Crowdsourced Data Management: Industry and Academic Perspectives

Adam Marcus
Unlimited Labs
marcua@marcua.net

Aditya Parameswaran
University of Illinois (UIUC)
adityagp@illinois.edu

the essence of knowledge
Boston — Delft

Foundations and Trends® in Databases

Published, sold and distributed by:
now Publishers Inc.
PO Box 1024
Hanover, MA 02339
United States
Tel. +1-781-985-4510
www.nowpublishers.com
sales@nowpublishers.com

Outside North America:
now Publishers Inc.
PO Box 179
2600 AD Delft
The Netherlands
Tel. +31-6-51115274

The preferred citation for this publication is

A. Marcus and A. Parameswaran. *Crowdsourced Data Management: Industry and Academic Perspectives*. Foundations and Trends® in Databases, vol. 6, no. 1-2, pp. 1–161, 2013.

This Foundations and Trends® issue was typeset in LaTeX using a class file designed by Neal Parikh. Printed on acid-free paper.

ISBN: 978-1-68083-091-0

Editorial Scope

Topics

Foundations and Trends® in Databases covers a breadth of topics relating to the management of large volumes of data. The journal targets the full scope of issues in data management, from theoretical foundations, to languages and modeling, to algorithms, system architecture, and applications. The list of topics below illustrates some of the intended coverage, though it is by no means exhaustive:

- Data models and query languages
- Query processing and optimization
- Storage, access methods, and indexing
- Transaction management, concurrency control, and recovery
- Deductive databases
- Parallel and distributed database systems
- Database design and tuning
- Metadata management
- Object management
- Trigger processing and active databases
- Data mining and OLAP
- Approximate and interactive query processing
- Data warehousing
- Adaptive query processing
- Data stream management
- Search and query integration
- XML and semi-structured data
- Web services and middleware
- Data integration and exchange
- Private and secure data management
- Peer-to-peer, sensornet, and mobile data management
- Scientific and spatial data management
- Data brokering and publish/subscribe
- Data cleaning and information extraction
- Probabilistic data management

Information for Librarians

Foundations and Trends® in Databases, 2013, Volume 6, 4 issues. ISSN paper version 1931-7883. ISSN online version 1931-7891. Also available as a combined paper and online subscription.

Foundations and Trends® in Databases
Vol. 6, No. 1-2 (2013) 1–161

DOI: 10.1561/1900000044

Crowdsourced Data Management: Industry and Academic Perspectives

Adam Marcus
Unlimited Labs
marcua@marcua.net

Aditya Parameswaran
University of Illinois (UIUC)
adityagp@illinois.edu

Contents

Abstract

Crowdsourcing and human computation enable organizations to accomplish tasks that are currently not possible for fully automated techniques to complete, or require more flexibility and scalability than traditional employment relationships can facilitate. In the area of data processing, companies have benefited from crowd workers on platforms such as Amazon's Mechanical Turk or Upwork to complete tasks as varied as content moderation, web content extraction, entity resolution, and video/audio/image processing. Several academic researchers from diverse areas ranging from the social sciences to computer science have embraced crowdsourcing as a research area, resulting in algorithms and systems that improve crowd work quality, latency, or cost. Given the relative nascence of the field, the academic and the practitioner communities have largely operated independently of each other for the past decade, rarely exchanging techniques and experiences. In this book, we aim to narrow the gap between academics and practitioners. On the academic side, we summarize the state of the art in crowd-powered algorithms and system design tailored to large-scale data processing. On the industry side, we survey 13 industry users (e.g., Google, Facebook, Microsoft) and 4 marketplace providers of crowd work (e.g., CrowdFlower, Upwork) to identify how hundreds of engineers and tens of million dollars are invested in various crowdsourcing solutions. Through the book, we hope to simultaneously introduce academics to real problems that practitioners encounter every day, and provide a survey of the state of the art for practitioners to incorporate into their designs. Through our surveys, we also highlight the fact that crowd-powered data processing is a large and growing field. Over the next decade, we believe that most technical organizations will in some way benefit from crowd work, and hope that this book can help guide the effective adoption of crowdsourcing across these organizations.

A. Marcus and A. Parameswaran. *Crowdsourced Data Management: Industry and Academic Perspectives*. Foundations and Trends® in Databases, vol. 6, no. 1-2, pp. 1–161, 2013.
DOI: 10.1561/1900000044.

1

Introduction

We are drowning in information, while starving for wisdom.

— E. O. Wilson

With the advent of the "data deluge" [176], organizations world-wide have been struggling with designing algorithms and systems to better process and analyze the massive quantities of data collected every day. It is estimated that 80% of this data is unstructured [205, 196], consisting largely of images, videos, and raw text. While there have been significant advances in automated mechanisms for interpreting and extracting information from unstructured data, algorithms to fully comprehend unstructured data have not been developed yet. It is widely acknowledged that we are at least several decades away from this goal [162, 120].

Humans, on the other hand, are able to analyze certain aspects of unstructured data with relative ease. Humans have an innate understanding of language, speech, and images; they are able to process, reason about, and provide solutions to problems faced often in managing and processing unstructured data. Moreover, the abundance of cheap and reliable internet connectivity throughout the world has given rise to *crowdsourcing* or *crowd work* marketplaces, such as Mechanical Turk [10] and Upwork [17], enabling the inclusion of human crowd workers in on-demand data processing tasks.

In particular, crowdsourcing has been applied in the following large-scale unstructured data processing applications (among others):

- **Content Moderation**. Workers in crowdsourcing marketplaces are often consulted for content moderation of images uploaded on web sites [5]. That is, humans are asked to determine whether user-uploaded images are appropriate for viewing by a general audience.

- **Web Extraction**. Crowd workers also contribute to tasks like information extraction from web sites. That is, workers are asked to provide specific information by looking up web sites and finding, say, phone numbers or prices at restaurants [91]. Workers can aid machines in semi-automatic information extraction systems—for instance, companies like Yahoo! [18] use crowdsourcing to build web extraction wrappers, and to verify extracted information [40, 87, 88, 142, 60].

- **Search Relevance**. Most companies with a search engine, e.g., Bing [11], Google [9], and Yahoo!, include crowd workers in evaluating the performance of their search algorithms [26].

- **Entity Resolution**. Entity Resolution, or deduplication [78] refers to the problem of identifying if two textual records refer to the same entity. Groupon and Yahoo! both use crowdsourcing for entity resolution [105, 104, 34].

- **Text Processing**. Crowdsourcing is used in spam identification [137], text classification [30, 172], translation [199], and text editing [36]. Crowdsourcing is also being used commercially for transliteration of documents [20].

- **Video and Image Processing**. Crowdsourcing is used in video analysis [53], for image labeling [160, 185], and as a visual aid [39].

Unfortunately, in all of these applications, and overall, crowdsourcing can be subjective or error-prone; it can be time-consuming (crowd workers take longer than computers); and it can be relatively costly (human workers need to be paid). Moreover, these three aspects—accuracy, latency, and cost—are correlated in complex ways, making it difficult to optimize the trade-offs among them while designing data processing algorithms and systems.

As an example of these tradeoffs, consider content moderation of images. We can ask one human worker to verify if each image is appropriate, but they may make mistakes. As a result, we may need to ask multiple humans to verify each image. However, asking multiple human workers has higher monetary cost, and might incur higher latency. Furthermore, we can ask multiple human workers to verify each image in parallel, or ask humans in sequence. The former option can incur lower latency, while the latter might have lower monetary cost since we can choose to not ask subsequent questions based on worker agreement on answers to previous ones.

