Journal of Machine Learning Research 18 (2018) 1–46 Submitted 04/17; Revised 04/18; Published 05/18

Making Better Use of the Crowd: How Crowdsourcing Can Advance Machine Learning Research

Jennifer Wortman Vaughan JENN@MICROSOFT.COM
Microsoft Research
641 Avenue of the Americas, 7th Floor
New York, NY 10011

Editor: Qiang Liu

Abstract

This survey provides a comprehensive overview of the landscape of crowdsourcing research, targeted at the machine learning community. We begin with an overview of the ways in which crowdsourcing can be used to advance machine learning research, focusing on four application areas: 1) data generation, 2) evaluation and debugging of models, 3) hybrid intelligence systems that leverage the complementary strengths of humans and machines to expand the capabilities of AI, and 4) crowdsourced behavioral experiments that improve our understanding of how humans interact with machine learning systems and technology more broadly. We next review the extensive literature on the behavior of crowdworkers themselves. This research, which explores the prevalence of dishonesty among crowdworkers, how workers respond to both monetary incentives and intrinsic forms of motivation, and how crowdworkers interact with each other, has immediate implications that we distill into best practices that researchers should follow when using crowdsourcing in their own research. We conclude with a discussion of additional tips and best practices that are crucial to the success of any project that uses crowdsourcing, but rarely mentioned in the literature.

Keywords: crowdsourcing, data generation, model evaluation, hybrid intelligence, behavioral experiments, incentives, mechanical turk

1. Introduction

Crowdsourcing allows us to harness the power of human computation to solve tasks that are notoriously difficult to solve with computers alone, such as determining whether or not an image contains a tree, rating the quality of a website, or verifying the phone number of a business. The machine learning community was early to embrace crowdsourcing as a tool for quickly and inexpensively obtaining the vast quantities of labeled data needed to train machine learning systems. Crowdsourcing has been used to generate the image annotations that are needed to train computer vision systems (Deng et al., 2009; Patterson and Hays, 2012; Raykar et al., 2010; Wah et al., 2011), provide the linguistic annotations needed for common natural language processing tasks (Callison-Burch and Dredze, 2010; Snow et al., 2008), and collect the relevance judgments needed to optimize search engines (Alonso, 2013; Alonso et al., 2008). This simple idea—that crowds could be used to generate training data for machine learning algorithms—inspired a flurry of algorithmic work on how to best elicit and aggregate potentially noisy labels (e.g., Ghosh et al., 2011; Karger et al., 2011; Khetan and Oh, 2016; Liu et al., 2012a; Sheng et al., 2008; Welinder et al., 2010; Zhang et al.,

2016b; Zhou et al., 2012), still an active area of research (Zheng et al., 2017). Meanwhile, machine learning researchers have begun to put crowdsourcing to use in other ways, most commonly as a tool to evaluate and debug machine learning models (Chang et al., 2009; Ribeiro et al., 2016).

Crowdsourcing has flourished as a research tool outside of the machine learning community as well. In human-computer interaction and related fields, researchers are building "hybrid intelligence systems" with the goal of expanding the capabilities of current AI technology by incorporating humans in the loop (e.g., Bernstein et al., 2010; Kamar, 2016; Lasecki et al., 2017; Zhang et al., 2012). And psychologists and social scientists have increasingly moved experiments that traditionally would have been run in physical labs onto crowdsourcing platforms (Buhrmester et al., 2011; Mason and Suri, 2012). While these bodies of research are less well known within the machine learning community, there are countless opportunities for machine learning research to both influence and benefit from these lines of work. For example, human-in-the-loop clustering algorithms have been designed that produce better clusters by drawing on the common sense knowledge and experience of the crowd (Gomes et al., 2011; Heikinheimo and Ukkonen, 2013; Tamuz et al., 2011), while behavioral experiments run on the crowd offer insight about how to encourage human trust in algorithmic predictions (Dietvorst et al., 2015, 2016; Dzindolet et al., 2002).

The first goal of this survey is to expand the horizons of how machine learning researchers think about crowdsourcing, providing a broad overview of ways in which crowdsourcing can benefit (and sometimes benefit from) machine learning research. Unlike other surveys, which go into greater depth on algorithms for aggregating crowdsourced labels (Zhang et al., 2016a; Zheng et al., 2017), we address the label aggregation problem only briefly, devoting relatively more attention and detail to applications that are less well known within the machine learning community, in the hope of inspiring new connections and directions of research. We break the applications into four broad categories:

- **Data generation.** (Section 2) Crowdsourcing platforms are well suited to generating data, but challenges arise since the data supplied by crowdworkers can be prone to errors. We start with a brief review of two lines of research aimed at improving the quality of crowdsourced labels. The first assumes that data points are redundantly assigned to multiple workers and seeks algorithms for aggregating workers' responses that take into account the quality of individual workers (e.g., Zhang et al., 2016a; Zheng et al., 2017). Though there are many notable exceptions, much of this work builds on the influential model and expectation-maximization framework of Dawid and Skene (1979). The second line of work focuses on developing incentive schemes to motivate high quality responses. Much of this work builds on the literature on peer prediction, a framework in which crowdworkers' payments are a function of their own reported labels and the labels of other workers (Jurca and Faltings, 2009; Miller et al., 2005; Radanovic et al., 2016). We then review ways in which crowdsourcing has been applied to generate other forms of data, including transcriptions of printed text (von Ahn et al., 2008), translations of sentences from one language to another (Zaidan and Callison-Burch, 2011), and image annotations (Kovashka et al., 2016).

- **Evaluating and debugging models.** (Section 3) Crowdsourcing is also commonly used to evaluate or debug models, including unsupervised learning models, which can

be difficult to evaluate objectively since there is often no clear notion of ground truth. Along these lines, we discuss the use of crowdsourcing to evaluate the coherence of topic models, generative models used to discover and explore the thematic topics discussed in a set of documents (Chang et al., 2009). We also explore ways in which crowdsourcing has been used to evaluate the interpretability of explanations of predictions in supervised learning settings (Ribeiro et al., 2016) and to debug the components of a pipeline in a complex computer vision system (Mottaghi et al., 2013, 2016; Parikh and Zitnick, 2011).

- **Hybrid intelligence systems.** (Section 4) Hybrid intelligence or "human-in-the-loop" systems advance the capabilities of current AI technology by leveraging the complementary strengths of humans and machines. We explore several compelling examples of hybrid systems that suggest their great potential: hybrid systems for clustering data points (Gomes et al., 2011; Heikinheimo and Ukkonen, 2013; Tamuz et al., 2011), transcribing speech in real time (Kushalnagar et al., 2012; Lasecki and Bigham, 2012; Lasecki et al., 2012, 2013, 2017; Naim et al., 2013), scheduling conference sessions (André et al., 2013; Bhardwaj et al., 2014; Chilton et al., 2014; Kim et al., 2013), and forecasting geopolitical or economic events (Atanasov et al., 2017; Baron et al., 2014; Mellers et al., 2015b). These systems are able to achieve more than would be possible with state-of-the-art machine learning or AI systems alone because they can make use of people's common sense knowledge, life experience, subjective beliefs, and flexible reasoning skills.

- **Behavioral studies to inform machine learning research.** (Section 5) As machine learning becomes a larger part of people's everyday lives, interest in understanding how real people interact with machine learning systems continues to grow. There has been a surge of recent research on questions like how to design machine learning models that are more interpretable (Doshi-Velez and Kim, 2017; Lipton, 2016) or how to understand the ways that algorithmic decisions impact people's lives (Angwin et al., 2016; Barocas and Selbst, 2016). These questions are interdisciplinary in nature and require gaining a better understanding of the underlying principles behind humans' interactions with machine learning and other technological systems. At the same time, psychologists and social scientists have started using crowdsourcing platforms as a fast and easy way to gain access to large pools of subjects for behavioral experiments. This presents a natural opportunity for researchers to conduct studies on crowdsourcing platforms that improve our understanding of how humans interact with technology broadly and machine learning algorithms in particular. Rather than evaluating the way in which people interact with one particular algorithm, this line of research aims to develop an understanding of components of human behavior that could inform the use and design of machine learning systems more broadly. We walk through illustrative examples of behavioral studies aimed at understanding user trust in algorithmic predictions (Dietvorst et al., 2015, 2016; Dzindolet et al., 2002; Poursabzi-Sangdeh et al., 2018) and how users react to online advertising (Goldstein et al., 2013, 2014). The latter study, while not directly motivated by a machine learning question, could provide insights that inform the design of better machine learning algorithms for ad pricing and display.

Having explored these applications and motivated the use of crowdsourcing in machine learning research, the remainder of the survey addresses our second goal: to provide the reader with best practices for crowdsourcing by drawing deeply on the vast and cross-disciplinary literature aimed at studying and understanding the behavior of the crowd itself. In Section 6, we describe several studies that quantify and measure the prevalence of dishonest or spammy behavior on crowdsourcing platforms (Chandler and Paolacci, 2017; Suri et al., 2011; Wessling et al., 2017). We discuss how to set payments for tasks in light of both ethical considerations (Williamson, 2016) and a body of research that explores how the quality and quantity of crowdwork are impacted by monetary incentives (e.g., Buhrmester et al., 2011; Ho et al., 2015; Mason and Watts, 2009; Shaw et al., 2011). We examine how workers react to intrinsic sources of motivation, such as gamification (von Ahn and Dabbish, 2008) or the satisfaction of performing meaningful work (Chandler and Kapelner, 2013; Rogstadius et al., 2011), and how intrinsically motivating workers can increase workers' willingness to complete tasks. Finally, we explore communication and collaboration patterns of workers and see that crowdworkers are not independent and isolated, but rather part of a rich communication network (Gray et al., 2016; Yin et al., 2016). We discuss the implications of this work on the practice of using crowdsourcing for research, outlining best practices that should be followed whether one wishes to use crowdsourcing for data generation, model evaluation, building hybrid systems, running behavioral studies, or beyond.

Section 7 concludes with a discussion of additional tips and tricks that are crucial for the success of any crowdsourcing-based project, but rarely discussed in the literature.

1.1 A Note on Scope

As described above, this survey was written with two overarching goals in mind. The first is to inspire machine learning researchers to discover unexpected applications of crowdsourcing in their own research. Because of this, relatively more space is devoted to applications that may be less familiar to the machine learning community, such as hybrid intelligence systems and behavioral studies of users interacting with technology, compared with more familiar applications like data generation and model evaluation. Full surveys have been written just on the problem of aggregating noisy labels from a crowd; readers who are most interested in this problem are encouraged to look at Zheng et al. (2017), Zhang et al. (2016a), or the comprehensive summary of related work in Zhang et al. (2016b).

The second goal is to introduce machine learning researchers to the extensive cross-disciplinary literature on crowd behavior and motivate why understanding this literature is crucial for successfully employing crowdsourcing in research. Because of this, Section 6 draws heavily on work from outside the machine learning community, and emphasizes experimental work over pure theory.

Several other crowdsourcing surveys have been published, each focused on a different set of themes and problems. Readers interested in gaining a broader perspective on crowdsourcing may be interested in recent surveys and guides from the computer vision community (Kovashka et al., 2016), the databases community (Li et al., 2016), and the marketing community (Goodman and Paolacci, 2017), or the older but still widely cited and applicable how-to guides on crowdsourcing user studies (Kittur et al., 2008) and behavioral research (Mason and Suri, 2012).

1.2 Background on Mechanical Turk and Other Crowdsourcing Platforms

In this survey, we use the term crowdsourcing very generally to encompass both paid and volunteer crowdwork, done by experts or nonexperts, on any general or specialized crowdsourcing platform. Regardless, the majority of research covered was conducted using paid crowdworkers, and the majority of that on one particular platform, Amazon Mechanical Turk.

Amazon Mechanical Turk[1] is the most commonly used crowdsourcing platform among researchers. It is designed for crowdsourcing relatively small "microtasks" (often referred to as "HITs," for human intelligence tasks) such as labeling a set of images or completing a survey, though it can also be used for more complex short-term tasks, such as participating in behavioral experiments (Mason and Suri, 2012) or even writing fiction (Kim et al., 2017). Task requesters come to Mechanical Turk to post their tasks, stating up front the amount of money that they are willing to pay to have their tasks completed. Requesters can also specify certain criteria that a crowdworker must meet to be eligible for their task, such as having an approval rate of more than a particular amount (say, 97%) on previous tasks or being located in a particular country. Crowdworkers can then browse the set of tasks available and choose the tasks they would like to work on. After a crowdworker completes a task, the requester approves her work and a payment is made.

Although Mechanical Turk is used broadly in the research community, it is not the right choice for everyone. In particular, it can be difficult to use from outside of the United States. Luckily, many alternatives are available. For example:

- CrowdFlower[2] is a crowdsourcing platform widely used in both industry and research. CrowdFlower offers specialized enterprise solutions for businesses with artificial intelligence and data science needs including search relevance evaluation, sentiment analysis, and data classification.

- ClickWorker[3] is a German crowdsourcing platform that attracts European workers. It provides support for specialized tasks such as translation, web research, and web content generation. It also provides tools for mobile crowdsourcing.

- Prolific Academic[4] is a UK-based crowdsourcing platform focused primarily on connecting researchers with participants for behavioral or user studies.

- Upwork[5] is an online freelancer marketplace focused not on microtasks but rather on larger scale jobs such as writing an article or designing a website.

- TopCoder[6] hosts crowdsourcing contests (Chawla et al., 2015; DiPalantino and Vojnovic, 2009; Gao et al., 2012) in which coders design and develop software in response to specific challenges and compete for prizes. Unlike on the other sites mentioned, only those who create the top-judged submissions are paid.

1. https://www.mturk.com
2. https://www.crowdflower.com
3. https://www.clickworker.com
4. https://www.prolific.ac
5. https://www.upwork.com
6. https://www.topcoder.com

Several researchers have compared commercially available platforms along different dimensions. Vakharia and Lease (2015) provided a thorough qualitative content analysis of seven platforms: ClickWorker, CrowdComputing Systems (now WorkFusion), CloudFactory, CrowdFlower, CrowdSource, MobileWorks (now LeadGenius), and oDesk (now Upwork), examining factors like infrastructure and tools, support for fraud protection, and quality of work. Peera et al. (2017) provided a detailed experimental comparison of three platforms: Mechanical Turk, CrowdFlower, and Prolific Academic. They compared dropout rates, response rates, workers' performance on attention-check questions, workers' reliability, workers' familiarity with common psychology studies (which would signal an overused population of subjects), and the ability to replicate classic psychology studies on each platform. They found, for example, that workers on both CrowdFlower and Prolific Academic were less familiar with common psychology studies and less dishonest than workers on Mechanical Turk, with Prolific Academic producing higher quality data than CrowdFlower.

2. Data Generation

Perhaps the most common application of crowdsourcing within the machine learning community is data generation. We first describe techniques for crowdsourcing binary or categorical labels, reviewing the literature on how to improve label quality through redundancy and incentives. We then discuss several examples of ways in which crowdsourcing has been used to generate more complex forms of data, such as image annotations and translations of text. As mentioned above, entire surveys could be written on the topic of crowdsourced data generation alone, and indeed some have (Zhang et al., 2016a; Zheng et al., 2017). Our treatment of this topic is comparatively brief, intended only to give the reader the flavor of this work with pointers to the literature for readers who wish to learn more.

