multiple annotators: distinguishing good from random labelers. *Pattern Recogn Lett* 34(12):1428–1436
- Rogstadius J, Kostakos V, Kittur A, Smus B, Laredo J, Vukovic M (2011) An assessment of intrinsic and extrinsic motivation on task performance in crowdsourcing markets. *ICWSM* 11:17–21
- Saar-Tsechansky M, Provost F (2004) Active sampling for class probability estimation and ranking. *Mach Learn* 54(2):153–178
- Sheng VS, Provost F, Ipeirotis PG (2008) Get another label? Improving data quality and data mining using multiple, noisy labelers. In: *Proceedings of the 14th ACM SIGKDD international conference on knowledge discovery and data mining*, pp 614–622
- Wang J, Ipeirotis PG, Provost F (2017) Cost-effective quality assurance in crowd labeling. *Inf Syst Res* 28:137–158

- Wauthier FL, Jordan MI (2011) Bayesian bias mitigation for crowdsourcing. In: Bartlett P, Pereira F, Shawe-Taylor J, Zemel R (eds.) *Advances in neural information processing systems (NIPS)*, pp 1800–1808
- Yang L, Carbonell J (2012) Adaptive Proactive learning with cost-reliability trade-off. In: Seel NM (ed) *Encyclopedia of the sciences of learning*. Springer, New York, pp 121–127
- Zhang Jing, Xindong Wu, Sheng Victor S (2015) Active learning with imbalanced multiple noisy labeling. *IEEE Trans Cybern* 45(5):1095–1107
- Zhang Jing, Xindong Wu, Sheng Victor S (2016) Learning from crowdsourced labeled data: a survey. *Artif Intell Rev* 46(4):543–576
- Zhou D, Basu S, Mao Y, Platt JC (2012) Learning from the wisdom of crowds by minimax entropy. In: *Advances in neural information processing systems*, pp 2204–2212

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Tomer Geva¹ · Maytal Saar-Tsechansky² · Harel Lustiger¹

Tomer Geva
tgeva@tau.ac.il

¹ Coller School of Management, Tel Aviv University, 6997801, Ramat Aviv, Tel-Aviv, Israel

² McCombs School of Business, The University of Texas at Austin, Austin, TX, USA