With nearly a decade passing since crowdsourcing marketplaces have become commonplace, academic researchers and industry users alike have explored various mechanisms for orchestrating large scale data processing work by assembling human workers in workflows that attempt to optimize the three aspects described above (accuracy, latency, and cost), while also expanding our understanding of what is actually feasible using human workers. On the one hand, academic researchers have proposed programming languages, frameworks, systems, and algorithms, and have prototyped creative solutions to problems that are just now feasible to solve with the advent of crowdsourcing. On the other hand, several companies have been founded whose core business is to explore the use of crowd work for various "unsolvable" tasks, and many companies have embraced crowd work as a mechanism for accomplishing what was previously infeasible or inefficient.

However, *progress in academia and industry on how to best leverage crowd work for large scale data processing has largely proceeded independently.* It is essential that these two communities work in concert with one-another. Industrial users and marketplace providers have a lot of wisdom to share about the problems that are the most crucial to solve, which techniques work well in practice and which don't, as well as "best-practice" implementations of workflows involving crowds. Academia has much to say about how to leverage large scale data processing in an optimized fashion in many settings.

The primary goal of our book is to bridge the gap between crowdsourcing practitioners and academic crowdsourcing researchers. With this goal in mind, we will:

- summarize the state of the art in research on crowd-powered algorithms and systems for data processing, and
- survey industry users and marketplace providers of crowd work to identify their accomplishments and highlight the unsolved problems they struggle with.

By describing the state-of-the-art in crowd-powered data processing from academia, we hope to provide a reference for industry participants to see if academia have solved their problems, and to articulate the areas that have the most potential for future research. By engaging industry users and marketplace vendors, we hope to highlight their chief pain-points and concerns, identify the status quo, and articulate which areas of future research have the most potential for impact. Identifying the "tried-and-true" methods that work well in industry settings that are yet to be formally analyzed in academia would also be valuable for academics. Furthermore, industry and marketplace vendors can see if they all face the same challenges, or if other industry or marketplace participants have solved the problems that they face.

Overall, by connecting the marketplace providers, industry users, and academia, we hope that these groups are educated about the problems and solutions that each of them has been working on, in order to facilitate more transparency, more openness, and also the ability to begin a frank dialog about the problems and the future of crowdsourcing.

A secondary goal of this book is to argue that *crowdsourcing is here to stay*. A common criticism in academia is that crowdsourcing is a fad; that not too many industry users care about crowdsourcing; and that the recent interest in crowdsourcing is going to disappear in a few years. Our thesis is that this is simply not the case. As we will find out in the industry portions of this book, crowdsourcing is *an essential ingredient for any company working with large datasets*. Companies are sometimes not willing to talk about how much they use crowdsourcing because they are either ashamed about admitting that they rely on crowds instead of sophisticated software or hardware, or paradoxically because they consider it to be their "secret sauce." Through our conversations with industry users, we will highlight the hundreds of employees and tens of millions of dollars that companies invest into crowd work.

A reader might note that in our coverage of industry users and marketplace providers of crowdsourcing, we do not dedicate attention to an important third group in crowd work: the crowd workers themselves. We first note that the study of crowd workers is relatively well-explored, with several seminal and ongoing surveys of different crowds over time [161, 99, 177, 165]. Second, our focus in this study is on the gap between industry and academia, especially as it relates to large-scale data processing, and we did not view workers as having a large influence on this gap. Understanding and designing for crowd workers is of utmost importance for the health and future of crowd work, but given the existing studies of the crowd and our specific research aims, it will not be the focus of our attention.

1.1 Chapter Summaries

We have structured the book into the following chapters[1]:

- **Background (Remainder of this chapter)**. To establish fluency in crowdsourcing or crowd work, we present the lifecycle of an example task, touching on terminology we will use throughout the book.

- **Related work (Chapter 2)**. The research literature has over half a decade of contributions on various aspects of of crowdsourcing, and we summarize many of the fields and papers that have influenced crowd-powered data processing.

- **Crowd-powered algorithms (Chapter 3)**. At its core, data processing relies on a set of algorithms to filter, sort, summarize, categorize, enumerate, and join datasets. In this chapter, we summarize the state of the art of making these algorithms crowd-powered, and highlight some core models and considerations for crowd-powered algorithm design.

- **Crowd-powered systems (Chapter 4)**. Some of the earliest contributions to crowd-powered data processing research were database systems that integrated the concept of humans to optimize and perform

[1] As you explore the chapters, keep in mind that crowd-powered data processing is an active and fast-moving field. As new developments arise, we hope to make updates. If you disagree with anything in the book, or if you as an industry user or marketplace provider wish to tell us about how this book compares or contrasts with your experiences with crowd work, please reach out to us at `marcua@marcua.net` and `adityagp@illinois.edu`.

data processing. We summarize these key systems (CrowdDB, Deco, and Qurk), and identify their approaches to facilitating declarative data processing.

- **Industry user survey: summary (Chapter 5)**. To get an industry perspective, we survey 13 industry users of crowd work ranging from large Fortune 500 companies to small single-purpose startups. While we find both creative and common uses, and best-practices around crowd work, we also identify several areas for future research and development. In this chapter, we describe our methodology and participants, and summarize our key findings.

- **Survey of industry users: crowd statistics and management (Chapter 6)**. Some of our participants have invested tens of millions of dollars into thousands of crowd workers and dozens of full-time employees to refine their crowd-powered data processing workflows. In this chapter, we provide summary statistics describing the scope of these operations and their management.

- **Survey of industry users: use cases and prior approaches (Chapter 7)**. To better understand the benefit of crowd work, we ask participants what their crowd-powered data processing use cases are. We also ask them to describe prior approaches, if they existed, to solving these problems.

- **Survey of industry users: task quality, worker incentives, and workflow decomposition (Chapter 8)**. We conclude our industry survey by summarizing various design and implementation decisions that participants told us about. Specifically, we summarize participants' approaches to managing quality, worker incentives, and task decomposition. One key learning was that the most advanced approaches coming out of academia do not appear to be making their way into industry.

- **Marketplace provider survey (Chapter 9)**. We survey four of the largest marketplaces that connect crowd workers and industry users to understand their view of the market. The four providers differ significantly in their methods, scope, and scale, resulting in very different use cases, approaches, and problems. We shed light on the problems facing marketplace providers, which are not always the same as those facing industry users.

1.2 Crowdsourcing Background

In this section, we describe the basic concepts underlying crowd work, and define some common terms we will use throughout the book. We follow this with a short introduction to crowdsourcing and crowdsourcing marketplaces using an example task.

1.2.1 Fundamental Concepts

There are many conflicting opinions [153] on how to define crowdsourcing, and whether crowdsourcing is indeed the same concept as *human computation*. We avoid this debate by relying on a paired definition of crowdsourcing and human computation:

> From Luis Von Ahn's Ph.D. Thesis [182]: *"Crowdsourcing (or Human Computation) is a paradigm that utilizes human processing power to solve problems that computers cannot yet solve."*

We often use crowd work instead of crowdsourcing or human computation, which also refers to the same concept: using human input to solve problems.

We now describe how we can leverage crowd work. Crowd work typically operates via *crowdsourcing marketplaces*, a market-based approach in which requesters monetarily compensate contributors (or crowds). Alternatively, *voluntary or game-based mechanisms* provide other motivating factors that incentivize human input. In this book, we focus primarily on paid market-based approaches to crowd work.

Crowdsourcing Marketplaces. There are a number of online crowdsourcing marketplaces. The canonical example of a crowdsourcing marketplace is Amazon's Mechanical Turk [10] (also referred to as MTurk for short); other examples include Samasource [14], Upwork [17], Clickworker [2], and Crowdflower [6]. There are estimated to be over 30 crowdsourcing marketplaces, and these marketplaces are growing rapidly. In addition, as we will see in subsequent sections, many large companies leverage crowdsourcing via internal crowdsourcing marketplaces, where the scenario is similar, i.e., workers get monetarily compensated for their work, but the workers are employed in-house or through contractual relationships that companies and workers establish. Note that these are not strictly crowdsourcing marketplaces in the traditional sense since these workers have longer-term relationships with companies and are paid a 9–5 wage to work on tasks.