2.1 Generating Binary or Categorical Labels

We start with the setting in which crowdworkers are presented with unlabeled data instances (for instance, websites) and are asked to supply labels (for instance, a binary label indicating whether or not the website contains profanity). The main challenge arises from the fact that the supplied labels are often noisy or inaccurate, either because workers are imperfect or because workers are unmotivated to put high effort into the labeling task. There are two primary lines of work aimed at improving the quality of crowdsourced labels. The first assumes that each instance is presented to multiple crowdworkers and explores algorithmic techniques for aggregating the workers' responses (e.g., Abraham et al., 2016; Aydin et al., 2014; Demartini et al., 2012; Fan et al., 2015; Gao et al., 2016; Ghosh et al., 2011; Ho et al., 2013; Karger et al., 2011, 2014; Khetan and Oh, 2016; Kim and Ghahramani, 2012; Li et al., 2014a,b; Liu et al., 2012a,b; Ma et al., 2015; Raykar et al., 2010; Shah and Lee, 2018; Sheng et al., 2008; Tian and Zhu, 2015; Venanzi et al., 2014; Welinder et al., 2010; Whitehill et al., 2009; Zhang et al., 2016b; Zhou et al., 2012). The second explores the use of well-designed incentives to encourage higher quality work (e.g., Dasgupta and Ghosh, 2013; Ho et al., 2016; Kamar and Horvitz, 2012; Kamble et al., 2015; Radanovic et al., 2016; Shah et al., 2015; Shah and Zhou, 2016a,b).

Much of the research on label aggregation builds on the model proposed in the seminal paper of Dawid and Skene (1979). The basic Dawid-Skene model assumes that instances are

homogeneous. A worker's probability of labeling any given instance correctly is controlled by one or more worker-specific quality parameters. In the original model, each worker's quality is governed by a latent confusion matrix that specifies the worker's probability of choosing each possible label conditioned on the true label of the instance. A variety of extensions have been studied. For example, several papers have explored models in which difficulty varies by instance (Khetan and Oh, 2016; Ma et al., 2015; Whitehill et al., 2009; Zhou et al., 2012) or workers have diverse sets of skills (Fan et al., 2015; Ho et al., 2013; Welinder et al., 2010). Most of this work assumes a fixed assignment of instances to workers, but some assumes that workers arrive online and are assigned to instances upon arrival (Abraham et al., 2016; Ho and Vaughan, 2012; Ho et al., 2013; Karger et al., 2011, 2014). Some work requires the existence of "gold-standard" or "control" tasks (Le et al., 2010; Liu et al., 2013; Oleson et al., 2011), data instances for which ground truth labels are known a priori, while some does not. Extensions of the Dawid-Skene model have also been applied to the problem of aggregating ordinal labels or rankings (Zhou et al., 2014).

While there are exceptions, much of this research builds on the general expectation-maximization (EM) algorithmic framework that Dawid and Skene (1979) proposed. This framework is outlined in Algorithm 1. The basic approach involves iteratively updating estimates of both workers' quality parameters and the labels of each instance. At each step, quality parameters are updated treating the current label estimates as ground truth. The instance labels are then updated to their most likely values treating the quality parameters as ground truth. The details of these updates vary. Zheng et al. (2017) provide a thorough survey and empirical comparison of seventeen algorithms that are based on this general framework, characterizing them in terms of the way in which instances and workers are modeled as well as the specifics of how the calculations of quality parameters and label assignments are made (through what they call direct computation, using optimization methods, or using probabilistic graphical models). While their experiments reveal that different algorithms perform best on different data sets, they show that the original Dawid-Skene algorithm is itself fairly robust in practice.

Algorithm 1 The basic EM framework of Dawid and Skene (1979).

Input: Sets of worker-generated labels for each instance
Initialize each instance's label based on a simple majority vote
repeat
  for all Workers w **do**
    Calculate w's quality parameter(s), treating each instance's current label as ground truth
  end for
  for all Instances i **do**
    Calculate the most likely label for i, treating each worker's approximated quality parameter(s) as ground truth
  end for
until Label assignments have converged
Output: The current label assignments for each instance

One broadly studied class of incentive schemes for crowdsourcing is built on the literature on peer prediction (e.g., Dasgupta and Ghosh, 2013; Jurca and Faltings, 2009; Kamar and Horvitz, 2012; Kamble et al., 2015; Miller et al., 2005; Prelec, 2004; Prelec et al., 2017; Radanovic and Faltings, 2013; Waggoner and Chen, 2014; Witkowski and Parkes, 2012), a framework in which workers are rewarded based on a function of their own reported labels and the reports of others. Under many peer prediction mechanisms, higher rewards are given to reports that are "surprisingly common" among workers, where "surprise" could be measured, for example, in terms of a common prior or the frequency of a label. As just one example, to determine the payment for a worker w who labels a particular instance as x, the mechanism proposed by Radanovic et al. (2016) selects another worker w' at random and compares the label provided by w to the label provided by w'. If w' reports x (so the two workers' labels agree), then w receives payment $1/f$ where f is the empirical frequency with which other workers have reported the label x over all instances. If w' does not report x, then w receives no payment. Radanovic et al. (2016) showed that under some assumptions on workers' beliefs, truthful reporting (i.e., each worker reporting the label they think is most likely) is an equilibrium under this mechanism.

The primary benefit of peer prediction style methods is that there is no need to know ground truth labels in order to calculate payments. One major drawback is that most such methods leave open the opportunity for workers to benefit by coordinating and colluding on incorrect reports, behavior that has been observed when these methods were tested on real workers (Gao et al., 2014). There is an active line of research on developing peer prediction techniques for which such undesirable equilibria do not exist, or at least are less profitable than truth telling (Agarwal et al., 2017; Dasgupta and Ghosh, 2013; Shnayder et al., 2016).

When gold-standard labels are available for some instances, workers can be rewarded directly for the quality of labels they supply for these instances, using either simple bonuses for accuracy (Ho et al., 2015; Shaw et al., 2011) or more complex incentive schemes (Ho et al., 2016; Shah et al., 2015; Shah and Zhou, 2016a,b). See Section 6.3 for a more detailed discussion of how workers respond to monetary incentives in practice.

2.2 Generating Transcriptions, Translations, and Image Annotations

Crowdsourcing is also used to generate more complex and free-form labels, such as transcriptions, translations of language, or image annotations.

Perhaps the best known example of a crowdsourcing system for transcription is reCAPTCHA (von Ahn et al., 2008). CAPTCHAs (or Completely Automated Public Turing tests to tell Computers and Humans Apart) are security tools designed to prevent bots from accessing online services (von Ahn et al., 2003, 2004). People attempting to access a website or create an account are asked to perform a task that is difficult for computers to perform but that humans find easy, such as reading and transcribing distorted characters. CAPTCHAs are used to stop ticket scalpers from using bots to buy out popular shows and to prevent spammers from opening arbitrary numbers of email accounts.

Von Ahn et al. (2008) found a way to put the vast quantities of human effort exerted on solving CAPTCHAs to use, harnessing this effort to digitize old books that current optical character recognition (OCR) systems were unable to handle. Their reCAPTCHA system presents two images of words, both taken from scanned text on which state-of-the-

art OCR systems have failed. One of these images is a gold-standard data point for which the correct transcription is already known. This is the image used to test whether or not the transcriber is human. The true label of the second image is unknown. By completing the CAPTCHA, the human is essentially entering a label for this data. Since reCAPTCHA was acquired by Google, similar techniques have been used to annotate images and build other large-scale machine learning data sets.[7]

Within the natural language processing community, crowdsourcing has been successfully used to generate translations of sentences from one language to another (Ambati et al., 2012; Callison-Burch and Dredze, 2010; Pavlick et al., 2014; Post et al., 2012; Zaidan and Callison-Burch, 2011; Zbib et al., 2012). This approach is especially effective for language pairs for which not much data exists. As one example, Zaidan and Callison-Burch (2011) used crowdsourcing to generate translations of sentences from Urdu to English. They assigned crowdworkers different jobs such as translating a sentence, editing other workers' translations to make them more fluent and grammatical, or ranking the quality of a translation. Finally, they used machine learning methods to predict the highest quality translation based on sentence-level features, worker-level features, and worker-generated ranks. The crowd-generated translations collected by this and other systems can then be used as training data for machine translation tasks.

Within the computer vision community, crowdsourcing is commonly used to collect human-generated labels and annotations for images and video (e.g., Deng et al., 2009; Gebru et al., 2017; Patterson and Hays, 2012; Patterson et al., 2014; Raykar et al., 2010; Russakovsky et al., 2015b; Sorokin and Forsyth, 2008; Su et al., 2012; Vijayanarasimhan and Grauman, 2009; Welinder et al., 2010). For example, the widely used ImageNet database[8] was constructed by leveraging workers on Mechanical Turk to perform tasks such as verifying image annotations and generating bounding boxes (Deng et al., 2009; Russakovsky et al., 2015a; Su et al., 2012). Other data generation and labeling tasks appropriate for crowdsourcing include object classification, attribute (or feature) generation, and image segmentation. A full taxonomy of applications of crowdsourcing to computer visions tasks is beyond the scope of this paper; see instead the comprehensive survey of Kovashka et al. (2016).

3. Evaluating and Debugging Models

Aside from data generation, the most common use of crowdsourcing within the machine learning community is to evaluate or debug models. Crowdsourced evaluation is especially common for unsupervised models, which generally cannot be evaluated in terms of simple metrics like accuracy or precision because there is no objective notion of ground truth. More recently, crowdsourcing has been used to evaluate human-centric properties of supervised models, such as model interpretability. In this section, we review several examples of applications of crowdsourcing to model evaluation and debugging. This list of examples is not intended to be exhaustive, but to give a flavor of different ways in which crowdsourcing can be used in this context.

7. `https://www.google.com/recaptcha`
8. `http://www.image-net.org`

3.1 Evaluating Unsupervised Models

It is increasingly common to see crowdsourcing used to evaluate unsupervised models, such as topic models (e.g., Chang et al., 2009; Hu et al., 2014; Newman et al., 2011; Paul and Dredze, 2014). Topic models are widely used to discover thematic topics from a set of documents such as New York Times articles from the past year or transcripts of Supreme Court hearings (Blei and Lafferty, 2009; Blei et al., 2003; Boyd-Graber et al., 2017). In this context, a topic is a distribution over words in a vocabulary. Every word in the vocabulary occurs in every topic, but with a different probability or weight. For example, a topic model might produce a food topic that places high weight on `cheese`, `kale`, and `bread`, or a politics topic that places high weight on `election`, `senate`, and `bill`. Each document is then represented as a distribution over topics.

Topic models are often used for data exploration and summarization, especially in the social sciences (Boyd-Graber et al., 2017). In order to be useful in these contexts, the inferred topics must be meaningful to end users. For example, if the set of words that appear with high weight in an individual topic are not coherent, the topic will not be useful to end users trying to understand the content of their documents. However, "meaningfulness" is hard to measure analytically, leading many researchers to instead evaluate topic models in terms of easier to quantify criteria, such as predictive power. To address this problem, Chang et al. (2009) proposed using crowdsourcing to measure the quality of a set of topics. The researchers designed a word intrusion task in which a crowdworker is presented with a randomly ordered list of the most common words from a topic. Included in that list is one intruder word that has low weight for the topic but high weight for another topic. The worker is then asked to identify the intruder. If the topic is coherent, then picking out the intruder should be easy (think {`cheese`, `bread`, `steak`, `election`, `mushroom`, `kale`}). If a topic is incoherent, identifying the intruder would be harder. The average error that crowdworkers make on this task can thus be used as a proxy for how coherent topics are. The researchers found that previous measures of success like high log likelihood of held out data do not necessarily imply coherence, illustrating the value of crowd-based evaluation over other techniques.

3.2 Evaluating Model Interpretability

In supervised learning, models are often evaluated in terms of objective performance metrics such as accuracy, precision, or recall. However, even if a model performs well in terms of these criteria, users may hesitate to rely on the model if they do not understand the model's predictions, especially in critical domains like health or criminal justice. Because of this, there is now wide interest in developing models that are human-interpretable (e.g., Doshi-Velez and Kim, 2017; Jung et al., 2017; Koh and Liang, 2017; Lipton, 2016; Lou et al., 2012, 2013; Paul, 2016; Ribeiro et al., 2016; Ustun and Rudin, 2016). Because of the subjective and inherently human-centric nature of interpretability, it is natural to use crowdsourcing to evaluate the interpretability of models.

As one example, Ribeiro et al. (2016) proposed an algorithm, Local Interpretable Model-agnostic Explanations, or LIME, that generates simple, locally faithful explanations for individual predictions made by potentially complex black-box models. They used crowdsourcing to test how how these explanations impact people's ability to perform tasks such as as-

sessing the classifier's quality or determining instances on which the classifier will make a mistake. In one study, the researchers presented crowdworkers with predictions from two SVM classifiers trained on different data sets along with explanations for their predictions. Workers were asked to select the classifier they believed would have better performance. The researchers found that explanations improved workers' ability to choose the best classifier and that LIME produced more effective explanations than a greedy technique. In another study, the researchers used crowdworkers to test which explanations best helped people find flaws in a trained model by identifying features used in the explanation that are irrelevant for the prediction task.

In this example, the researchers compared the effectiveness of specific techniques for generating explanations of predictions. In Section 5.1, we contrast this with crowdsourcing approaches that can be used to gain a better general understanding of how various properties associated with the "interpretability" of a model impact different aspects of human behavior in different scenarios.

3.3 Debugging Components of a Pipeline

In fields like computer vision, speech recognition, translation, and natural language processing, systems often consist of several discrete components linked together to perform a complex task. For example, consider the problem of semantic segmentation, which involves partitioning an image into semantically meaningful parts and labeling each part with a class. There are promising approaches to this problem that use machine learning models such as conditional random fields (CRFs) to integrate feedback from independent components that perform various scene understanding tasks like object detection, scene recognition, and segmentation. If a system designer wants to improve performance, it is not always clear which component to focus attention on.

To solve this problem, Parikh and Zitnick (2011) proposed the idea of "human debugging," in which humans are used to uncover bottlenecks in AI systems. The goal of human debugging is to identify which component in a system is the "weakest link." The basic idea is simple. To quantify how much potential improvements to a particular component would benefit the system as a whole, we could imagine replacing the component with something (close to) perfectly accurate and testing how much the system improves. Since for many vision and language tasks human performance is an upper bound on what we might expect from a machine, we can instead replace the component with a human.

Mottaghi et al. (2013, 2016) applied this idea in order to analyze the performance of a CRF that has been used in the computer vision community for scene understanding (Yao et al., 2012). They replaced each component with crowdworkers from Mechanical Turk and measured the change in performance of both the component in isolation and the system as a whole. One of their most interesting findings was that humans are actually less accurate than machines at one particular task (classifying super-pixels), yet when human classifications were plugged into the CRF, the system performance improved. One interpretation of this result is that perhaps making fewer mistakes classifying super-pixels is not enough. Rather it may be more important that the classifier makes the right kind of mistakes—the kind made by humans. This kind of feedback helps designers know where to focus their effort.

Recently, Nushi et al. (2017) took this idea one step further, allowing crowdworkers to propose targeted fixes to the machine components of a larger system and then evaluating the effect of various component fixes on the overall system performance.

4. Hybrid Intelligence Systems

Despite the current hype around AI and the great technological advances that have been made in recent years, AI systems are still far from perfect. In some cases, AI systems can benefit by involving humans in the loop to perform tasks that rely on life experience, judgment, or domain knowledge (Kamar, 2016). Such hybrid intelligence systems can leverage the complementary strengths of humans and machines to accomplish more than would be possible using humans or machines alone.