The structure of marketplaces vary, but below, we describe one representative design that is similar to the design adopted by MTurk. There are two interfaces for accessing a typical crowdsourcing marketplace. The first is seen by *task requesters*, the second is seen by *workers*.

- The first interface is the one used by the task requesters or *task designers*—these are the individuals or teams who have tasks for which they would like to leverage crowd work. Tasks are typically introduced with a task definition or description, and often provide a form consisting of text boxes, drop-down menus, or radio buttons to elicit meaningful information from workers. Task designers design suitable tasks, and they typically specify the monetary reward or compensation associated with these tasks to be paid upon completion. Optionally, they may specify: *(a)* the *assignment*, i.e., the number of identical copies of the same task to be attempted by different individuals independently, *(b)* the amount of time allocated for that task before the task "expires," or *(c)* additional criteria (e.g., a spoken language) that individuals who want to work on these tasks must satisfy.

- The second interface is the one used by crowd workers, or simply workers, to access the entire set of tasks for which they are eligible, and to complete work on those tasks. Workers can browse the list of available tasks, pick up tasks that they wish to attempt, and work on them. In some cases, the matching or assignment to tasks is done automatically. The same task may be attempted by multiple crowd workers. If so, the workers work on tasks independently, and each one is compensated on completion of the task within the specified time limit.

Voluntary or Gaming-based Crowdsourcing. In addition to paid crowdsourcing marketplaces, there are other mechanisms by which humans are

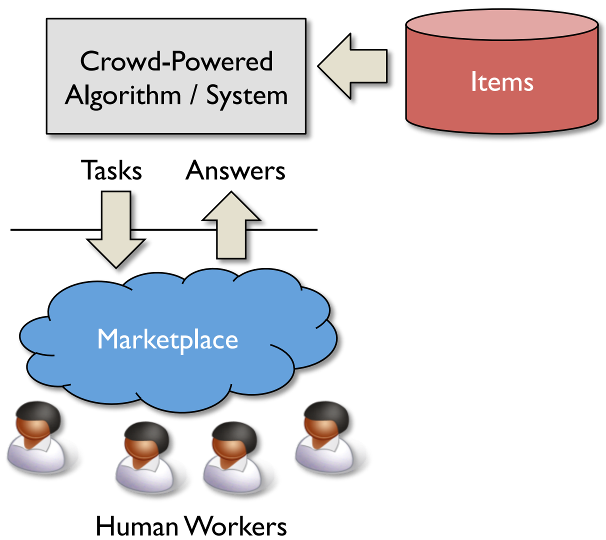

Figure 1.1: Interacting with a Marketplace

incentivized to work on tasks. One such mechanism is to solicit volunteers to work on tasks for a worthy cause. As an example, volunteers were asked to help translate tweets during the Haiti earthquake [206], or help identify galaxies in astronomical images [154, 195]. Yet another mechanism relies on games [185]. In this mechanism, people play games for fun, without realizing that the games are, in fact, tasks that need to be solved.

Even though our focus is on crowdsourcing marketplaces, the crowd-powered algorithms and systems that we talk about can also be used in conjunction with voluntary or gaming mechanisms, since there is still a limited budget of human attention that those mechanisms require that can be treated as analogous to monetary cost in crowdsourcing marketplaces.

1.2.2 Interacting with a Crowdsourcing Marketplace

We now describe how crowd-powered algorithms or systems interact with a marketplace to create tasks for crowd workers. An informal diagram of the interaction is shown in Figure 1.1. The algorithms and systems we describe operate on data items like images, videos, or text, and construct tasks to be asked to workers. These tasks are generally expressed using HTML markup

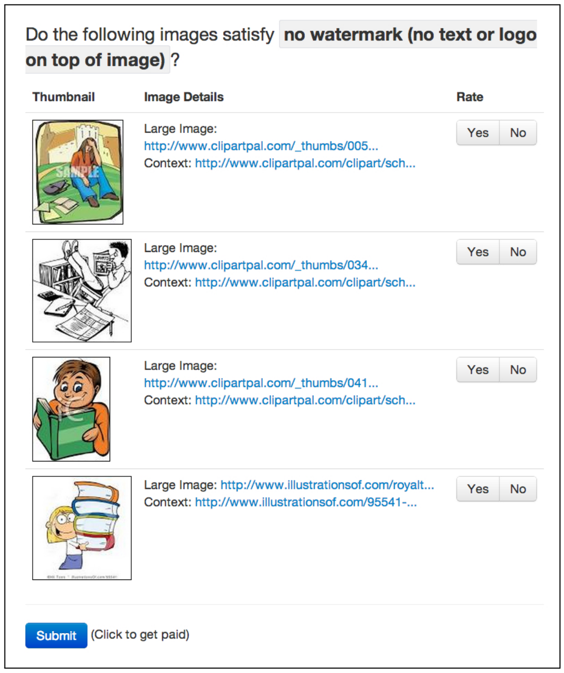

Figure 1.2: Filtering Task

for descriptions or examples, and HTML forms for input. Tasks are posted on the crowdsourcing marketplace using an API specific to the marketplace, along with worker requirements and payment policies. These tasks are answered by workers independently. Once answers to these tasks are provided back to the crowd-powered algorithm or system, the algorithm or system may choose to issue additional tasks once again, or may instead terminate.

Since workers may be concurrently working on different tasks, we can view the algorithm or system as having workers work on tasks in parallel, waiting for their responses, then having workers work on additional tasks in parallel, and so on. However, note that the system can in fact issue new tasks to the crowdsourcing marketplace before the outstanding ones are complete.

Example Tasks

We show two example tasks, as seen by workers, in Figures 1.2, and 1.3. Once a crowd worker completes either of these tasks, the worker can submit their responses to receive compensation for their effort.

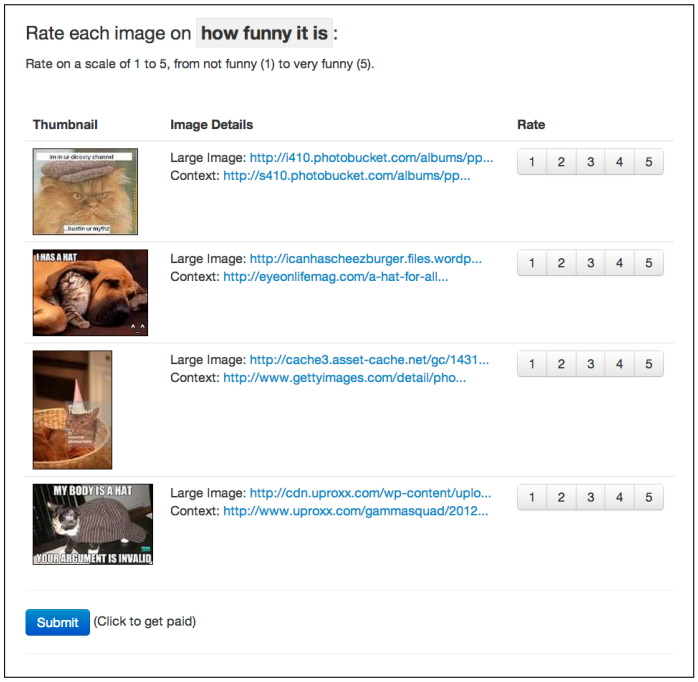

Figure 1.3: Rating Task

The first task consists of a batch of four *filtering questions*. These questions check if specific items (in this case, images) satisfy a given filtering predicate (in this case, whether they do or do not have a watermark). In this task, notice that only the last image does not have a watermark; while it is easy to make out the watermark in the first and third images, the watermark in the second image is much harder to distinguish from the rest of the image, and crowd workers may be more likely to make a mistake on this image compared to the other images. Thus, ensuring that we get correct answers for filtering questions on some items may be more difficult than others.

The second task consists of a batch of four *rating questions*, or questions requesting ratings for specific items (once again, images) for the predicate *how funny it is*. In this task, since humor is subjective, different crowd workers may have different opinions on what constitutes a funny image. Furthermore, some workers may be much more generous than others in providing high ratings. Thus, given various worker answers, inferring the true rating for each image is not trivial.

1.2.3 Terminology

There are several terms we use throughout the book; we collect them here to serve as an easy reference:

- **Crowdsourcing/Human Computation/Crowd Work**. Leveraging human processing power to solve problems that computers cannot yet solve.
- **Marketplace/Platform**. The online forum where requesters can post tasks, and workers can pick up tasks and work on them. We will use both marketplace and platform to refer to both popular forums such as Mechanical Turk (see below) or CrowdFlower, as well as in-house operations where workers work on tasks from 9–5.
- **MTurk/Mechanical Turk**. One of the popular crowdsourcing marketplaces, often used by academics.
- **Marketplace Provider**. Companies like Mechanical Turk and CrowdFlower that provide a marketplace or platform for crowdsourcing.
- **Worker/Contributor/Crowd Worker/Human Worker/Contractor**. The human being completing the task at hand.
- **Requester/Designer/Developer**. The human being or team designing and developing the task for crowd workers to complete.
- **Task definition**. The high-level description and implementation of the task being completed (e.g., *Please identify the gender of the person in each of the following images*).
- **Task/Item/Unit/Question**. A unit of work that a crowd worker must complete (e.g., *Identify the gender of the person in the following image: (image 1)*).
- **Interface**. This is the view presented to the crowd worker when they choose to work on a task. This could involve textual descriptions, as well as forms.
- **Answer/Response**. The response given by a crowd worker for a task.

- **Assignment**. A matching of a worker to a task—this may be done automatically by the marketplace, or on-demand by the workers, or on-demand by the requester. Tasks are often assigned redundantly to multiple workers.
- **Microtask**. The most popular form of task in traditional crowd work environments, in which short, relatively precise and often limited responses are allowed (e.g., multiple choice questions, yes/no questions).
- **Macrotask**. A task that is higher-level and more freeform, and takes longer to elicit a response (e.g., *Research and write up three pages on the British banking system*).
- **Reward/Compensation**. The incentive provided to the workers upon completion of the task.
- **Crowd-Powered Algorithm**. An algorithm where the unit operations are performed by crowd workers as an integral component. For example, sorting images where crowd workers compare pairs of images.
- **Crowd-Powered System**. A system or framework that uses crowd work as an integral component.
- **Latency**. The time taken by a crowd-powered algorithm or system to complete.
- **Error Rate**. The rate at which workers end up answering tasks incorrectly. This is typically a number between 0 and 1.
- **Worker Quality/Worker Accuracy**. One minus the error rate of workers. This is how often workers end up answering tasks correctly.

1.3 Crowdsourcing Best Practices

In as much as there is deep science and research behind effective crowdsourced task design, there are also some practices to follow that should provide good results. Recent work has also cataloged similar best practices specifically for information retrieval tasks [24]. Here are a few practices to follow when designing tasks:

- **Decomposition**. Break larger tasks down into smaller ones. For example, say you wish to find images of cats in a large collection of animal photos. Avoid asking workers to spend an hour searching for an example image of a cat in a stream of photos. Instead, show workers one image at a time, and ask them whether the photo contains a cat.

- **Closed-Ended, Easy to Answer Responses**. Opt for well-defined, closed-ended responses where possible, and pick interactions that make it as hard to answer a question incorrectly as it is to answer correctly. Imagine that you wish to identify the key character in a paragraph excerpted from a book. If you ask workers to fill in the name of the character in a free response text field, it is easier to leave the field empty or with unhelpful text than it is to fill in the correct character. Further, in filling in the correct character, the workers may unwittingly end up making errors. If you instead create a multiple choice interface where the characters of the book are pre-populated, selecting the key character is as simple as providing an incorrect response.

- **Instructions and Examples**. Write detailed instructions, and provide several examples. Most workers appreciate thoughtful step-by-step instructions to complete tasks correctly, and find nuanced examples helpful so that they can acclimate themselves to how you would complete various tasks. Providing a list of "do's" and "don'ts" is also helpful.

- **Debug**. After you have prototyped a task, have a colleague who is not familiar with your work complete the task. Watch them complete it and have them talk you through their understandings and actions to identify any places for improvement in your interfaces or terminology.

- **Pay Fair**. Fair pay is as critical in crowd work as it is in any other form of work. Once you have settled on a task design and implemented it, find a different colleague that has not seen the task before. Time their completion of several tasks, and from that, determine how many tasks per hour you can expect someone to complete. Keep in mind that your colleagues might have certain subject matter expertise that allow them to complete tasks faster, and be prepared to correct for poor estimates.

Based on the expected tasks completed per hour, price your tasks such that they result in a fair hourly rate. Rates differ by platform and task, but expect to pay a rate that is higher than the American minimum wage.

- **Respond to Feedback**. Either through the platform or through forums that workers use (e.g., TurkerNation [16]), seek out worker feedback and respond to it quickly. Expect to iterate on your task design and implementation as you learn from your collaboration with workers [25].

- **Manage Quality**. Because your instructions might be misleading, and because workers might make mistakes, you should expect multiple workers to answer each question/task. If the responses to the task you have created are closed-ended, send each task assignment to multiple workers and combine redundant responses. Combine their responses with simple techniques like majority voting, or more complex ones that we describe in Section 2.3.2. If instead your task is open-ended (like typing up free-response text), take multiple workers' responses and show them to a different set of workers that can identify the best responses [36]. Once you have determined which workers tend to effectively answer questions, provide them with bonuses for their good work, and offer them future work with you as a reward.

Note that much of this advice applies mostly to microtask-based work, and won't all be relevant as tasks become more complex. At a high level, iteratively testing your designs and establishing trusted relationships with crowd workers [165] will improve your experience and theirs, and this advice applies to any form of crowd work.

1.4 Assumptions in this Book

Crowdsourcing has come to encompass a large corpus of work distribution mechanisms. For the purposes of this book, we focus primarily on paid microtask-based crowd work. While our surveys and interviews touch on other areas of the design space, our primary areas of study for crowd-powerd data processing systems assume small, well-defined tasks that many workers have access to on a paid basis through a marketplace provider of crowd work.

References

[1] BitcoinReserve (Retrieved 16 March 2015). http://bitcoinreserve.org/.

[2] ClickWorker (Retrieved 22 July 2013). http://clickworker.com.

[3] ClixSense (Retrieved 16 March 2015). http://www.clixsense.com/.

[4] Cloud Me Baby (Retrieved 22 July 2013). http://www.cloudmebaby.com.

[5] CrowdFlower Content Moderation Platform (Retrieved 14 August 2013). http://crowdflower.com/type-content-moderation.

[6] CrowdFlower (Retrieved 22 July 2013). http://crowdflower.com.

[7] CSCW (Retrieved 22 July 2013). http://cscw.acm.org.

[8] Duolingo Inc. (Retrieved 14 August 2013). http://www.duolingo.com.

[9] Google, Inc. (Retrieved 14 August 2013). http://www.google.com.

[10] Mechanical Turk (Retrieved 22 July 2013). http://www.mturk.com.

[11] Microsoft Bing (Retrieved 14 August 2013). http://www.bing.com.

[12] Samasource Impact Report (Retrieved 16 March 2015). http://www.samasource.org/impact/.

[13] Samasource Jobs (Retrieved 16 March 2015). http://www.samasource.org/people/#jobs.

[14] Samasource (Retrieved 22 July 2013). http://samasource.com.

[15] Swagbucks (Retrieved 16 March 2015). http://www.swagbucks.com/.

[16] Turker Nation (Retrieved 22 July 2013). http://www.turkernation.com.