Hybrid systems have been designed to perform tasks from grading students' work (Kulkarni et al., 2014; Wright et al., 2015) to writing essays or novels (Bernstein et al., 2010; Kim et al., 2017; Kittur et al., 2011; Salehi et al., 2017; Teevan et al., 2016) to building better topic models (Hu et al., 2014). In this section, we walk through four illustrative examples: hybrid systems for clustering data points (Gomes et al., 2011; Heikinheimo and Ukkonen, 2013; Tamuz et al., 2011), transcribing speech in real time (Kushalnagar et al., 2012; Lasecki and Bigham, 2012; Lasecki et al., 2012, 2013, 2017; Naim et al., 2013), scheduling conference sessions (André et al., 2013; Bhardwaj et al., 2014; Chilton et al., 2014; Kim et al., 2013), and forecasting geopolitical or economic events (Atanasov et al., 2017; Baron et al., 2014; Mellers et al., 2015b).

4.1 Hybrid Clustering

Hybrid intelligence systems can be put to use to solve traditional machine learning problems like clustering in scenarios in which data points are easier for humans to understand and categorize than they are for machines. For example, given a data set of celebrity images, a human could potentially use their life experience and background knowledge to categorize them into sets like "actors" or "politicians," while a machine without access to this knowledge and experience could not.

Many hybrid clustering techniques have been proposed (Davidson et al., 2014; Gomes et al., 2011; Karaletsos et al., 2016; Mazumdar and Saha, 2016; Tamuz et al., 2011; Vesdapunt et al., 2014; Vinayak and Hassibi, 2016; Vinayak et al., 2014; Wah et al., 2014; Wang et al., 2012b; Wauthier et al., 2012; Yi et al., 2012a,b), the majority of which solicit human judgments or comparisons in order to actively generate a similarity matrix or other similarity function. Approaches vary in terms of the types of queries given to human judges, the algorithms used to aggregate their responses, and whether or not additional features of each object are available to the algorithm. Some researchers focus primarily on the entity resolution setting, in which the goal is to cluster together objects that refer to the same entity (e.g., Marcus et al., 2011; Mazumdar and Saha, 2016; Vesdapunt et al., 2014; Wang et al., 2012b), while others consider clustering more broadly.

Tamuz et al. (2011) designed an adaptive algorithm that estimates a similarity matrix from human judgments based on comparisons of triples ("*Is object A more similar to object B or object C?*"). Their approach requires only a relatively small number of human judgments to obtain a good approximation. Using this approach, they were able to answer questions

like which necktie would be a good substitute for another, a task that would perhaps be difficult for a machine without specialized human knowledge. As a complement to this algorithmic work, Wilber et al. (2014) outlined and studied several user interface techniques that allow high quality comparisons of triples to be collected faster.

Gomes et al. (2011) proposed a crowd-clustering approach in which each member of a crowd is presented a relatively small set of objects and asked to cluster just these objects. The sets of objects presented to different workers are distinct but overlap. The partial clusterings generated by the workers are then aggregated into one full clustering of all objects using an algorithm based on Bayesian inference.

Heikinheimo and Ukkonen (2013) took a different approach to hybrid clustering, proposing a crowd-powered version of the k-means algorithm. Running the standard k-means algorithm (Lloyd, 1982) requires the ability to perform two main operations:

1. Given a set S of objects and one new object x, find the object in S that is closest to x.
2. Find the center of a set of objects S, that is, the object in S with the lowest sum of distances to other objects in S.

Heikinheimo and Ukkonen (2013) showed how to perform each of these operations with a crowd. The first operation is straight-forward, assuming the size of the set S (which, in this case, is the number of clusters k since S is the set of cluster representatives) is sufficiently small: they simply present a worker with S and x and ask the worker which object in S is closest to x. (Noisy responses from multiple workers can be combined using standard techniques.) To perform the second operation, the researchers proposed the following simple algorithm, which they call `crowd-median`. Each crowdworker is presented with a set of three objects from the set S and asked which object is the outlier. After many such triples have been presented to workers, the object chosen least often as the outlier is selected as the center. The researchers gave both theoretical and empirical evidence that the output of `crowd-median` coincides with existing definitions of a centroid, and empirical evidence that the resulting crowd-powered k-means algorithms produces coherent clusters.

4.2 Hybrid Speech Recognition

Quickly and reliably converting speech to text requires a level of contextual understanding beyond the capabilities of machines and a level of speed beyond the capabilities of most humans. Closed captioning systems that rely on automatic speech recognition work very well under ideal circumstances (for example, when the voice recording is high quality and the system has been trained on data from the particular speaker), but can fail when presented with low quality audio, speakers with novel accents, or language with technical jargon that falls outside the vocabulary on which the system was trained. In these scenarios, professional stenographers produce the best results, but high quality stenographers can be prohibitively expensive and are not available on demand.

With this in mind, Lasecki et al. developed Scribe (Kushalnagar et al., 2012; Lasecki and Bigham, 2012; Lasecki et al., 2012, 2013, 2017; Naim et al., 2013), a hybrid speech recognition system that provides relatively inexpensive, real-time, and on-demand closed captioning to deaf or hard of hearing users to help them understand lectures, meetings, or other day-to-day conversations. As illustrated in Figure 1, Scribe combines algorithmic

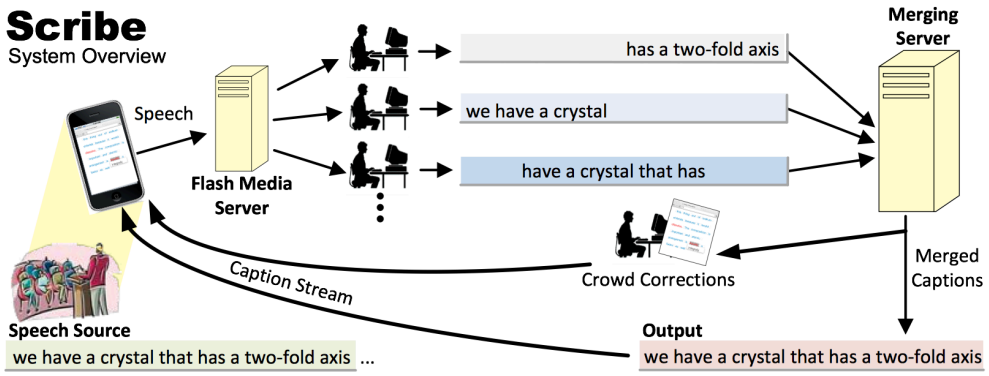


Figure 1: The architecture of Scribe. Image originally appeared in Lasecki et al. (2012).

techniques with the power of the crowd. As soon as a user starts recording, the recorded audio is sent simultaneously to several crowdworkers. These workers are not expected to fully transcribe the speech, which is generally not possible without a specialized stenotype keyboard. Instead, each worker transcribes sentence fragments. Scribe adjusts the speed and volume of the speech adaptively for each worker in order to focus workers' attention on distinct, overlapping components. It then uses multiple sequence alignment techniques (Edgar and Batzoglou, 2006; Lermen and Reinert, 2000; Naim et al., 2013) to combine the workers' text into one complete and coherent stream that is delivered back to the user with a delay of only a few seconds.

More recently, Gaur et al. (2015, 2016) developed an approach in which a single crowdworker is used to correct mistakes made by an automatic speech recognition system in real time. The crowdworker views the transcription output by a speech recognition system while listening to the corresponding audio. She then types out corrections to mistakes in the transcription as she notices them. These corrections are automatically incorporated into the appropriate spot in the transcription. Initial experiments showed that when this system was run with crowdworkers from Mechanical Turk, the word error rate improved. However, only 30% of possible corrections were made, leaving significant room for improvement in future work (Gaur et al., 2015).

As new breakthroughs continue to improve automatic speech recognition (Yu and Deng, 2014), this hybrid approach may eventually become unnecessary. However, the ability for speech recognition systems to achieve true human parity under nonideal conditions likely remains far in the future (Bigham, 2017). This example shows that crowdsourcing can be an effective way of compensating for a lack of sufficient machine learning or AI solutions until a time when the technology improves.

4.3 Hybrid Scheduling

Several researchers have explored the potential use of hybrid intelligence systems to solve complex tasks with global constraints or consistency requirements. Examples of such tasks include itinerary planning (Zhang et al., 2012), taxonomy creation (Bragg et al., 2013;

Chilton et al., 2013), and writing (Bernstein et al., 2010; Kim et al., 2017; Kittur et al., 2011; Salehi et al., 2017; Teevan et al., 2016). In this section we describe Cobi[9] (André et al., 2013; Bhardwaj et al., 2014; Chilton et al., 2014; Kim et al., 2013), a hybrid conference scheduling system. Rather than enlisting the help of anonymous crowdworkers, Cobi is based on the idea of "communitysourcing." It draws on the specialized expertise of people within a research community.

The problem of scheduling conference sessions can be viewed as a constrained optimization problem in which the solver has no direct access to the constraints. The goal of conference organizers is to group similar talks together in sessions while minimizing conflicts between talks that are scheduled at the same time, but conference organizers generally do not know which sets of talks attendees want to see. This optimization problem can be large; for example, Cobi was used to map out the schedule of the Conference on Human Factors in Computing Systems (CHI) in 2013, which required scheduling 400 talks in 16 parallel tracks.

Cobi was designed to efficiently collect information about attendee preferences from the community and use this information to optimize the conference schedule. (It is worth noting that the conference chairs always retain control and can choose to overwrite the optimized schedule.) As one example, their "authorsourcing" component presents authors with papers that are potentially similar to their own and asks which ones would be a good fit to appear in the same session. In order to generate the lists of potentially similar papers in the first place, their "committeesourcing" component makes use of hybrid clustering techniques like those discussed in Section 4.1 (André et al., 2013).

Communitysourcing can be especially effective because the same people who are asked to provide information also benefit from high quality end results. Although participants were not directly compensated, when Cobi was first deployed at CHI, the authors of 87% of accepted submissions opted to engage with the system.

Cobi was subsequently deployed several times for scheduling at both CHI and the ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW). Anecdotally, users of the system found that the constraints generated by the authorsourcing component were crucial for producing a schedule of coherent and nonconflicting sessions; without this input the scheduler performed poorly.

4.4 Hybrid Forecasting

Significant resources are devoted to producing forecasts about geopolitical events and economic indicators. Humans are flexible in their ability to reason about arbitrary events, but human forecasts can be limited by cognitive biases or the inability to digest and process information at scale. Statistical and data-driven models, on the other hand, are able to take advantage of vast quantities of available data, but are difficult to design and train for one-of-a-kind events. Hybrid forecasting systems aim to combine the computational power of machines with the flexibility of humans to produce accurate forecasts.

In the most basic hybrid forecasting systems, algorithmic techniques are used primarily as a way to elicit and aggregate human-generated forecasts. One common example is a prediction market (Berg et al., 2008; Wolfers and Zitzewitz, 2004), a financial market in which traders can buy or sell securities with payoffs that are linked to future events. For

9. `http://projectcobi.com`

example, in an election market, traders might buy or sell a security that is worth \$1 if the incumbent candidate wins and nothing otherwise. If a trader believes that the probability of this candidate winning is p and wants to maximize her expected payoff, then she should be willing to buy this security at any price less than \$$p$, since with probability p she would get \$1. Similarly, she should be willing to sell at any price greater than \$$p$. For this reason, we can think of the current market price of this security as capturing traders' collective beliefs about how likely it is that the incumbent will win. Prediction markets can be operated as continuous double auctions, much like the stock market, requiring very little algorithmic ingenuity. However, when the level of trade is low, there can be advantages to operating prediction markets using algorithmic market makers that automatically set prices based on the history of trade (Chen and Pennock, 2007; Hanson, 2003). Prediction markets have recently gained more attention in the machine learning community due to the discovery of strong mathematical connections between these algorithmic market makers and no-regret online learning algorithms (Abernethy et al., 2013; Chen and Vaughan, 2010).

As part of the Good Judgment Project (Schoemaker and Tetlock, 2016; Tetlock et al., 2017; Ungar et al., 2012), a large-scale project funded by U.S. Intelligence, researchers evaluated and compared a wide range of algorithmic techniques for eliciting and aggregating human forecasts, including prediction markets and prediction polls, in which forecasters are asked to directly provide a probability estimate about the likelihood of an event. Through a series of randomized controlled trials, they found that prediction markets produce more accurate forecasts than those obtained by simply averaging forecasts from prediction polls. However, even higher accuracy could be obtained by aggregating prediction poll forecasts using more clever statistical methods (Atanasov et al., 2017) and extremizing aggregated forecasts (Baron et al., 2014). They also studied the importance of identifying and teaming up the top-performing individual forecasters (Mellers et al., 2015b) as well as the psychological traits shared by these top forecasters (Mellers et al., 2015a).

Building on lessons learned from the Good Judgment Project, the U.S. Office of the Director of National Intelligence has recently launched a new program aimed at producing hybrid forecasting systems that more comprehensively integrate modern data-driven approaches with human forecasting capabilities.[10]

4.5 Hybrid Intelligence Systems in Industry

So far we have focused on hybrid intelligence systems that have come out of the research community, but it is worth mentioning that human-in-the-loop systems are widely used in industry as well. To name just a few examples, Stitch Fix, which provides personalized style recommendations, trains machine learning algorithms to suggest items that a user might like and then sends the output of these algorithms to a human expert to curate and pare down.[11] Twitter employs contract workers to interpret search terms that suddenly spike in meaning, often due to recent mentions in the media or pop culture that an algorithm trained on stale data would miss.[12] PatternEx uses machine learning to identify suspicious activity that could indicate a security threat. This suspicious activity is then examined by a

10. https://www.iarpa.gov/index.php/research-programs/hfc
11. http://multithreaded.stitchfix.com/blog/2016/03/29/hcomp1/
12. http://nyti.ms/2gzOo4K

human, who determines whether there is a real attack, and the human's feedback is used to improve the system.[13] And even search engines like Google and Bing can be viewed as hybrid intelligence systems, since the relevance judgments that are used to adaptively improve the systems are generated by humans (Alonso, 2013; Alonso et al., 2008; Kazai, 2011).

5. Behavioral Studies to Inform Machine Learning Research

Within the past few years, there has been an increased interest, both from within and outside the machine learning community, in understanding how humans interact with machine learning systems. Researchers are striving to make machine learning models human-interpretable (Doshi-Velez and Kim, 2017; Jung et al., 2017; Koh and Liang, 2017; Lipton, 2016; Lou et al., 2012, 2013; Paul, 2016; Ribeiro et al., 2016; Ustun and Rudin, 2016), to make machine learning tools easier to use (Brooks et al., 2015; Patel et al., 2010; Simard et al., 2017), and to understand how algorithmic decisions impact people's lives (Angwin et al., 2016; Barocas and Selbst, 2016; Campolo et al., 2017; Chouldechova, 2017; Corbett-Davies et al., 2017; Sweeney, 2013).

During the same time period, psychologists and social scientists have increasingly turned to crowdsourcing platforms to run behavioral experiments that traditionally would have been conducted on undergraduates in a physical lab. One of the papers frequently credited with introducing Mechanical Turk to the psychology community, Buhrmester et al.'s "Amazon's Mechanical Turk: A New Source of Inexpensive, Yet High-Quality, Data?" (Buhrmester et al., 2011), has been cited almost 5500 times according to Google Scholar, and this number continues to rise. Crowdsourcing platforms provide fast and easy access to large pools of subjects, and promote faster iteration between the development of new theories and experimentation, allowing researchers to speed up their overall research process (Mason and Suri, 2012). Additionally, studies have shown that classic psychology and behavioral economics results can be replicated using crowdworkers (Horton et al., 2011; Paolacci et al., 2010; Simons and Chabris, 2012; Suri and Watts, 2011). On the down side, there are some concerns about overuse of subjects (Chandler et al., 2014; Peera et al., 2017), with the same crowdworkers participating in many variants of the same standard psychology experiments, with which they have become familiar.

These developments open up the opportunity for interdisciplinary research that uses behavioral experiments conducted on crowdsourcing platforms to improve our understanding of how humans interact with machine learning and AI systems. This line of work goes beyond the idea of using crowdsourcing to evaluate one particular machine learning model or user interface, as discussed in Section 3, and instead seeks a general understanding of the components of human behavior that could inform the use and design of machine learning systems more broadly. Such experiments can help us develop better models of human behavior that could be used, in turn, to develop better algorithms and interfaces (Chen et al., 2016).

It is worth noting that behavioral experiments and other crowdsourced user studies could benefit other subfields of computer science as well, and there are already examples where it has. For example, Mechanical Turk experiments have been used to study lay people's perceptions of password security (Ur et al., 2016) and perceptions of graphics and visualizations (Heer and Bostock, 2010).

13. `http://tek.io/1Vy01KB`

Compared with the bodies of work on crowdsourcing for data generation, model evaluation, and hybrid intelligence, there is relatively little research in this area to date. In this section, we describe a few examples of behavioral studies using crowdsourcing platforms that have the potential to change the way we think about applications of machine learning in the hope that these examples will inspire additional research.

5.1 Understanding Trust in Predictive Models

There is a large body of work cutting across psychology, management, and other research communities that studies human trust in algorithmic predictions and models (Dietvorst et al., 2015, 2016; Dzindolet et al., 2002; Logg, 2017; Promberger and Baron, 2006; Sinha and Swearingen, 2001; Yeomans et al., 2017). While the earlier work was performed in traditional labs, the newer studies primarily rely on crowdworkers or a mix of crowdworkers and in-person participants (Dietvorst et al., 2015, 2016; Logg, 2017; Yeomans et al., 2017). This body of work provides invaluable insights about how real end users interact with machine learning systems in practice that can be put to use immediately in the design of models and user interfaces.

As one example, there is significant evidence in the literature of algorithm aversion, a phenomenon in which people fail to trust an algorithm once they have seen the algorithm make a mistake, even if the algorithm outperforms human predictors (Dietvorst et al., 2015; Dzindolet et al., 2002). This is worrisome since essentially all machine learning models make errors at least occasionally. Dietvorst et al. (2016) ran a sequence of experiments, some on crowdworkers and some on students, to examine the question of whether algorithm aversion can be overcome using simple interventions. In particular, the researchers asked whether people would choose to use an algorithm more if they were given the ability to intervene and make minor adjustments to the algorithm's prediction when they believed it to be wrong.

In one study, they asked crowdworkers to predict students' test scores on a standardized test using nine features such as the student's favorite subject in school and the region of the country the student lives in. Workers would be paid based on the accuracy of their predictions. The workers were told that analysts had trained a model to make the same predictions and were given the average error of the model (and thus explicitly told that the model was not perfect). They were then asked to decide whether they wanted to use the model or not. The workers were assigned to one of four randomized conditions. In the first, workers were told that if they chose to use the model they would not be allowed to adjust the predictions of the model at all. In the other three, workers were told that they would be given the ability to adjust each prediction of the model by up to 2 points, 5 points, or 10 points respectively. The researchers found that participants were indeed significantly more likely to choose to use the model if they were given the ability to intervene and adjust the model's prediction. In both the adjust-by-5 and adjust-by-10 conditions, 71% of participants chose to use the model, along with 68% in the adjust-by-2 condition. On the other hand, those who would not have the opportunity to intervene only chose to use the model 47% of the time.

Participants in the conditions in which adjustments were allowed also had significantly better predictive accuracy than those who were not allowed to adjust the model. This is not because their adjustments helped; on the contrary, participants would have done better

by always following the model exactly. They were more accurate because they were more willing to rely on the model.

This finding has immediate implications about which strategies might help gain user trust in machine learning models. In particular, users might be more willing to trust a model if they have the ability to intervene. Even if this human intervention leads to a worse prediction, allowing the intervention may still be beneficial because the user will be more likely to use the model in the first place.

It is natural to imagine that similar ideas could be applied to study model interpretability. As one very recent step in this direction, Poursabzi-Sangdeh et al. (2018) ran a large-scale human-subject experiment on Mechanical Turk that was designed to test how two different attributes of a model (the number of features and whether the details of the model are presented to the user or the model is presented as a black box) impact either users' understanding of the model or users' trust in the model. Unlike the experiments described in Section 3.2, in which crowdworkers were used to evaluate the interpretability of one specific model, Poursabzi-Sangdeh et al. (2018) tried to uncover how different factors of a model influence different aspects of interpretability more broadly, with the goal of providing general guidance on how to develop future interpretable models.

5.2 Understanding Reactions to Ads

Internet advertising is a huge business. Revenues from internet advertising in the U.S. alone hit $72.5 billion in 2016, up 22% from the previous year.[14] Naturally, significant effort is devoted to developing and optimizing machine learning algorithms for online advertising, especially in industry. New algorithms are generally put through rigorous A-B testing before deployment to quantify and measure their impact when run on real users. However, one might ask whether it is possible to design better algorithms using general insights about human behavior that transcend any one particular algorithm.

Goldstein et al. (2013; 2014) set out to quantify the impact of "annoying" display ads on user behavior, under the hypothesis that web publishers might be losing money and driving away users by displaying annoying ads. They designed a two-stage experiment run on Mechanical Turk. The first stage was designed to identify examples of "good" and "bad" ads. To do this, they presented crowdworkers with overlapping sets of display ads and asked for judgments of how annoying the ads were, a standard data labeling task as discussed in Section 2.1. By aggregating these judgments across workers, they produced lists of the most and least annoying banner ads.

In the second stage, Goldstein et al. (2013; 2014) ran a behavioral experiment aimed at estimating how much monetary value web users get from not being subjected to annoying ads. They posted another Mechanical Turk task in which workers were asked to label emails from the Enron email database as either spam or not spam. Workers were randomly assigned to different experimental conditions. Some workers saw good ads (as determined in stage 1) next to each email, while others saw bad, annoying ads. A third group saw no ads at all. The researchers also randomly varied how much users were paid for labeling each email. Workers were free to label as many emails as they wanted. By comparing the number of emails that workers chose to classify in each experimental condition, the

14. `https://read.bi/2Ks0StR`

researchers produced an estimate of the amount of extra money it would be necessary to pay workers who were exposed to bad ads in order to get them to perform the same number of email classification tasks as those exposed to good ads or no ads at all. From that, they were able to measure people's annoyance at bad ads.

They found that people were willing to classify almost as many emails when shown good ads compared with no ads at all, suggesting that displaying good ads does not hurt publishers. The same was not true for annoying ads. The researchers estimated that they would have to pay approximately $1 extra to generate 1000 views using a bad ad compared with a good ad or no ad. This is a significant amount as a typical banner ad might cost $1-$5 per 1000 views. In other words, publishers may very well be losing money by displaying annoying ads unless they charge significantly more per view.

While there are some limitations to the applicability of these results (for example, it is plausible that people react differently to ads when performing classification tasks than they would when browsing the news), it is a valuable step towards a model of user reactions to annoying ads. It is easy to imagine that such a model would prove useful when designing machine learning algorithms for pricing and displaying banner ads.

6. Understanding the Crowd

In the previous section, we argued that crowdsourced studies of human behavior can be valuable for understanding how lay people interact with machine learning systems. In this section, we argue that such studies are also useful for understanding the behavior of the crowd itself. This understanding helps us better model the crowd and allows us to define concrete recommendations of best practices that can be put to use whether using the crowd for data generation, model evaluation, hybrid intelligence systems, behavioral research, or any other purpose.

The studies described in this section help us understand how real crowdworkers respond to incentives, yielding immediately applicable guidelines for setting payments in addition to more accurate ways of modeling crowd behavior in theoretical work on incentive design. They tell us how to most effectively gamify crowdwork and provide other sources of intrinsic motivation for workers. They help us get a grip on the question of how widespread dishonesty is on crowdsourcing platforms, and how dishonest behavior can be mitigated. And they show us that crowdworkers are not independent and isolated workers, but have a rich social network.

6.1 Crowdworker Demographics

Over the years there have been several studies published that examine the demographics of workers on Mechanical Turk. We mention only a few key statistics that help paint a picture of the worker pool. These are based on a November 2016 snapshot from MTurk Tracker,[15] a project aimed at tracking the demographics of Mechanical Turk over time by continually releasing tasks containing demographic surveys on Mechanical Turk to obtain up-to-date information about workers (Difallah et al., 2015).

15. http://www.behind-the-enemy-lines.com/2015/04/demographics-of-mechanical-turk-now.html

These statistics should be taken with a grain of salt for several reasons. First, worker demographics change over time. For example, there is evidence that the worker pool on Mechanical Turk is shifting to be more heavily composed of American workers due to changes in Amazon's rules and regulations (Silberman et al., 2015). Second, not all workers on Mechanical Turk choose to work on surveys, so these statistics perhaps better reflect the population of workers who do survey work. That said, these demographics are more or less in line with those reported in other contemporaneous studies.

According to the MTurk Tracker data:

- About 70-80% of Mechanical Turk workers are from the United States, while about 10-20% are from India, but the breakdown of workers varies significantly throughout the day. The prevalence of workers from the U.S. and India makes sense because Mechanical Turk offers payment only in U.S. dollars, Indian rupees, or Amazon credit.
- The breakdown between male and female workers is fairly close to even, though it varies a bit by country.
- For crowdworkers in the U.S., the (self-reported) median household income is in the range of $40K-$60K, which is in line with the median U.S. household income. The median for Indian workers is less than $15K, with many Indian workers reporting a household income of less than $10K per year.

There is evidence that other crowdsourcing platforms, such as CrowdFlower and Prolific Academic, attract more European workers and lower income workers than Mechanical Turk (Peera et al., 2017).

Goodman and Paolacci (2017) provide a nice overview of the similarities and differences between the population of Mechanical Turk workers and the U.S. population as a whole, as well as the populations traditionally used for consumer studies.

The demographics of available workers vary widely based on the time of day and, to a lesser extent, day of the week. Through a large study of intertemporal demographic differences on Mechanical Turk, Casey et al. (2017) found that, even restricting attention to workers from the U.S., the demographics of available workers change dramatically over the course of a day. For example, they found that workers who completed their task at night were more likely to be single than those who completed it in the morning, and more likely to be completing the task on a smartphone.

6.2 Dishonesty Among Workers

Researchers may be deterred from using crowdsourcing due to concerns about the prevalence of dishonest workers. In this section, we discuss the results of behavioral experiments aimed at quantifying the presence of dishonest behavior on Mechanical Turk.

Suri et al. (2011) borrowed a trick from a lab study run by Fischbacher and Föllmi-Heusi (2013) that allowed them to measure how trustworthy crowdworkers are on the whole without the ability to detect individual lies. They posted a task on Mechanical Turk in which each worker was asked to roll a die (or simulate rolling a die on an external website) and report the value of her roll, a random number between 1 and 6. For completing this task, the worker received a base payment of $0.25 plus an additional bonus of $0.25 times her reported roll. For example, if a worker reported rolling a 4, she would receive a total

payment of $1.25. Thus the workers, who knew that there was no way for the outcomes of their rolls to be verified, were given a direct incentive to lie.

If all workers honestly reported the outcome of a die roll, the mean of the reported rolls of the 175 participants in the study would be close to 3.5. The researchers instead observed a mean reported roll of 3.91. On the whole, workers had a tendency to overreport rolls of 5 and 6 and underreport rolls of 1 and 2. In fact, it is possible that roughly a third of participants who rolled a 1 or 2 chose to misreport. (It is also possible that some participants ignored the instructions to roll a die at all, choosing to report an arbitrary number.)

Suri et al. (2011) next examined whether increasing the ability of the task requester to detect cheating would lead to higher levels of honesty. They conducted a variant of the die rolling experiment in which each worker was asked to report the results of a sequence of thirty die rolls instead of just one. Workers in this experiment received a base payment of $0.25 plus the sum of their reported die rolls, for a total payment between $0.55 and $2.05. This time, the mean reported role of the 232 participants was 3.57—still statistically significantly higher than the expected mean of 3.5, but practically much closer. Only three of the 232 experiment participants were found to have an average report significantly greater than what would be expected if they reported their rolls honestly, and only one reported the payoff-maximizing sequence of thirty rolls of 6. This suggests that dishonest behavior may be less prevalent when there is a chance that this behavior will be detected (though the results of the two experiments are not directly comparable since they were performed at different times on potentially different populations).

Two recent contemporaneous studies (Chandler and Paolacci, 2017; Wessling et al., 2017) examined whether crowdworkers misrepresent themselves in order to satisfy the eligibility criteria of tasks. Chandler and Paolacci (2017) asked over two thousand crowdworkers to report whether they are the parent or guardian of a child with autism. Workers were randomly assigned to one of two experimental conditions. In one, workers were told that this question was part of a prescreening test for a subsequent study, while in another (the control), they were not. 7.8% of workers in the prescreening condition reported being the parent or guardian of a child with autism, compared with 4.3% in the control, suggesting that at least 3.5% of workers in the prescreening condition provided a fraudulent response. While 3.5% of workers may seem small, note that due to the low number of workers who would truly qualify for a follow-up study based on this question, this level of dishonesty would lead to a high prevalence of impostors (45%) in a follow-up study. In a separate study, the same researchers found that when payments were sufficiently high, 16% of participants made a second attempt to pass a prescreening survey, identifying themselves as a different gender after initially being blocked. Similarly, Wessling et al. (2017) provided additional evidence of misrepresentation in prescreening tests.

Based on these findings, both papers include detailed discussions of best practices that can be followed to mitigate dishonesty in prescreening tests. One recommendation is to collect screening data as part of a stand-alone task ahead of time and call back eligible participants later. As an alternative, when costs are not prohibitive, all workers can be allowed to perform a task and those who do not meet the screening criteria can be filtered out after all data have been collected.

All of this work suggests that the rate of dishonest behavior and spam depends on the particular task and the motivation to lie. There is some anecdotal evidence that spammers

are especially drawn to surveys (Mason and Suri, 2012) and multiple-choice questions (Rao and Michel, 2016) since these are especially easy tasks to complete, so rates of spam may be even higher on these questions.

It is worth noting that requesters on crowdsourcing platforms can be dishonest or otherwise malicious too. For example, requesters may ask workers to create social media accounts to post specific content or engage in other dodgy Internet marketing practices. It fact, there are now entire crowdsourcing platforms in China specifically devoted to these "crowdturfing" tasks (Wang et al., 2012a).

6.3 Monetary Incentives

One of the first questions that many researchers have when they decide to incorporate crowdsourcing into their work is how much to pay per task. When crowdsourcing first began to gain popularity among researchers, part of the appeal was the ability to generate data or run experiments cheaply. For example, Snow et al. (2008) boasted that they were able to offer workers $0.02 to complete a set of 30 annotations, obtaining 1500 annotations per dollar. However, over the last decade, the views of the community have shifted as the ethics of crowdsourcing have received more attention (Kittur et al., 2013; Salehi et al., 2015; Silberman et al., 2010; Williamson, 2016). Although crowdworkers are legally considered contractors, making minimum wage laws inapplicable, guidelines put forth by the Dynamo project[16] (Salehi et al., 2015) recommend paying the equivalent of the current U.S. federal minimum wage or more. An effective and widely used method of setting the payment for a task is to first estimate the time it takes to complete the task (for example, by asking colleagues or students to try out the task, or by posting a small test batch of tasks) and then use that estimate to ensure that the per-hour payment is higher than U.S. minimum wage.