[17] Upwork (Retrieved August 4 2015). https://www.upwork.com/.

[18] Yahoo! Inc. (Retrieved 14 August 2013). http://www.yahoo.com.

[19] The Apache Pig project, April 2012. http://pig.apache.org/.

[20] Captricity website, April 2012. http://captricity.com/.

[21] Mobileworks website, April 2012. http://www.mobileworks.com/.

[22] Salman Ahmad, Alexis Battle, Zahan Malkani, and Sepandar D. Kamvar. The jabberwocky programming environment for structured social computing. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, pages 53–64, 2011.

[23] Miklós Ajtai, Vitaly Feldman, Avinatan Hassidim, and Jelani Nelson. Sorting and selection with imprecise comparisons. In *Proceedings of the International Colloquium on Automata, Languages and Programming*, pages 37–48, 2009.

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

[25] Omar Alonso, Catherine C. Marshall, and Marc Najork. Debugging a crowdsourced task with low inter-rater agreement. In *Proceedings of the 15th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 101–110, New York, NY, USA, 2015. ACM.

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

[27] Yael Amsterdamer, Yael Grossman, Tova Milo, and Pierre Senellart. Crowd mining. In *SIGMOD Conference*, pages 241–252, 2013.

[28] Arvind Arasu, Michaela Götz, and Raghav Kaushik. On active learning of record matching packages. In *SIGMOD Conference*, pages 783–794, 2010.

[29] Shivnath Babu and Jennifer Widom. Continuous queries over data streams. *ACM Sigmod Record*, 30(3):109–120, 2001.

[30] Eytan Bakshy, Jake M. Hofman, Winter A. Mason, and Duncan J. Watts. Everyone's an influencer: quantifying influence on twitter. In *Proceedings of the ACM International Conference on Web Search and Data Mining*, pages 65–74, 2011.

[31] Maria-Florina Balcan, Alina Beygelzimer, and John Langford. Agnostic active learning. *Journal of Computer and System Sciences*, 75(1):78–89, 2009.

[32] Daniel W. Barowy, Charlie Curtsinger, Emery D. Berger, and Andrew McGregor. Automan: a platform for integrating human-based and digital computation. In *Proceedings of the ACM Conference on Object-Oriented Programming Systems, Languages, and Applications*, pages 639–654, 2012.

[33] Jeff Barr and Luis-Felipe Cabrera. AI gets a brain. *ACM Queue*, 4(4):24–29, 2006.

[34] Kedar Bellare, Suresh Iyengar, Aditya G. Parameswaran, and Vibhor Rastogi. Active sampling for entity matching. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1131–1139, 2012. Online: http://doi.acm.org/10.1145/2339530.2339707.

[35] Michael S. Bernstein, Joel Brandt, Robert C. Miller, and David R. Karger. Crowds in two seconds: enabling realtime crowd-powered interfaces. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, pages 33–42, 2011.

[36] Michael S. Bernstein, Greg Little, Robert C. Miller, Björn Hartmann, Mark S. Ackerman, David R. Karger, David Crowell, and Katrina Panovich. Soylent: a word processor with a crowd inside. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, pages 313–322, 2010.

[37] Alina Beygelzimer, Sanjoy Dasgupta, and John Langford. Importance weighted active learning. In *Proceedings of the International Conference on Machine Learning*, page 7, 2009.

[38] Alina Beygelzimer, Daniel Hsu, John Langford, and Tong Zhang. Agnostic active learning without constraints. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 199–207, 2010.

[39] Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samuel White, and Tom Yeh. Vizwiz: nearly real-time answers to visual questions. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, pages 333–342, 2010.

[40] Philip Bohannon, Srujana Merugu, Cong Yu, Vipul Agarwal, Pedro DeRose, Arun Shankar Iyer, Ankur Jain, Vinay Kakade, Mridul Muralidharan, Raghu Ramakrishnan, and Warren Shen. Purple sox extraction management system. *SIGMOD Record*, 37(4):21–27, 2008.

[41] Alessandro Bozzon, Marco Brambilla, and Stefano Ceri. Answering search queries with crowdsearcher. In *Proceedings of the International World Wide Web Conference*, pages 1009–1018, 2012.

[42] Alessandro Bozzon, Marco Brambilla, Stefano Ceri, and Andrea Mauri. Reactive crowdsourcing. In *Proceedings of the International World Wide Web Conference*, pages 153–164, 2013.

[43] Alessandro Bozzon, Marco Brambilla, Stefano Ceri, Matteo Silvestri, and Giuliano Vesci. Choosing the right crowd: expert finding in social networks. In *Proceedings of the International Conference on Extending Database Technology*, pages 637–648, 2013.

[44] Jonathan Bragg, Mausam, and Daniel S. Weld. Crowdsourcing multi-label classification for taxonomy creation. In *Proceedings of the 1st AAAI Conference on Human Computation and Crowdsourcing*, 2013.

[45] Steve Branson, Catherine Wah, Boris Babenko, Florian Schroff, Peter Welinder, Pietro Perona, and Serge Belongie. Visual recognition with humans in the loop. In *European Conference on Computer Vision*, Heraklion, Crete, Sept. 2010.

[46] Nicolas Bruno. Minimizing database repros using language grammars. In *Proceedings of the International Conference on Extending Database Technology*, pages 382–393, 2010.

[47] David R. Butenhof. *Programming with POSIX Threads*. Addison-Wesley Professional, 1997.

[48] Caleb Chen Cao, Jieying She, Yongxin Tong, and Lei Chen. Whom to ask? jury selection for decision making tasks on micro-blog services. *Proceedings of the VLDB Endowment*, 5(11):1495–1506, 2012.

[49] Stuart K Card, Thomas P Moran, and Allen Newell. *The psychology of human-computer interaction*. Lawrence Erlbaum Associates, 1983.

[50] Ruggiero Cavallo and Shaili Jain. Efficient crowdsourcing contests. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems*, pages 677–686, 2012.

[51] Xiaoyong Chai, Ba-Quy Vuong, AnHai Doan, and Jeffrey F. Naughton. Efficiently incorporating user feedback into information extraction and integration programs. In *SIGMOD Conference*, pages 87–100, 2009.

[52] Dana Chandler and John Joseph Horton. Labor allocation in paid crowdsourcing: Experimental evidence on positioning, nudges and prices. In *Human Computation*, 2011.

[53] Kuan-Ta Chen, Chen-Chi Wu, Yu-Chun Chang, and Chin-Laung Lei. A crowdsourceable qoe evaluation framework for multimedia content. In *ACM Multimedia*, pages 491–500, 2009.

[54] Xi Chen, Qihang Lin, and Dengyong Zhou. Optimistic knowledge gradient policy for optimal budget allocation in crowdsourcing. In *Proceedings of the 30th International Conference on Machine Learning*, pages 64–72, 2013.

[55] Lydia B. Chilton, Greg Little, Darren Edge, Daniel S. Weld, and James A. Landay. Cascade: crowdsourcing taxonomy creation. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 1999–2008, 2013.

[56] David A. Cohn, Les E. Atlas, and Richard E. Ladner. Improving generalization with active learning. *Machine Learning*, 15(2):201–221, 1994.

[57] Seth Cooper, Firas Khatib, Adrien Treuille, Janos Barbero, Jeehyung Lee, Michael Beenen, Andrew Leaver-Fay, David Baker, Zoran Popović, and Foldit players. Predicting protein structures with a multiplayer online game. *Nature*, 466(7307):756–760, 2010.

[58] Peng Dai, Christopher H. Lin, Mausam, and Daniel S. Weld. Pomdp-based control of workflows for crowdsourcing. *Artif. Intell.*, 202:52–85, 2013.

[59] Nilesh Dalvi, Anirban Dasgupta, Ravi Kumar, and Vibhor Rastogi. Aggregating crowdsourced binary ratings. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 285–294. International World Wide Web Conferences Steering Committee, 2013.

[60] Nilesh Dalvi, Ravi Kumar, Bo Pang, Raghu Ramakrishnan, Andrew Tomkins, Philip Bohannon, Sathiya Keerthi, and Srujana Merugu. A web of concepts. In *Symposium on Principles of Database Systems*, pages 1–12, 2009.

[61] Nilesh Dalvi, Aditya G. Parameswaran, and Vibhor Rastogi. Minimizing uncertainty in pipelines. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 2951–2959, 2012.

[62] Nilesh Dalvi and Dan Suciu. Efficient query evaluation on probabilistic databases. *Very Large Data Base Journal (VLDBJ)*, 16(4):523–544, 2007.

[63] Sanjoy Dasgupta, Daniel Hsu, and Claire Monteleoni. A general agnostic active learning algorithm. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, 2007.

[64] Sanjoy Dasgupta and John Langford. Tutorial summary: Active learning. In *Proceedings of the International Conference on Machine Learning*, page 178, 2009.

[65] Susan B. Davidson, Sanjeev Khanna, Tova Milo, and Sudeepa Roy. Using the crowd for top-k and group-by queries. In *Proceedings of the International Conference on Database Theory*, pages 225–236, 2013.

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

[67] Jeffrey Dean and Sanjay Ghemawat. Mapreduce: Simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, January 2008.

[68] Gianluca Demartini, Djellel Eddine Difallah, and Philippe Cudré-Mauroux. ZenCrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In *WWW '12: Proceedings of the 21st International Conference on World Wide Web*. ACM, April 2012.

[69] Gianluca Demartini, Beth Trushkowsky, Tim Kraska, and Michael J. Franklin. Crowdq: Crowdsourced query understanding. In *Proceedings of the Conference on Innovative Data Systems Research*, 2013.

[70] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38, 1977.

[71] Amol Deshpande and Samuel Madden. Mauvedb: supporting model-based user views in database systems. In *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data*, pages 73–84, New York, NY, USA, 2006. ACM.

[72] Daniel Deutch, Ohad Greenshpan, Boris Kostenko, and Tova Milo. Using markov chain monte carlo to play trivia. In *International Conference on Data Engineering*, pages 1308–1311, 2011.

[73] AnHai Doan, Michael J. Franklin, Donald Kossmann, and Tim Kraska. Crowdsourcing Applications and Platforms: A Data Management Perspective. *Proceedings of the VLDB Endowment*, 4(12):1508–1509, 2011.

[74] AnHai Doan, Raghu Ramakrishnan, and Alon Y. Halevy. Crowdsourcing systems on the world-wide web. *Communications of the ACM*, 54(4):86–96, 2011.

[75] Pinar Donmez, Jaime G. Carbonell, and Jeff G. Schneider. Efficiently learning the accuracy of labeling sources for selective sampling. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 259–268, 2009.

[76] John R. Douceur. The Sybil Attack. In *Proceedings of the International Workshop of Peer-to-Peer Systems*, pages 251–260, 2002.

[77] Steven Dow, Anand Pramod Kulkarni, Scott R. Klemmer, and Björn Hartmann. Shepherding the crowd yields better work. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, pages 1013–1022, 2012.

[78] Ahmed K. Elmagarmid, Panagiotis G. Ipeirotis, and Vassilios S. Verykios. Duplicate Record Detection: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 19(1):1–16, 2007.

[79] Amber Feng, Michael J. Franklin, Donald Kossmann, Tim Kraska, Samuel Madden, Sukriti Ramesh, Andrew Wang, and Reynold Xin. CrowdDB: Query Processing with the VLDB Crowd. *Proceedings of the VLDB Endowment*, 4(12):1387–1390, 2011.

[80] Michael J. Franklin, Donald Kossmann, Tim Kraska, Sukriti Ramesh, and Reynold Xin. CrowdDB: answering queries with crowdsourcing. In *SIGMOD Conference*, pages 61–72, 2011.

[81] Yihan Gao and Aditya G. Parameswaran. Finish them!: Pricing algorithms for human computation. *Proceedings of the VLDB Endowment*, 7(14):1965–1976, 2014.

[82] Arpita Ghosh. Game theory and incentives in human computation systems. In *Handbook of Human Computation*. Springer, 2013.

[83] Arpita Ghosh, Satyen Kale, and Preston McAfee. Who Moderates the Moderators? Crowdsourcing Abuse Detection in User-Generated Content. In *Proceedings of the ACM Conference on Economics and Computation*, pages 167–176, 2011.

[84] Arpita Ghosh and R. Preston McAfee. Crowdsourcing with endogenous entry. In *Proceedings of the International World Wide Web Conference*, pages 999–1008, 2012.

[85] Chaitanya Gokhale, Sanjib Das, AnHai Doan, Jeffrey F. Naughton, Narasimhan Rampalli, Jude Shavlik, and Xiaojin Zhu. Corleone: Hands-Off Crowdsourcing for Entity Matching. *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, March 2014.

[86] Ryan Gomes, Peter Welinder, Andreas Krause, and Pietro Perona. Crowdclustering. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 558–566, 2011.

[87] Pankaj Gulhane, Amit Madaan, Rupesh R. Mehta, Jeyashankher Ramamirtham, Rajeev Rastogi, Sandeepkumar Satpal, Srinivasan H. Sengamedu, Ashwin Tengli, and Charu Tiwari. Web-scale information extraction with vertex. In *International Conference on Data Engineering*, pages 1209–1220, 2011.

[88] Pankaj Gulhane, Rajeev Rastogi, Srinivasan H. Sengamedu, and Ashwin Tengli. Exploiting content redundancy for web information extraction. *Proceedings of the VLDB Endowment*, 3(1):578–587, 2010.

[89] Stephen Guo, Aditya G. Parameswaran, and Hector Garcia-Molina. So who won?: dynamic max discovery with the crowd. In *SIGMOD Conference*, pages 385–396, 2012. Online: http://doi.acm.org/10.1145/2213836.2213880.

[90] Maya R. Gupta and Yihua Chen. Theory and use of the em algorithm. *Foundations and Trends in Signal Processing*, 4(3):223–296, 2010.

[91] Daniel Haas, Jason Ansel, Lydia Gu, and Adam Marcus. Argonaut: Macrotask Crowdsourcing for Complex Data Processing. *Proceedings of the VLDB Endowment*, 8(12):1642–1653, 2015.

[92] Steve Hanneke. A bound on the label complexity of agnostic active learning. In *Proceedings of the International Conference on Machine Learning*, pages 353–360, 2007.

[93] Björn Hartmann and Panagiotis G. Ipeirotis. What's the right price? pricing tasks for finishing on time. 2011.

[94] Hannes Heikinheimo and Antti Ukkonen. The Crowd-Median Algorithm. In *Proceedings of the 1st AAAI Conference on Human Computation and Crowdsourcing*, 2013.

[95] Paul Heymann and Hector Garcia-Molina. Turkalytics: analytics for human computation. In *Proceedings of the 20th international conference on World wide web*, pages 477–486, 2011.

[96] John Joseph Horton and Lydia B. Chilton. The labor economics of paid crowdsourcing. In *ACM Conference on Electronic Commerce*, pages 209–218, 2010.

[97] Eric Huang, Haoqi Zhang, David C. Parkes, Krzysztof Z. Gajos, and Yiling Chen. Toward automatic task design: a progress report. In *Proceedings of the Conference on Human Computation and Crowdsourcing*, New York, NY, USA, 2010.

[98] Panagiotis G. Ipeirotis. Analyzing the amazon mechanical turk marketplace. *XRDS: Crossroads*, 17:16–21, December 2010.

[99] Panagiotis G. Ipeirotis. Demographics of mechanical turk. Technical report, March 2010.