A natural question is whether paying higher wages increases the quality of work. There is evidence that, at least in some scenarios, the answer is no. Several behavioral studies examining the impact of payments on quality found that setting higher payments increased the quantity of work that crowdworkers were willing to do, but not the quality (Buhrmester et al., 2011; Litman et al., 2014; Mason and Watts, 2009; Rogstadius et al., 2011). Mason and Watts (2009) conjectured that this might be due, at least in part, to an anchoring effect: workers' opinions about fair payment rates are anchored by the payments they are offered. In one of their experiments, workers were asked to sort sets of images and were randomly assigned to experimental conditions in which they received $0.01 per task, $0.05 per task, or $0.10 per task respectively. (All tasks were advertised at a rate of $0.01 and additional payments were made via bonuses to avoid selection effects.) In a post-hoc survey, workers who were paid $0.01 felt they should have received $0.05 for the task on average, while workers who were paid $0.05 felt they should have received $0.08, and workers who were paid $0.10 felt they should have received $0.13.

On the other hand, two recent studies have shown that in other scenarios, higher payments do increase work quality (Ho et al., 2015; Ye et al., 2017). There are several possible reasons for this discrepancy. One is that the type of task being performed might impact the effectiveness of increased payments. Ho et al. (2015) and Ye et al. (2017) both used "effort-responsive tasks"—tasks for which workers are able to improve their output by spending

16. http://wiki.wearedynamo.org/index.php/Guidelines_for_Academic_Requesters

more time or effort—in their studies, whereas the study mentioned above used a task that was fairly easy to complete without much effort. Mason and Watts (2009) did use an effort-responsive task (solving word puzzles) in a second study, but found that for this particular task, worker quality was more correlated with enjoyment of word puzzles than the rate of pay offered, an argument that intrinsic motivation (the topic of Section 6.4) can overpower monetary incentives. Another possibility is that in some studies, payments were so low that the differences between conditions were not salient. Buhrmester et al. (2011) offered either \$0.02, \$0.10, or \$0.50 for up to 30 minutes of work, while Ho et al. (2015) and Ye et al. (2017) aimed to pay at least minimum wage and had larger per-hour gaps between payments in different experimental conditions.

While this research is somewhat inconclusive, it does appear that paying higher wages consistently increases the number of tasks that crowdworkers are willing to perform and therefore speeds up the rate at which a requester's tasks are completed.

Another body of work is aimed at answering the question of whether the quality of crowdwork can be improved through the use of performance-based payments, payment schemes that explicitly reward crowdworkers for higher quality work (Harris, 2011; Ho et al., 2015; Shaw et al., 2011; Yin et al., 2013, 2014). While in theory, such payment schemes could be arbitrarily complex (Ho et al., 2016), in practice they are generally implemented in the form of a bonus payment awarded for exceeding a particular quality threshold. Bonus payments of this form are common on Mechanical Turk.

Here, too, results have been mixed. Harris (2011) asked workers to evaluate the relevance of resumes and found that performance-based payments increased both quality and the amount of time that workers spent on the task. On the other hand, Shaw et al. (2011) compared fourteen incentives schemes, including four that involved performance-based payments, and did not find significant increases in quality, and Yin et al. (2013) varied the bonus sizes offered to workers and found no significant difference in quality between experimental conditions.

The most comprehensive experimental study of performance-based payments was performed by Ho et al. (2015). They showed that performance-based payments can improve quality for particular tasks (again, those that are effort-responsive). They found that the effectiveness of performance-based payments is not heavily dependent on the precise quality threshold chosen. They also tested how sensitive their results were to the size of the bonus offered and found that as long as the bonus offered was big enough, quality improved. Offering a very small bonus (say, \$0.05 on a \$0.50 base payment) actually led to a small apparent decrease in performance, though this decrease is not statistically significant. This may explain the negative results of Shaw et al. (2011), since their bonus payments were very small (\$0.03 on a base payment of \$0.30). It may also explain the results of Yin et al. (2013), who considered only bonuses that were relatively large compared with the base, a regime in which Ho et al. (2015) also saw no statistically significant differences in quality when varying the bonus size.

It is not always immediately apparent whether or not a task is effort-responsive. As an example, Ho et al. (2015) were surprised to find that handwriting recognition is not. When they gave workers a handwriting recognition task, the majority of workers were able to identify most words with little effort. When workers could not make out a word, additional time spent did not help. The researchers suggest that to determine whether or not a task

is effort-responsive, a requester should run a pilot experiment in which they ask workers to complete the task for a fixed payment and examine the relationship between time spent and quality of results. The results of such a pilot could be used to determine an appropriate payment scheme for the task.

6.4 Intrinsic Motivation

In addition to monetary incentives, researchers have also explored the ways in which intrinsic sources of motivation, such as having fun or doing meaningful work, affect the quantity and quality of work that crowdworkers perform. This research is important for informing the design of volunteer-based or citizen science platforms, like the Zooniverse[17] and Science at Home[18], as well as paid crowdsourcing platforms like Mechanical Turk.

In one study, Chandler and Kapelner (2013) found that crowdworkers are more active when the tasks they are asked to perform are framed as meaningful. The researchers recruited workers on Mechanical Turk to label medical images. In one experimental condition, workers were told that they were labeling tumor cells and that the results of their work would be used to assist medical researchers. In the control condition, they were given no context for the task at all. In a third condition, they were given no context and additionally told that the labels they generated would not be recorded; that is, all of their work would be discarded. The researchers found that when workers were told their work would benefit medical research, the quantity of work that they produced increased compared with the control, but their work was not significantly more accurate. On the other hand, when workers were told their work would be discarded, the quality of their work was worse than the control, but the quantity of work produced was similar. Similar effects were observed by Rogstadius et al. (2011), who compared the behavior of workers who were told they were performing work for a nonprofit organization "dedicated to saving lives by improving health throughout the world" with workers who were told they were working for a for-profit pharmaceutical manufacturer. This work suggests that when the task being performed has the potential to do good in the world, it is worth emphasizing this to the workers.

Gamification is an another effective way of increasing crowdworkers' motivation that can be applied in both paid and unpaid crowdsourcing settings (Feyisetan et al., 2015; Lee et al., 2013; von Ahn and Dabbish, 2008). Von Ahn and Dabbish (2008) pioneered the study of "games with a purpose," computer games in which players, as a side effect of their play, accomplish tasks that are difficult for machines, such as generating training data for machine learning algorithms. For example, in the ESP Game (von Ahn and Dabbish, 2004) (now the Google Image Labeler), randomly matched pairs of players are presented with an image and asked to type in words that describe it (i.e., labels), as in Figure 2. The players receive points if both members of the pair produce the same word. Since players are unable to communicate with each other, their best strategy for producing the same word is to type in words that are relevant to the image, thus providing useful labels. As another example, Verbosity (von Ahn et al., 2006) is a game designed to collect common-sense facts about objects. Players are again randomly paired, with one partner serving as a describer and the other as a guesser. The describer is given an object (such as a sock) and asked to list

17. `https://www.zooniverse.org`
18. `https://www.scienceathome.org`

Figure 2: A screenshot of the ESP Game. Image originally appeared in von Ahn and Dabbish (2008).

facts about the object ("it is a kind of clothing" or "it is related to feet"). The guesser is shown these facts and must try to guess the object as quickly as possible. To win, it is in the describer's best interest to come up with facts that succinctly and accurately describe the object, which then become a part of a database of common knowledge statements.

Von Ahn and Dabbish (2008) credit the success of these games in part to following known game-design principles, such as incorporating extra challenges through game features like timed responses, score keeping and leaderboards, and randomness (Malone, 1980, 1982). However, there are some caveats of gamification. Hamari et al. (2014) recently conducted an extensive survey of the literature on gamification and found that it is more successful with some types of users than others, and may generally be less effective in scenarios in which people are prone towards exhibiting rational behavior, which may include paid crowdsourcing sites since at least some crowdworkers are already be in the mindset of maximizing pay.

Law et al. (2016) examined the possibility of appealing to workers' curiosity as a source of intrinsic motivation. Their work was inspired by the information gap theory of curiosity, which suggests that when people are made aware that there is a gap in their knowledge, they actively seek out the information needed to fill in this gap. They suggested several "curiosity interventions" aimed at stoking workers' curiosity and showed that these interventions led to workers completing more tasks and performing better. Curiosity interventions are especially effective on tasks that are inherently less interesting.

6.5 Crowdworker Communication and Forum Usage

Finally, several recent papers have explored the questions of how, why, and to what extent crowdworkers communicate with each other.

Gray et al. (2016) conducted a mixed-method study to understand communication among crowdworkers on four crowdsourcing platforms. Through extensive ethnographic fieldwork and in-person interviews with 118 crowdworkers in India, the researchers uncov-

ered three common categories of communication among workers. First, workers help each other with administrative overhead in order to reduce their costs. For example, a crowdworker might seek help figuring out how to receive her payments from a crowdsourcing platform, which can be nontrivial for Indian workers. Second, crowdworkers share information about good tasks and reputable (or irreputable) task requesters. Third, crowdworkers help each other complete specific tasks. More generally, crowdworkers seek each other out to recreate the social connections and support structures that are common in most traditional jobs but missing from crowdwork. This idea that crowdworkers communicate and collaborate with each other is supported by the work of Gupta et al. (2014), who also conducted interviews of crowdworkers in India.

This line of work suggests that crowdworkers are not independent, but rather that there is a hidden communication network among workers. Yin et al. (2016) attempted to quantify and map the hidden network of workers on Mechanical Turk in order to better understand its scale and structure as well as how is it used. To do this, the researchers designed and launched a task on Mechanical Turk. When a worker accepted the task, she was first asked to create a nickname for herself. She then filled out a brief demographic survey and was asked to answer two free-form questions about her experiences on Mechanical Turk: why she started Turking and what motivated her to keep Turking. (These questions were selected based on the results of a pilot study in which crowdworkers were asked what they most wanted to know about other crowdworkers, with the goal of generating interesting questions.) The worker was then asked to pause and swap nicknames with other workers she knows who had already completed the task or might be interested in completing it. If she entered another worker's nickname, she was asked several questions about their communication patterns, and an edge was created between them in the network. Finally, the worker was given the chance to explore the partially constructed network, viewing basic information on all workers (including their answers to the two free-form questions above), and more extensive information about those workers with whom she had exchanged nicknames. She was also given a personalized link she could use to return to the network later and add more connections.

The resulting communication network is shown in Figure 3. Over a period of several weeks, 10,354 workers completed the task. (Stewart et al. 2015 estimated the number of active workers on Mechanical Turk to be only 7,300, so it is likely that this task was completed by a large fraction of active workers, mitigating potential issues with sample bias.) These 10,354 workers reported a total of 5,268 connections. Since workers were not financially incentivized to add connections, this number is probably an underestimate of the true number of connected pairs. Roughly 13% of workers were connected to at least one other worker. On average these workers had 7.6 connections, and the maximum degree of any worker was 321. The largest connected component contained 994 workers, or about 72% of connected workers.

While many different methods of communicating were reported, by far the most common was through online forums. In fact, 90% of all edges were between pairs of workers who communicate via forums. This finding is in line with other work that examined the important role that forums play in crowdsourcing (Martin et al., 2014). Different online forums create different but overlapping subcommunities in the network, as illustrated in Figure 4. The researchers' analysis showed that these subcommunities differ in terms of topo-

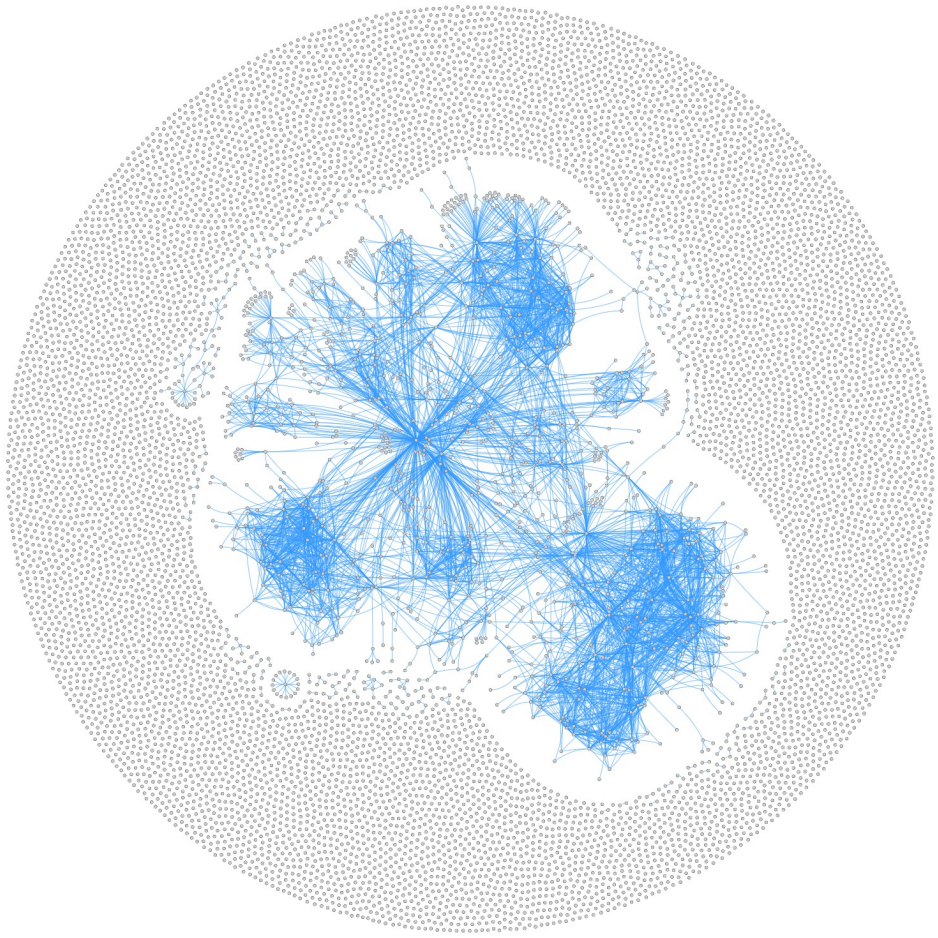
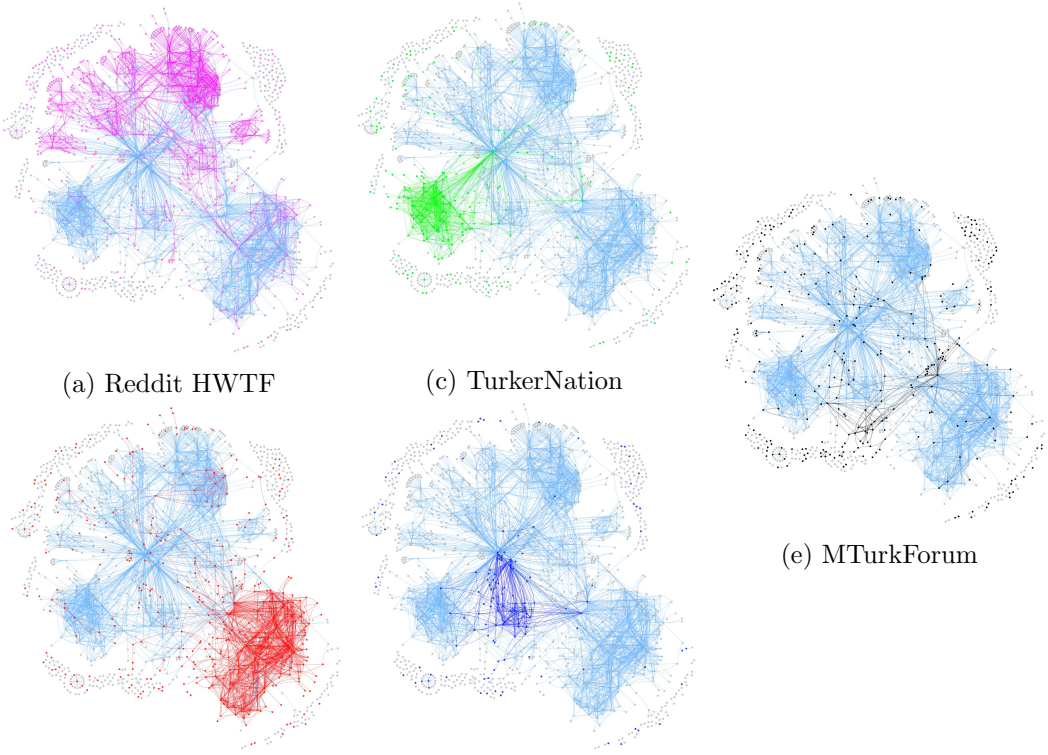


Figure 3: The communication network among Amazon Mechanical Turk workers. Image originally appeared in Yin et al. (2016).

logical structure, dynamics, and the content of communication, with some acting more as social communities and others more like broadcasting platforms.

Although it is impossible to establish causal claims on the basis of their study, Yin et al. (2016) did find correlations between a worker's connectivity and different measures of success on Mechanical Turk. In particular, they found that connected workers tended to find their task faster, were more likely to have been active on Mechanical Turk longer, were more likely to have achieved Mechanical Turk's "Master" qualification, and had a higher approval rate on average.

The high level of communication among crowdworkers and widespread use of forums have several immediate implications for researchers. First, researchers should keep in mind that the set of crowdworkers who choose to do a task may not be an independent sample of the worker pool since workers often share good tasks. Second, it can be in a researcher's best



(a) Reddit HWTF (c) TurkerNation (e) MTurkForum

(b) MTurkGrind (d) Facebook

Figure 4: Subnetworks for Reddit HWTF (magenta), MTurkGrind (red), TurkerNation (green), Facebook (blue), and MTurkForum (black). Images originally appeared in Yin et al. (2016).

interest to monitor popular forums such as Turker Nation[19], MTurk Forum[20], and Reddit HITs Worth Turking For[21] while their tasks are running to be aware of any potential issues that workers are discussing.

7. Discussion and Additional Best Practices

We have explored examples of four different ways in which machine learning researchers can put crowdsourcing to use in their own research: to generate data, to evaluate and debug models, to build hybrid intelligence systems, and to run behavioral experiments that inform the design of future machine learning systems. We have also reviewed the results of a variety of behavioral and user studies aimed at understanding the crowd and have discussed the implications of these studies for researchers who use crowdsourcing in their own research.

We conclude this survey with a discussion of additional crowdsourcing best practices that are rarely mentioned in the literature, despite their importance to the success of a project.

19. `http://turkernation.com`
20. `http://www.mturkforum.com`
21. `https://www.reddit.com/r/HITsWorthTurkingFor/`

7.1 Maintain a Good Relationship With Crowdworkers

There are several reasons why it can be valuable for a researcher to build a relationship with the community of crowdworkers and maintain a good reputation among them. As discussed in Section 6.5, workers share information about both tasks and requesters among themselves, especially through forums. Workers will be discouraged from accepting a task if other workers have complained about bugs, slow payments, or other issues. Experienced Mechanical Turk workers also commonly use tools like TurkOpticon[22] that allow them to view requester ratings broken down by communicativity, generosity, fairness, and promptness (Irani and Silberman, 2013). There are also tools available that allow workers to be notified when a favorite requester posts a new task. Being known as a good requester therefore brings more attention to one's tasks, while being known as a bad requester can deter experienced workers.

To maintain a good reputation, researchers should actively monitor and respond to any email or questions from workers so that any potential problems or bugs can be caught quickly. It is worth planning in advance to make sure that someone will be available to communicate with workers while a new task is running.

Researchers should pay fair wages (at least the equivalent of U.S. federal minimum wage, as discussed in Section 6.3) and approve work quickly since on many platforms workers are not paid until their work is approved. It is also good practice to avoid rejecting work. On Mechanical Turk, for example, a lowered approval rate can have a damaging effect on a worker's ability to find new work. Rejecting the work of a well-meaning worker who makes a mistake can therefore harm that worker's future income (Mason and Suri, 2012).

Finally, researchers should strive to be a ethical requesters. A good way to start is by reviewing the Dynamo project's guidelines for academic requesters[23] (Salehi et al., 2015).

7.2 Good Task Design Matters

Crowdworkers cannot be expected to excel at a task with unclear instructions or a confusing user interface. Any ambiguity will increase the rate of errors since it is difficult and time consuming for workers to ask clarifying questions (Rao and Michel, 2016). Both the instructions and UI should be piloted on a small batch of workers to make sure that they are clear before launching a task at a large scale. Some evidence suggests that including examples in a task's description or instructions is correlated with lower levels of confusion among workers and with workers more quickly accepting the task (Jain et al., 2017). Including quiz questions can also be an effective way to check for clarity. In addition to increased clarity, an attractive and easy-to-use UI may help keep workers engaged.

7.3 Pilot, Pilot, Pilot

Last but not least, the effective use of pilots is crucial to a project's success. Crowdsourcing platforms allow for quick and easy iteration of task design and experimentation (Mason and Suri, 2012). One of the primary benefits of crowdsourcing platforms is the ability to quickly pilot new or modified task designs on small batches of crowdworkers before making a commitment. As is the case with any software system, tasks often have bugs that are

22. `https://turkopticon.ucsd.edu`
23. `http://wiki.wearedynamo.org/index.php/Guidelines_for_Academic_Requesters`

hard to catch until they are deployed on real users. It is often best to launch a task slowly, iterating as many times as necessary to ensure that the task is clear and bug-free.

Acknowledgments

This survey grew out of lecture notes that were originally prepared to accompany my tutorial "Crowdsourcing: Beyond Label Generation" at NIPS 2016. Thanks to all of the people—far too many to name—who sent pointers and suggestions of research to include. Thanks to Dan Goldstein, Chien-Ju Ho, Jake Hofman, Andrew Mao, Roozbeh Mottaghi, Sid Suri, Jaime Teevan, Ming Yin, and Haoqi Zhang for discussing their research and sending along slides and other material while I was preparing the tutorial. Thanks to the authors of Lasecki et al. (2012) and von Ahn and Dabbish (2004) and to my coauthors on Yin et al. (2016) for allowing me to include images from their papers. Thanks to Chien-Ju Ho, Andrew Mao, Joelle Pineau, Sid Suri, Hanna Wallach, and especially Ming Yin for extended discussions and valuable feedback on previous versions of this survey. Finally, huge thanks to the four anonymous JMLR reviewers who went above and beyond their duty to give countless valuable comments, suggestions, and references that greatly improved the breadth and structure of this survey. This was a crowdsourced effort.

References

Jacob Abernethy, Yiling Chen, and Jennifer Wortman Vaughan. Efficient market making via convex optimization, and a connection to online learning. *ACM Transactions on Economics and Computation*, 1(2):Article 12, 2013.

Ittai Abraham, Omar Alonso, Vasilis Kandylas, Rajesh Patel, Steven Shelford, and Aleksandrs Slivkins. How many workers to ask? Adaptive exploration for collecting high quality labels. In *SIGIR*, 2016.

Arpit Agarwal, Debmalya Mandal, David C. Parkes, and Nisarg Shah. Peer prediction with heterogeneous users. In *ACM EC*, 2017.

Omar Alonso. Implementing crowdsourcing-based relevance experimentation: An industrial perspective. *Information Retrieval*, 16(2):101–120, 2013.

Omar Alonso, Daniel E. Rose, and Benjamin Stewart. Crowdsourcing for relevance evaluation. *ACM SigIR Forum*, 42(2):9–15, 2008.

Vamshi Ambati, Stephan Vogel, and Jaime Carbonell. Collaborative workflow for crowdsourcing translation. In *CSCW*, 2012.

Paul André, Haoqi Zhang, Juho Kim, Lydia B. Chilton, Steven P. Dow, and Robert C. Miller. Community clustering: Leveraging an academic crowd to form coherent conference sessions. In *HCOMP*, 2013.

Julia Angwin, Je Larson, Surya Mattu, and Lauren Kirchner. Machine bias: There's software used across the country to predict future criminals and it's biased

against blacks. ProPublica article accessed at https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing, 2016.

Pavel D. Atanasov, Phillip Rescober, Eric Stone, Samuel A. Swift, Emile Servan-Schreiber, Philip E. Tetlock, Lyle Ungar, and Barbara Mellers. Distilling the wisdom of crowds: Prediction markets versus prediction polls. *Management Science*, 63(3):691–706, 2017.

Bahadir Ismail Aydin, Yavuz Selim Yilmaz, Yaliang Li, Qi Li, Jing Gao, and Murat Demirbas. Crowdsourcing for multiple-choice question answering. In *AAAI*, 2014.

Solon Barocas and Andrew Selbst. Big data's disparate impact. *California Law Review*, 104, 2016.

Jonathan Baron, Barbara A. Mellers, Philip E. Tetlock, Eric Stone, and Lyle H. Ungar. Two reasons to make aggregated probability forecasts more extreme. *Decision Analysis*, 11(2):133–145, 2014.

Joyce Berg, Robert Forsythe, Forrest Nelson, and Thomas Rietz. Results from a dozen years of election futures markets research. *Handbook of experimental economics results*, 1:742–751, 2008.

Michael Bernstein, Greg Little, Rob Miller, Bjoern Hartmann, Mark Ackerman, David Karger, David Crowell, and Katrina Panovich. Soylent: A word processor with a crowd inside. In *UIST*, 2010.

Anant Bhardwaj, Juho Kim, Steven P. Dow, David Karger, Sam Madden, Robert C. Miller, and Haoqi Zhang. Attendee-sourcing: Exploring the design space of community-informed conference scheduling. In *HCOMP*, 2014.

Jeffrey P. Bigham. Reaching dubious parity with hamstrung humans. Blog post accessed at http://jeffreybigham.com/blog/2017/reaching-dubious-parity-with-hamstrung-humans.html, 2017.

David M. Blei and John D. Lafferty. Topic models. *Text mining: Classification, clustering, and applications*, 10(71):34, 2009.

David M. Blei, Andrew Y. Ng, , and Michael I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.

Jordan Boyd-Graber, Yuening Hu, and David Mimno. Applications of topic models. *Foundations and Trends in Information Retrieval*, 11(2–3):143–296, 2017.

Jonathan Bragg, Mausam, and Daniel S. Weld. Crowdsourcing multi-label classification for taxonomy creation. In *HCOMP*, 2013.

Michael Brooks, Saleema Amershi, Bongshin Lee, Steven Drucker, Ashish Kapoor, and Patrice Simard. FeatureInsight: Visual support for error-driven feature ideation in text classification. In *IEEE VAST*, 2015.

Michael Buhrmester, Tracy Kwang, and Samuel D. Gosling. Amazon's Mechanical Turk: A new source of inexpensive, yet high-quality, data? *Perspectives on Psychological Science*, 6(1):3–5, 2011.

Chris Callison-Burch and Mark Dredze. Creating speech and language data with Amazon's Mechanical Turk. In *NAACL HLT Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, 2010.

Alex Campolo, Madelyn Sanfilippo, Meredith Whittaker, and Kate Crawford. AI Now 2017 Report. Accessed at `https://ainowinstitute.org/AI_Now_2017_Report.pdf`, 2017.

Logan Casey, Jesse Chandler, Adam Seth Levine, Andrew Proctor, and Dara Z. Strolovitch. Intertemporal differences among MTurk worker demographics. Working paper on PsyArXiv, 2017.

Dana Chandler and Adam Kapelner. Breaking monotony with meaning: Motivation in crowdsourcing markets. *Journal of Economic Behavior and Organization*, 90:123–133, 2013.

Jesse Chandler, Pam Mueller, and Gabriele Paolacci. Nonnaïveté among Amazon Mechanical Turk workers: Consequences and solutions for behavioral researchers. *Behavior Research Methods*, 46(1):112–130, 2014.

Jesse J. Chandler and Gabriele Paolacci. Lie for a dime: When most prescreening responses are honest but most study participants are imposters. *Social Psychological and Personality Science*, 8(5):500–508, 2017.

Jonathan Chang, Jordan Boyd-Graber, Chong Wang, Sean Gerrish, and David M. Blei. Reading tea leaves: How humans interpret topic models. In *NIPS*, 2009.

Shuchi Chawla, Jason D. Hartline, and Balasubramanian Sivan. Optimal crowdsourcing contests. *Games and Economic Behavior*, 2015.

Yiling Chen and David M. Pennock. A utility framework for bounded-loss market makers. In *UAI*, 2007.

Yiling Chen and Jennifer Wortman Vaughan. A new understanding of prediction markets via no-regret learning. In *ACM EC*, 2010.

Yiling Chen, Arpita Ghosh, Michael Kearns, Tim Roughgarden, and Jennifer Wortman Vaughan. Mathematical foundations of social computing. *Communications of the ACM*, 59(12):102–108, December 2016.

Lydia Chilton, Juho Kim, Paul André, Felicia Cordeiro, James Landay, Dan Weld, Steven P. Dow, Robert C. Miller, and Haoqi Zhang. Frenzy: Collaborative data organization for creating conference sessions. In *CHI*, 2014.

Lydia B. Chilton, Greg Little, Darren Edge, Daniel S. Weld, and James A. Landay. Cascade: Crowdsourcing taxonomy creation. In *CHI*, 2013.

Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data, Special Issue on Social and Technical Trade-Offs*, 2017.

Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *KDD*, 2017.

Anirban Dasgupta and Arpita Ghosh. Crowdsourced judgement elicitation with endogenous proficiency. In *WWW*, 2013.

Susan B. Davidson, Sanjeev Khanna, Tova Milo, and Sudeepa Roy. Top-k and clustering with noisy comparisons. *ACM Transactions on Database Systems*, 39(4):35:1–39, 2014.

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

Gianluca Demartini, Djellel Eddine Difallah, and Philippe Cudré-Mauroux. Zencrowd: Leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In *WWW*, 2012.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009.

Berkeley J. Dietvorst, Joseph P. Simmons, and Cade Massey. Algorithm aversion: People erroneously avoid algorithms after seeing them err. *Journal of Experimental Psychology: General*, 144(1):114, 2015.

Berkeley J. Dietvorst, Joseph P. Simmons, and Cade Massey. Overcoming algorithm aversion: People will use imperfect algorithms if they can (even slightly) modify them. *Management Science*, 2016.

Djellel Eddine Difallah, Michele Catasta, Gianluca Demartini, Panagiotis G. Ipeirotis, and Philippe Cudré-Mauroux. The dynamics of micro-task crowdsourcing: The case of Amazon MTurk. In *WWW*, 2015.

Dominic DiPalantino and Milan Vojnovic. Crowdsourcing and all-pay auctions. In *ACM EC*, 2009.

Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. CoRR arXiv:1702.08608, 2017.

Mary T. Dzindolet, Linda G. Pierce, Hall P. Beck, and Lloyd A. Dawe. The perceived utility of human and automated aids in a visual detection task. *Human Factors*, 44(1): 79–94, 2002.

Robert C. Edgar and Serafim Batzoglou. Multiple sequence alignment. *Current opinion in structural biology*, 16(3):368–373, 2006.

Ju Fan, Guoliang Li, Beng Chin Ooi, Kian-lee Tan, and Jianhua Feng. icrowd: An adaptive crowdsourcing framework. In *SIGMOD*, 2015.

Oluwaseyi Feyisetan, Elena Simperl, Max Van Kleek, and Nigel Shadbolt. Improving paid microtasks through gamification and adaptive furtherance incentives. In *WWW*, 2015.

Urs Fischbacher and Franziska Föllmi-Heusi. Lies in disguise: An experimental study on cheating. *Journal of the European Economic Association*, 11(3):525–547, 2013.

Chao Gao, Yu Lu, and Dengyong Zhou. Exact exponent in optimal rates for crowdsourcing. In *ICML*, 2016.

Xi Alice Gao, Yoram Bachrach, Peter Key, and Thore Graepel. Quality expectation-variance tradeoffs in crowdsourcing contests. In *AAAI*, 2012.

Xi Alice Gao, Andrew Mao, Yiling Chen, and Ryan Prescott Adams. Trick or treat: Putting peer prediction to the test. In *ACM EC*, 2014.

Yashesh Gaur, Florian Metze, Yajie Miao, and Jeffrey P. Bigham. Using keyword spotting to help humans correct captioning faster. In *INTERSPEECH*, 2015.

Yashesh Gaur, Florian Metze, and Jeffrey P. Bigham. Manipulating word lattices to incorporate human corrections. In *INTERSPEECH*, 2016.

Timnit Gebru, Jonathan Krause, Jia Deng, and Li Fei-Fei. Scalable annotation of fine-grained objects without experts. In *CHI*, 2017.

Arpita Ghosh, Satyen Kale, and Preston McAfee. Who moderates the moderators? Crowdsourcing abuse detection in user-generated content. In *ACM EC*, 2011.

Daniel G. Goldstein, R. Preston McAfee, and Siddharth Suri. The cost of annoying ads. In *WWW*, 2013.

Daniel G. Goldstein, Siddharth Suri, R. Preston McAfee, Matthew Ekstrand-Abueg, and Fernando Diaz. The economic and cognitive costs of annoying display advertisements. *Journal of Marketing Research*, 51(6):742—752, 2014.

Ryan Gomes, Peter Welinder, Andreas Krause, and Pietro Perona. Crowdclustering. In *NIPS*, 2011.

Joseph K. Goodman and Gabriele Paolacci. Crowdsourcing consumer research. *Journal of Consumer Research*, 44(1):196–210, 2017.

Mary L. Gray, Siddharth Suri, Syed Shoaib Ali, and Deepti Kulkarni. The crowd is a collaborative network. In *CSCW*, 2016.

Neha Gupta, David Martin, Benjamin V. Hanrahan, and Jacki O'Neil. Turk-life in India. In *the Intnernational Conference on Supporting Groupwork*, 2014.

Juho Hamari, Jonna Koivisto, and Harri Sarsa. Does gamification work? – A literature review of empirical studies on gamification. In *Hawaii International Conference on System Sciences*, 2014.

Robin Hanson. Combinatorial information market design. *Information Systems Frontiers*, 5(1):105–119, 2003.

Christopher G. Harris. You're hired! An examination of crowdsourcing incentive models in human resource tasks. In *WSDM Workshop on Crwdsourcing for Search and Data Mining*, 2011.

Jeffrey Heer and Michael Bostock. Crowdsourcing graphical perception: Using Mechanical Turk to assess visualization design. In *CHI*, 2010.

Hannes Heikinheimo and Antti Ukkonen. The crowd-median algorithm. In *HCOMP*, 2013.

Chien-Ju Ho and Jennifer Wortman Vaughan. Online task assignment in crowdsourcing markets. In *AAAI*, 2012.

Chien-Ju Ho, Shahin Jabbari, and Jennifer Wortman Vaughan. Adaptive task assignment for crowdsourced classification. In *ICML*, 2013.

Chien-Ju Ho, Aleksandrs Slivkins, Siddharth Suri, and Jennifer Wortman Vaughan. Incentivizing high quality crowdwork. In *WWW*, 2015.

Chien-Ju Ho, Aleksandrs Slivkins, and Jennifer Wortman Vaughan. Adaptive contract design for crowdsourcing markets: Bandit algorithms for repeated principal-agent problems. *Journal of Artificial Intelligence Research*, 55:317–359, 2016.

John J. Horton, David Rand, and Richard Zeckhauser. The online laboratory: Conducting experiments in a real labor market. *Experimental Economics*, 14(3):399–425, 2011.

Yuening Hu, Jordan Boyd-Graber, Brianna Satinoff, and Alison Smith. Interactive topic modeling. *Machine Learning*, 95:423–469, 2014.

Lilly C. Irani and M. Six Silberman. Turkopticon: Interrupting worker invisibility in Amazon Mechanical Turk. In *CHI*, 2013.

Ayush Jain, Akash Das Sarma, Aditya Parameswaran, and Jennifer Widom. Understanding workers, developing effective tasks, and enhancing marketplace dynamics: A study of a large crowdsourcing marketplace. *Proceedings of the VLDB Endowment*, 10(7):829–840, 2017.

Jongbin Jung, Connor Concannon, Ravi Shro, Sharad Goel, and Daniel G. Goldstein. Simple rules for complex decisions. CoRR arXiv:1702.04690, 2017.

Radu Jurca and Boi Faltings. Mechanisms for making crowds truthful. *Journal of Artificial Intelligence Research*, 34:209–253, 2009.

Ece Kamar. Directions in hybrid intelligence: Complementing AI systems with human intelligence. Abstract for IJCAI Early Career Spotlight Track Talk, 2016.

Ece Kamar and Eric Horvitz. Incentives for truthful reporting in crowdsourcing (short paper). In *AAMAS*, 2012.

Vijay Kamble, David Marn, Nihar Shah, Abhay Parekh, and Kannan Ramachandran. Truth serums for massively crowdsourced evaluation tasks. CoRR arXiv:1507.07045, 2015.

Theofanis Karaletsos, Serge Belongie, and Gunnar Rätsch. Bayesian representation learning with oracle constraints. In *ICLR*, 2016.

David Karger, Sewoong Oh, and Devavrat Shah. Iterative learning for reliable crowdsourcing systems. In *NIPS*, 2011.

David Karger, Sewoong Oh, and Devavrat Shah. Budget-optimal task allocation for reliable crowdsourcing systems. *Operations Research*, 62:1–24, 2014.

Gabriella Kazai. In search of quality in crowdsourcing for search engine evaluation. In *ECIR*, 2011.

Ashish Khetan and Sewoong Oh. Achieving budget-optimality with adaptive schemes in crowdsourcing. In *NIPS*, 2016.

Hyun-Chul Kim and Zoubin Ghahramani. Bayesian classifier combination. In *AISTATS*, 2012.

Joy Kim, Sarah Sterman, Allegra Argent Beal Cohen, and Michael S. Bernstein. Mechanical novel: Crowdsourcing complex work through reflection and revision. In *CSCW*, 2017.

Juho Kim, Haoqi Zhang, Paul André, Lydia B. Chilton, Wendy Mackay, Michel Beaudouin-Lafon, Robert C. Miller, and Steven P. Dow. Cobi: A community-informed conference scheduling tool. In *UIST*, 2013.

Aniket Kittur, Ed Chi, and Bongwon Suh. Crowdsourcing user studies with Mechanical Turk. In *CHI*, 2008.

Aniket Kittur, Boris Smus, Susheel Khamkar, and Robert E. Kraut. Crowdforge: Crowdsourcing complex work. In *UIST*, 2011.

Aniket Kittur, Jeffrey V. Nickerson, Michael Bernstein, Elizabeth Gerber, Aaron Shaw, John Zimmerman, Matt Lease, and John Horton. The future of crowd work. In *CSCW*, 2013.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *ICML*, 2017.

Adriana Kovashka, Olga Russakovsky, Li Fei-Fei, and Kristen Grauman. Crowdsourcing in computer vision. *Foundations and Trends in Computer Graphics and Vision*, 10(3):177–243, 2016.

Chinmay E. Kulkarni, Richard Socher, Michael S. Bernstein, and Scott R. Klemmer. Scaling short-answer grading by combining peer assessment with algorithmic scoring. In *ACM Conference on Learning@scale*, 2014.

Raja S. Kushalnagar, Walter S. Lasecki, and Jeffrey P. Bigham. A readability evaluation of real-time crowd captions in the classroom. In *ASSETS*, 2012.

Walter S. Lasecki and Jeffrey P. Bigham. Online quality control for real-time crowd captioning. In *ASSETS*, 2012.

Walter S. Lasecki, Christopher D. Miller, Adam Sadilek, Andrew Abumoussa, Donato Borrello, Raja Kushalnagar, and Jeffrey P. Bigham. Real-time captioning by groups of non-experts. In *UIST*, 2012.

Walter S. Lasecki, Christopher D. Miller, and Jeffrey P. Bigham. Warping time for more effective real-time crowdsourcing. In *CHI*, 2013.

Walter S. Lasecki, Christopher D. Miller, Iftekhar Naim, Raja Kushalnagar, Adam Sadilek, Daniel Gildea, and Jeffrey P. Bigham. Scribe: Deep integration of human and machine intelligence to caption speech in real-time. *Communications of the ACM*, 60(11), 2017.

Edith Law, Ming Yin, Joslin Goh, Kevin Chen, Michael Terry, and Krzysztof Z. Gajos. Curiosity killed the cat, but makes crowdwork better. In *CHI*, 2016.

John Le, Andy Edmonds, Vaughn Hester, and Lukas Biewald. Ensuring quality in crowdsourced search relevance evaluation: The effects of training question distribution. In *SIGIR Workshop on Crowdsourcing for Search Evaluation*, 2010.

Tak Yeon Lee, Casey Dugan, Werner Geyer, Tristan Ratchford, Jamie Rasmussen, N. Sadat Shami, and Stela Lupushor. Experiments on motivational feedback for crowdsourced workers. In *ICWSM*, 2013.

Martin Lermen and Knut Reinert. The practical use of the A* algorithm for exact multiple sequence alignment. *Journal of Computational Biology*, 7(5):655–671, 2000.

Guoliang Li, Jiannan Wang, Yudian Zheng, and Michael J. Franklin. Crowdsourced data management: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 28(9): 2296–2319, 2016.

Qi Li, Yaliang Li, Jing Gao, Lu Su, Bo Zhao, Murat Demirbas, Wei Fan, and Jiawei Han. A confidence-aware approach for truth discovery on long-tail data. *Proceedings of the VLDB Endowment*, 8(4):425–436, 2014a.

Qi Li, Yaliang Li, Jing Gao, Bo Zhao, Wei Fan, and Jiawei Han. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In *SIGMOD*, 2014b.

Zachary C. Lipton. The mythos of model interpretability. CoRR arXiv:1606.03490, 2016.

Leib Litman, Jonathan Robinson, and Cheskie Rosenzweig. The relationship between motivation, monetary compensation, and data quality among US- and India-based workers on Mechanical Turk. *Behavioral Research Methods*, 47(2):519–528, 2014.

Qiang Liu, Jian Peng, and Alexander Ihler. Variational inference for crowdsourcing. In *NIPS*, 2012a.

Qiang Liu, Mark Steyvers, and Alexander Ihler. Scoring workers in crowdsourcing: How many control questions are enough? In *NIPS*, 2013.

Xuan Liu, Meiyu Lu, Beng Chin Ooi, Yanyan Shen, Sai Wu, and Meihui Zhang. CDAS: A crowdsourcing data analytics system. *Proceedings of the VLDB Endowment*, 5(10):1040–1051, 2012b.

Stuart P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 2(28):129–137, 1982.

Jennifer M. Logg. Theory of machine: When do people rely on algorithms? Harvard Business School NOM Unit Working Paper No. 17-086, 2017.

Yin Lou, Rich Caruana, and Johannes Gehrke. Intelligible models for classification and regression. In *KDD*, 2012.

Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *KDD*, 2013.

Fenglong Ma, Yaliang Li, Qi Li, Minghui Qiu, Jing Gao, Shi Zhi, Lu Su, Bo Zhao, Heng Ji, and Jiawei Han. FaitCrowd: Fine grained truth discovery for crowdsourced data aggregation. In *SIGMOD*, 2015.

Thomas W. Malone. What makes things fun to learn? Heuristics for designing instructional computer games. In *ACM SIGSMALL Symposium and the First SIGPC Symposium on Small Systems*, 1980.

Thomas W. Malone. Heuristics for designing enjoyable user interfaces: Lessons from computer games. In *CHI*, 1982.

Adam Marcus, Eugene Wu, David Karger, Samuel Madden, and Robert Miller. Human-powered sorts and joins. *Proceedings of the VLDB Endowment*, 5(1):13–24, 2011.

David Martin, Benjamin V. Hanrahan, Jacki O'Neill, and Neha Gupta. Being a Turker. In *CSCW*, 2014.

Winter Mason and Siddharth Suri. Conducting behavioral research on Amazon's Mechanical Turk. *Behavior Research Methods*, 44(1):1–23, 2012.

Winter Mason and Duncan J. Watts. Financial incentives and the "performance of crowds". In *HCOMP*, 2009.

Arya Mazumdar and Barna Saha. Clustering via crowdsourcing. CoRR arXiv:1604.01839, 2016.

Barbara Mellers, Eric Stone, Pavel Atanasov, Nick Rohrbaugh, S. Emlen Metz, Lyle Ungar, Michael M. Bishop, Michael Horowitz, Ed Merkle, and Philip Tetlock. The psychology of intelligence analysis: Drivers of prediction accuracy in world politics. *Journal of Experimental Psychology: Applied*, 21(2):1–14, 2015a.

Barbara Mellers, Eric Stone, Terry Murray, Angela Minster, Nick Rohrbaugh, Michael Bishop, Eva Chen, Joshua Baker, Yuan Hou, Michael Horowitz, Lyle Ungar, and Philip Tetlock. Identifying and cultivating superforecasters as a method of improving probabilistic predictions. *Perspectives on Psychological Science*, 10(2):267–281, 2015b.

Nolan Miller, Paul Resnick, and Richard Zeckhauser. Eliciting informative feedback: The peer-prediction method. *Management Science*, 51(9):1359–1373, 2005.

Roozbeh Mottaghi, Sanja Fidler, Jian Yao, Raquel Urtasun, and Devi Parikh. Analyzing semantic segmentation using hybrid human-machine CRFs. In *CVPR*, 2013.

Roozbeh Mottaghi, Sanja Fidler, Alan Yuille, Raquel Urtasun, and Devi Parikh. Human-machine CRFs for identifying bottlenecks in scene understanding. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

Iftekhar Naim, Daniel Gildea, Walter Lasecki, and Jeffrey P. Bigham. Text alignment for real-time crowd captioning. In *NAACL*, 2013.

David Newman, Edwin V. Bonilla, and Wray Buntine. Improving topic coherence with regularized topic models. In *NIPS*, 2011.

Besmira Nushi, Ece Kamar, Donald Kossmann, and Eric Horvitz. On human intellect and machine failures: Troubleshooting integrative machine learning systems. In *AAAI*, 2017.