[100] Panagiotis G. Ipeirotis, Foster Provost, and Jing Wang. Quality management on amazon mechanical turk. In *Proceedings of the Conference on Human Computation and Crowdsourcing*, New York, NY, USA, 2010.

[101] Shaili Jain, Yiling Chen, and David C. Parkes. Designing incentives for online question and answer forums. In *ACM Conference on Electronic Commerce*, pages 129–138, 2009.

[102] Shaili Jain and David C. Parkes. A Game-Theoretic Analysis of Games with a Purpose. In *Proceedings of the Conference on Web and Internet Economics*, pages 342–350, 2008.

[103] Shaili Jain and David C. Parkes. The role of game theory in human computation systems. In *Proceedings of the Conference on Human Computation and Crowdsourcing*, pages 58–61, 2009.

[104] Shawn R. Jeffery, Michael J. Franklin, and Alon Y. Halevy. Pay-as-you-go user feedback for dataspace systems. In *SIGMOD Conference*, pages 847–860, 2008.

[105] Shawn R. Jeffery, Liwen Sun, Matt DeLand, Nick Pendar, Rick Barber, and Andrew Galdi. Arnold: Declarative crowd-machine data integration. In *Proceedings of the Conference on Innovative Data Systems Research*, 2013.

[106] Manas Joglekar, Hector Garcia-Molina, and Aditya Parameswaran. Evaluating the Crowd with Confidence. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2013. Online: http://dl.acm.org/citation.cfm?id=2487575.2487595.

[107] Manas Joglekar, Hector Garcia-Molina, and Aditya Parameswaran. Evaluating the crowd with confidence. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 686–694. ACM, 2013.

[108] Manas Joglekar, Hector Garcia-Molina, and Aditya Parameswaran. Comprehensive and Reliable Crowd Assessment Algorithms. In *International Conference on Data Engineering*, 2015.

[109] Ece Kamar, Severin Hacker, and Eric Horvitz. Combining human and machine intelligence in large-scale crowdsourcing. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems*, pages 467–474, 2012.

[110] Ece Kamar and Eric Horvitz. Incentives for truthful reporting in crowdsourcing. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems*, pages 1329–1330, 2012.

[111] Nikos Karampatziakis and John Langford. Online importance weight aware updates. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 392–399, 2011.

[112] David R. Karger, Sewoong Oh, and Devavrat Shah. Iterative learning for reliable crowdsourcing systems. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 1953–1961, 2011.

[113] David R. Karger, Sewoong Oh, and Devavrat Shah. Effcient crowdsourcing for multi-class labeling. In *Special Interest Group on Measurement and Evaluation (SIGMETRICS)*, pages 81–92, 2013.

[114] Juho Kim, Haoqi Zhang, Paul André, Lydia B. Chilton, Anant Bhardwaj, David Karger, Steven P. Dow, and Robert C. Miller. Cobi: Community-Informed Conference Scheduling. In *1st AAAI Conference on Human Computation and Crowdsourcing*, 2013.

[115] Aniket Kittur, Ed H. Chi, and Bongwon Suh. Crowdsourcing user studies with Mechanical Turk. In *the ACM Conference on Human Factors in Computing Systems*, pages 453–456, 2008.

[116] Aniket Kittur, Boris Smus, and Robert Kraut. CrowdForge: crowdsourcing complex work. In *The ACM Conference on Human Factors in Computing Systems, Extended Abstracts*, pages 1801–1806, 2011.

[117] Shailesh Kochhar, Stefano Mazzocchi, and Praveen Paritosh. The anatomy of a large-scale human computation engine. In *Proceedings of the Conference on Human Computation and Crowdsourcing*, New York, NY, USA, 2010.

[118] Anand Kulkarni, Philipp Gutheim, Prayag Narula, David Rolnitzky, Tapan S. Parikh, and Björn Hartmann. MobileWorks: Designing for Quality in a Managed Crowdsourcing Architecture. *IEEE Internet Computing*, 16(5):28–35, 2012.

[119] Anand Pramod Kulkarni, Matthew Can, and Björn Hartmann. Collaboratively crowdsourcing workflows with turkomatic. *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, pages 1003–1012, 2012.

[120] Ray Kurzweil. *The Singularity Is Near: When Humans Transcend Biology*. Penguin (Non-Classics), 2006.

[121] Walter S. Lasecki, Kyle I. Murray, Samuel White, Robert C. Miller, and Jeffrey P. Bigham. Real-time crowd control of existing interfaces. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, pages 23–32, 2011.

[122] Edith Law and Luis von Ahn. *Human Computation*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2011.

[123] David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. In *SIGIR Conference*, pages 3–12, 1994.

[124] Christopher H. Lin, Mausam, and Daniel S. Weld. Crowdsourcing control: Moving beyond multiple choice. In *UAI*, pages 491–500, 2012.

[125] Christopher H. Lin, Mausam, and Daniel S. Weld. Dynamically switching between synergistic workflows for crowdsourcing. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, 2012.

[126] Greg Little, Lydia B. Chilton, Max Goldman, and Robert C. Miller. Turkit: tools for iterative tasks on mechanical turk. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Workshop on Human Computation*, pages 29–30, 2009.

[127] Qiang Liu, Jian Peng, and Alexander Ihler. Variational inference for crowdsourcing. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 701–709, 2012.

[128] 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, 2012.

[129] Ilia Lotosh, Tova Milo, and Slava Novgorodov. CrowdPlanr: Planning made easy with crowd. In *International Conference on Data Engineering*, pages 1344–1347, 2013.

[130] Adam Marcus, David R. Karger, Samuel Madden, Rob Miller, and Sewoong Oh. Counting with the crowd. *Proceedings of the VLDB Endowment*, 6(2):109–120, 2012.

[131] Adam Marcus, Eugene Wu, David R. Karger, Samuel Madden, and Robert C. Miller. Demonstration of qurk: a query processor for humanoperators. In *SIGMOD Conference*, pages 1315–1318, 2011.

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

[133] Adam Marcus, Eugene Wu, Samuel Madden, and Robert C. Miller. Crowdsourced databases: Query processing with people. In *Proceedings of the Conference on Innovative Data Systems Research*, pages 211–214, 2011.

[134] Winter A. Mason and Duncan J. Watts. Financial incentives and the "performance of crowds". In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Workshop on Human Computation*, pages 77–85, 2009.

[135] Geoffrey J. McLachlan and Thriyambakam Krishnan. *The EM Algorithm and Extensions (Wiley Series in Probability and Statistics)*. Wiley-Interscience, 2 edition, March 2008.

[136] Paul Mineiro. Cost-Sensitive Binary Classification and Active Learning, 2012. http://www.machinedlearnings.com/2012/01/cost-sensitive-binary-classification.html.

[137] Marti Motoyama, Kirill Levchenko, Chris Kanich, Damon McCoy, Geoffrey M. Voelker, and Stefan Savage. Re: Captchas-understanding captcha-solving services in an economic context. In *USENIX Security Symposium*, pages 435–462, 2010.

[138] Barzan Mozafari, Purnamrita Sarkar, Michael J. Franklin, Michael I. Jordan, and Samuel Madden. Active learning for crowd-sourced databases. *arXiv preprint arXiv:1209.3686*, 2012.

[139] Jon Noronha, Eric Hysen, Haoqi Zhang, and Krzysztof Z. Gajos. Platemate: crowdsourcing nutritional analysis from food photographs. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, pages 1–12, 2011.

[140] Christopher Olston, Benjamin Reed, Utkarsh Srivastava, Ravi Kumar, and Andrew Tomkins. Pig latin: a not-so-foreign language for data processing. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, New York, NY, USA, 2008.

[141] Aditya Parameswaran, Stephen Boyd, Hector Garcia-Molina, Ashish Gupta, Neoklis Polyzotis, and Jennifer Widom. Optimal crowd-powered rating and filtering algorithms. Technical report, Stanford University, 2013.