David Oleson, Alexander Sorokin, Greg Laughlin, Vaughn Hester, John Le, and Lukas Biewald. Programmatic gold: Targeted and scalable quality assurance in crowdsourcing. In *HCOMP*, 2011.

Gabriele Paolacci, Jesse Chandler, and Panagiotis G. Ipeirotis. Running experiments on Amazon Mechanical Turk. *Judgment and Decision Making*, 5:411–419, 2010.

Devi Parikh and C. Lawrence Zitnick. Human-debugging of machines. In *Second NIPS Workshop on Computational Social Science and the Wisdom of Crowds*, 2011.

Kayur Patel, Naomi Bancroft, Steven M. Drucker, James Fogarty, Andrew J. Ko, and James A. Landay. Gestalt: Integrated support for implementation and analysis in machine learning processes. In *UIST*, 2010.

Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *CVPR*, 2012.

Genevieve Patterson, Chen Xu, Hang Su, and James Hays. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision*, 108(1–2):59–81, 2014.

Michael J. Paul. Interpretable machine learning: Lessons from topic modeling. In *CHI Workshop on Human-Centered Machine Learning*, 2016.

Michael J. Paul and Mark Dredze. Discovering health topics in social media using topic models. *PloS one*, 9(8):e103408, 2014.

Ellie Pavlick, Matt Post, Ann Irvine, Dmitry Kachaev, and Chris Callison-Burch. The language demographics of Amazon Mechanical Turk. *Transactions of the Association for Computational Linguistics*, 2:79–92, 2014.

Eyal Peera, Laura Brandimarteb, Sonam Samatc, and Alessandro Acquistic. Beyond the Turk: An empirical comparison of alternative platforms for crowdsourcing online behavioral research. *Journal of Experimental Social Psychology*, 70:153–163, 2017.

Matt Post, Chris Callison-Burch, and Miles Osborne. Constructing parallel corpora for six Indian languages via crowdsourcing. In *Seventh Workshop on Statistical Machine Translation*, 2012.

Forough Poursabzi-Sangdeh, Daniel G. Goldstein, Jake Hofman, Jennifer Wortman Vaughan, and Hanna Wallach. Manipulating and measuring model interpretability. CoRR arXiv:1802.07810, 2018.

Dražen Prelec. A Bayesian truth serum for subjective data. *Science*, 306(5695):462–466, 2004.

Dražen Prelec, H. Sebastian Seung, and John McCoy. A solution to the single-question crowd wisdom problem. *Nature*, 541:532–535, 2017.

Marianne Promberger and Jonathan Baron. Do patients trust computers? *Journal of Behavioral Decision Making*, 19:455–468, 2006.

Goran Radanovic and Boi Faltings. A robust Bayesian truth serum for non-binary signals. In *AAAI*, 2013.

Goran Radanovic, Boi Faltings, and Radu Jurca. Incentives for effort in crowdsourcing using the peer truth serum. *ACM Transactions on Intelligent Systems and Technology*, 7(4):1–28, 2016.

Srinivas Rao and Amanda Michel. ProPublica's guide to Mechanical Turk. ProPublica article accessed at `https://www.propublica.org/article/propublicas-guide-to-mechanical-turk`, 2016.

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

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should I trust you?: Explaining the predictions of any classifier. In *KDD*, 2016.

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

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015a.

Olga Russakovsky, Li-Jia Li, and Li Fei-Fei. Best of both worlds: Human-machine collaboration for object annotation. In *CVPR*, 2015b.

Niloufar Salehi, Lilly Irani, Michael Bernstein, Ali Alkhatib, Eva Ogbe, Kristy Milland, and Clickhappier. We are dynamo: Overcoming stalling and friction in collective action for crowd workers. In *CHI*, 2015.

Niloufar Salehi, Jaime Teevan, Shamsi Iqbal, and Ece Kamar. Communicating context to the crowd for complex writing tasks. In *CSCW*, 2017.

Paul J. H. Schoemaker and Philip E. Tetlock. Superforecasting: How to upgrade your company's judgment. *Harvard Business Review*, 94:72–78, 2016.

Devavrat Shah and Christina E. Lee. Reducing crowdsourcing to graphon estimation, statistically. In *AISTATS*, 2018.

Nihar Shah, Dengyong Zhou, and Yuval Peres. Approval voting and incentives in crowdsourcing. In *ICML*, 2015.

Nihar B. Shah and Dengyong Zhou. Double or nothing: Multiplicative incentive mechanisms for crowdsourcing. *Journal of Machine Learning Research*, 17(165):1–52, 2016a.

Nihar B. Shah and Dengyong Zhou. No oops, you won't do it again: Mechanisms for self-correction in crowdsourcing. In *ICML*, 2016b.

Aaron D. Shaw, John J. Horton, and Daniel L. Chen. Designing incentives for inexpert human raters. In *CSCW*, 2011.

Victor Sheng, Foster Provost, and Panagiotis Ipeirotis. Get another label? Improving data quality using multiple, noisy labelers. In *KDD*, 2008.

Victor Shnayder, Arpit Agarwal, Rafael M. Frongillo, and David C. Parkes. Informed truthfulness in multi-task peer prediction. In *ACM EC*, 2016.

M. Six Silberman, Lilly Irani, and Joel Ross. Ethics and tactics of professional crowdwork. *XRDS: Crossroads, The ACM Magazine for Students*, 17(2):39–43, 2010.

M. Six Silberman, Kristy Milland, Rochelle LaPlant, Joel Ross, and Lilly Irani. Stop citing Ross et al. 2010, "Who are the crowdworkers?". Accessed at `https://medium.com/@silberman/stop-citing-ross-et-al-2010-who-are-the-crowdworkers-b3b9b1e8d300`, 2015.

Patrice Y. Simard, Saleema Amershi, David M. Chickering, Alicia Edelman Pelton, Soroush Ghorashi, Christopher Meek, Gonzalo Ramos, Jina Suh, Johan Verwey, Mo Wang, and John Wernsing. Machine teaching: A new paradigm for building machine learning systems. CoRR arXiv:1707.06742, 2017.

Daniel J. Simons and Christopher F. Chabris. Common (mis)beliefs about memory: A replication and comparison of telephone and Mechanical Turk survey methods. *PLoS ONE*, 7(12), 2012.

Rashmi R. Sinha and Kirsten Swearingen. Comparing recommendations made by online systems and friends. In *DELOS workshop: Personalisation and recommender systems in digital libraries*, 2001.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks. In *EMNLP*, 2008.

Alexander Sorokin and David Forsyth. Utility data annotation with Amazon Mechanical Turk. In *CVPRW*, June 2008.

Neil Stewart, Christoph Ungemach, Adam J. L. Harris, Daniel M. Bartels, Ben R. Newell, Gabriele Paolacci, and Jesse Chandler. The average laboratory samples a population of 7,300 Amazon Mechanical Turk workers. *Judgment and Decision Making*, September 2015.

Hao Su, Jia Deng, and Li Fei-Fei. Crowdsourcing annotations for visual object detection. In *HCOMP*, 2012.

Siddharth Suri and Duncan J. Watts. Cooperation and contagion in web-based, networked public goods experiments. *PLoS ONE*, 6(3), 2011.

Siddharth Suri, Daniel Goldstein, and Winter Mason. Honesty in an online labor market. In *HCOMP*, 2011.

Latanya Sweeney. Discrimination in online ad delivery. *Communications of the ACM*, 56 (5):44–54, 2013.

Omer Tamuz, Ce Liu, Serge Belongie, Ohad Shamir, and Adam Kalai. Adaptively learning the crowd kernel. In *ICML*, 2011.

Jaime Teevan, Shamsi Iqbal, and Curtis von Veh. Supporting collaborative writing with microtasks. In *CHI*, 2016.

Philip E. Tetlock, Barbara A. Mellers, and J. Peter Scoblic. Bringing probability judgments into policy debates via forecasting tournaments. *Science*, 355(6324):481–483, 2017.

Tian Tian and Jun Zhu. Max-margin majority voting for learning from crowds. In *NIPS*, 2015.

Lyle H. Ungar, Barbara A. Mellers, Ville Satopää, Philip Tetlock, and Jon Baron. The good judgment project: A large scale test of different methods of combining expert predictions. In *AAAI Fall Symposium: Machine Aggregation of Human Judgment*, 2012.

Blase Ur, Jonathan Bees, Sean M. Segreti, Lujo Bauer, and Lorrie Faith Cranor Nicolas Christin. Do users' perceptions of password security match reality? In *CHI*, 2016.

Berk Ustun and Cynthia Rudin. Supersparse linear integer models for optimized medical scoring systems. *Machine Learning Journal*, 102(3):349–391, 2016.

Donna Vakharia and Matthew Lease. Beyond Mechanical Turk: An analysis of paid crowd work platforms. In *Proceedings of the iConference*, 2015.

Matteo Venanzi, John Guiver, Gabriella Kazai, Pushmeet Kohli, and Milad Shokouhi. Community-based Bayesian aggregation models for crowdsourcing. In *WWW*, 2014.

Norases Vesdapunt, Kedar Bellare, and Nilesh Dalvi. Crowdsourcing algorithms for entity resolution. *Proceedings of the VLDB Endowment*, 7(12):1071–1082, 2014.

Sudheendra Vijayanarasimhan and Kristen Grauman. What's it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations. In *CVPR*, 2009.

Ramya Korlakai Vinayak and Babak Hassibi. Crowdsourced clustering: Querying edges vs triangles. In *NIPS*, 2016.

Ramya Korlakai Vinayak, Samet Oymak, and Babak Hassibi. Graph clustering with missing data: Convex algorithms and analysis. In *NIPS*, 2014.

Luis von Ahn and Laura Dabbish. Labeling images with a computer game. In *CHI*, 2004.

Luis von Ahn and Laura Dabbish. General techniques for designing games with a purpose. *Communications of the ACM*, 51(8):58–67, August 2008.

Luis von Ahn, Manuel Blum, Nicholas Hopper, and John Langford. CAPTCHA: Using hard AI problems for security. In *EUROCRYPT*, 2003.

Luis von Ahn, Manuel Blum, and John Langford. Telling humans and computers apart automatically. *Communications of the ACM*, pages 56–60, 2004.

Luis von Ahn, Mihir Kedia, and Manuel Blum. Verbosity: A game for collecting common-sense knowledge. In *CHI Notes*, 2006.

Luis von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. reCAPTCHA: Human-based character recognition via web security measures. *Science*, 321(5895):1465–1468, 2008.

Bo Waggoner and Yiling Chen. Output agreement mechanisms and common knowledge. In *HCOMP*, 2014.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Catherine Wah, Grant Van Horn, Steve Branson, Subhransu Maji, Pietro Perona, and Serge Belongie. Similarity comparisons for interactive fine-grained categorization. In *CVPR*, 2014.

Gang Wang, Christo Wilson, Xiaohan Zhao, Yibo Zhu, Manish Mohanlal, Haitao Zheng, and Ben Y. Zhao. Serf and turf: Crowdturfing for fun and profit. In *WWW*, 2012a.

Jiannan Wang, Tim Kraska, Michael J. Franklin, and Jianhua Feng. CrowdER: Crowdsourcing entity resolution. *Proceedings of the VLDB Endowment*, 5(11):1483–1494, 2012b.

Fabian L. Wauthier, Nebojsa Jojic, and Michael I. Jordan. Active spectral clustering via iterative uncertainty reduction. In *KDD*, 2012.

Peter Welinder, Steve Branson, Serge Belongie, and Perona Pietro. The multidimensional wisdom of crowds. In *NIPS*, 2010.

Kathryn Sharpe Wessling, Joel Huber, and Oded Netzer. Character misrepresentation by Amazon Turk workers: Assessment and solutions. *Journal of Consumer Research*, 44(1): 211–230, 2017.

Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier R. Movellan, and Paul L. Ruvolo. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *NIPS*, 2009.

Michael Wilber, Sam Kwak, and Serge Belongie. Cost-effective hits for relative similarity comparisons. In *HCOMP*, 2014.

Vanessa Williamson. On the ethics of crowdsourced research. *Political Science & Politics*, 49(1):77–81, 2016.

Jens Witkowski and David C. Parkes. A robust Bayesian truth serum for small populations. 2012.

Justin Wolfers and Eric Zitzewitz. Prediction markets. *The Journal of Economic Perspectives*, 18(2):107–126, 2004.

James R. Wright, Chris Thornton, and Kevin Leyton-Brown. Mechanical TA: Partially automated high-stakes peer grading. In *ACM Technical Symposium on Computer Science Education*, 2015.

Jian Yao, Sanja Fidler, and Raquel Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *CVPR*, 2012.

Teng Ye, Sangseok You, and Lionel P. Robert Jr. When does more money work? Examining the role of perceived fairness in pay on the performance quality of crowdworkers. In *ICWSM*, 2017.

Michael Yeomans, Anuj K. Shah, Sendhil Mullainathan, and Jon Kleinberg. Making sense of recommendations. *Management Science*, 2017.

Jinfeng Yi, Rong Jin, Anil K. Jain, and Shaili Jain. Crowdclustering with sparse pairwise labels: A matrix completion approach. In *HCOMP*, 2012a.

Jinfeng Yi, Rong Jin, Anil K. Jain, Shaili Jain, and Tianbao Yang. Semi crowdsourced clustering: Generalizing crowd labeling by robust distance metric learning. In *NIPS*, 2012b.

Ming Yin, Yiling Chen, and Yu-An Sun. The effects of performance-contingent financial incentives in online labor markets. In *AAAI*, 2013.

Ming Yin, Yiling Chen, and Yu-An Sun. Monetary interventions in crowdsourcing task switching. In *HCOMP*, 2014.

Ming Yin, Mary L. Gray, Siddharth Suri, and Jennifer Wortman Vaughan. The communication network within the crowd. In *WWW*, 2016.

Dong Yu and Li Deng. *Automatic speech recognition: A deep learning approach*. Springer, 2014.

Omar Zaidan and Chris Callison-Burch. Crowdsourcing translation: Professional quality from non-professionals. In *ACL*, 2011.

Rabih Zbib, Erika Malchiodi, Jacob Devlin, David Stallard, Spyros Matsoukas, Richard Schwartz, John Makhoul, Omar F. Zaidan, and Chris Callison-Burch. Machine translation of Arabic dialects. In *NAACL*, 2012.

Haoqi Zhang, Edith Law, Krzysztof Gajos, Eric Horvitz, Rob Miller, and David Parkes. Human computation tasks with global constraints. In *CHI*, 2012.

Jing Zhang, Xindong Wu, and Victor S. Sheng. Learning from crowdsourced labeled data: A survey. *Artificial Intelligence Review*, 46(4):543–576, 2016a.

Yuchen Zhang, Xi Chen, Dengyong Zhou, and Michael I. Jordan. Spectral methods meet EM: A provably optimal algorithm for crowdsourcing. *Journal of Machine Learning Research*, 17(102):1–44, 2016b.

Yudian Zheng, Guoliang Li, Yuanbing Li, Caihua Shan, and Reynold Cheng. Truth inference in crowdsourcing: Is the problem solved? *Proceedings of the VLDB Endowment*, 10(5): 541–552, 2017.

Dengyong Zhou, Sumit Basu, Yi Mao, and John Platt. Learning from the wisdom of crowds by minimax entropy. In *NIPS*, 2012.

Dengyong Zhou, Qiang Liu, John C. Platt, and Christopher Meek. Aggregating ordinal labels from crowds by minimax conditional entropy. In *ICML*, 2014.