[142] Aditya G. Parameswaran, Nilesh Dalvi, Hector Garcia-Molina, and Rajeev Rastogi. Optimal schemes for robust web extraction. *Proceedings of the VLDB Endowment*, 4(11):980–991, 2011.

[143] Aditya G. Parameswaran, Hector Garcia-Molina, Hyunjung Park, Neoklis Polyzotis, Aditya Ramesh, and Jennifer Widom. Crowdscreen: algorithms for filtering data with humans. In *SIGMOD Conference*, pages 361–372, 2012. Online: http://doi.acm.org/10.1145/2213836.2213878.

[144] Aditya G. Parameswaran, Hyunjung Park, Hector Garcia-Molina, Neoklis Polyzotis, and Jennifer Widom. Deco: declarative crowdsourcing. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, pages 1203–1212, 2012.

[145] Aditya G. Parameswaran and Neoklis Polyzotis. Answering Queries using Humans, Algorithms and Databases. In *Proceedings of the Conference on Innovative Data Systems Research*, pages 160–166, 2011.

[146] Aditya G. Parameswaran, Anish Das Sarma, Hector Garcia-Molina, Neoklis Polyzotis, and Jennifer Widom. Human-assisted graph search: it's okay to ask questions. *Proceedings of the VLDB Endowment*, 4(5):267–278, 2011.

[147] Aditya G. Parameswaran, Ming Han Teh, Hector Garcia-Molina, and Jennifer Widom. DataSift: An Expressive and Accurate Crowd-Powered Search Toolkit. In *Proceedings of the 1st AAAI Conference on Human Computation and Crowdsourcing*, 2013.

[148] Aditya G. Parameswaran, Ming Han Teh, Hector Garcia-Molina, and Jennifer Widom. DataSift: a crowd-powered search toolkit. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, pages 885–888, 2014.

[149] Hyunjung Park, Richard Pang, Aditya G. Parameswaran, Hector Garcia-Molina, Neoklis Polyzotis, and Jennifer Widom. Deco: A system for declarative crowdsourcing. *Proceedings of the VLDB Endowment*, 5(12):1990–1993, 2012.

[150] Hyunjung Park, Aditya Parameswaran, and Jennifer Widom. Query processing over crowdsourced data. Technical report, Stanford University, September 2012.

[151] Hyunjung Park and Jennifer Widom. CrowdFill: collecting structured data from the crowd. *SIGMOD Conference*, pages 577–588, 2014.

[152] Vassilis Polychronopoulos, Luca de Alfaro, James Davis, Hector Garcia-Molina, and Neoklis Polyzotis. Human-powered top-k lists. In *Proceedings of the 16th International Workshop on the Web and Databases*, pages 25–30, 2013.

[153] Alexander J. Quinn and Benjamin B. Bederson. Human computation: a survey and taxonomy of a growing field. In *The ACM Conference on Human Factors in Computing Systems*, pages 1403–1412, 2011.

[154] M. Jordan Raddick, Georgia Bracey, Pamela L. Gay, Chris J. Lintott, Phil Murray, Kevin Schawinski, Alexander S. Szalay, and Jan Vandenberg. Galaxy zoo: Exploring the motivations of citizen science volunteers. September 2009. http://arxiv.org/abs/0909.2925.

[155] A. Ramesh, A. Parameswaran, H. Garcia-Molina, and N. Polyzotis. Identifying reliable workers swiftly. Technical report, Stanford University, September 2012.

[156] Vikas C. Raykar and Shipeng Yu. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *Journal of Machine Learning Research*, 13:491–518, 2012.

[157] Vikas C. Raykar, Shipeng Yu, Linda H. Zhao, Anna K. Jerebko, Charles Florin, Gerardo Hermosillo Valadez, Luca Bogoni, and Linda Moy. Supervised learning from multiple experts: whom to trust when everyone lies a bit. In *Proceedings of the International Conference on Machine Learning*, page 112, 2009.

[158] T. Rekatsinas, A. Deshpande, and A. Parameswaran. CrowdGather: Entity Extraction over Structured Domains. *ArXiv e-prints*, February 2015.

[159] Daniela Retelny, Sébastien Robaszkiewicz, Alexandra To, Walter S. Lasecki, Jay Patel, Negar Rahmati, Tulsee Doshi, Melissa Valentine, and Michael S. Bernstein. Expert crowdsourcing with flash teams. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 75–85, New York, NY, USA, 2014. ACM.

[160] Flavio P. Ribeiro, Dinei A. F. Florêncio, and Vitor H. Nascimento. Crowdsourcing subjective image quality evaluation. In *Proceedings of the International Conference on Image Processing*, pages 3097–3100, 2011.

[161] Joel Ross, Lilly C. Irani, M. Six Silberman, Andrew Zaldivar, and Bill Tomlinson. Who are the crowdworkers?: shifting demographics in mechanical turk. In *The ACM Conference on Human Factors in Computing Systems, Extended Abstracts*. ACM, 2010.

[162] Stuart J. Russell and Peter Norvig. *Artificial Intelligence - A Modern Approach (3. internat. ed.)*. Pearson Education, 2010.

[163] Jeffrey M. Rzeszotarski and Aniket Kittur. Instrumenting the crowd: using implicit behavioral measures to predict task performance. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*. ACM Request Permissions, October 2011.

[164] Jeffrey M. Rzeszotarski and Aniket Kittur. Crowdscape: interactively visualizing user behavior and output. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, pages 55–62, 2012.

[165] Niloufar Salehi, Lilly C. Irani, and Michael S. Bernstein. We Are Dynamo: Overcoming Stalling and Friction in Collective Action for Crowd Workers. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 1621–1630. ACM, 2015.

[166] Akash Das Sarma, Ayush Jain, Arnab Nandi, Aditya G. Parameswaran, and Jennifer Widom. Surpassing humans and computers with JELLYBEAN: crowd-vision-hybrid counting algorithms. Technical report, 2015.

[167] Anish Das Sarma, Aditya Parameswaran, Hector Garcia-Molina, and Alon Halevy. Crowd-powered find algorithms. In *International Conference on Data Engineering*, 2014.

[168] Lauren Schmidt. Crowdsourcing for human subjects research. *Proceedings of CrowdConf*, 2010.

[169] Burr Settles. *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2012.

[170] H. Sebastian Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Proceedings of the Conference On Learning Theory*, pages 287–294, 1992.

[171] Victor S. Sheng, Foster J. Provost, and Panagiotis G. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 614–622, 2008.

[172] 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 *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, pages 254–263, 2008.

[173] Michael Stonebraker, Daniel Bruckner, Ihab F Ilyas, George Beskales, Mitch Cherniack, Stanley B. Zdonik, Alexander Pagan, and Shan Xu. Data Curation at Scale: The Data Tamer System. *Proceedings of the Conference on Innovative Data Systems Research*, 2013.

[174] Siddharth Suri, Daniel G. Goldstein, and Winter A. Mason. Honesty in an online labor market. In *Human Computation*, 2011.

[175] Omer Tamuz, Ce Liu, Serge Belongie, Ohad Shamir, and Adam Kalai. Adaptively learning the crowd kernel. In *Proceedings of the International Conference on Machine Learning*, pages 673–680, 2011.

[176] The Economist. The data deluge, February 2010.

[177] Michael Toomim, Travis Kriplean, Claus Pörtner, and James A. Landay. Utility of human-computer interactions: toward a science of preference measurement. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 2275–2284, 2011.

[178] Emma Tosch and Emery D. Berger. Surveyman: programming and automatically debugging surveys. In *Proceedings of the 2014 ACM International Conference on Object Oriented Programming Systems Languages & Applications*, pages 197–211, 2014.

[179] Beth Trushkowsky, Tim Kraska, Michael J. Franklin, and Purnamrita Sarkar. Crowdsourced enumeration queries. In *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*, pages 673–684, 2013.

[180] Petros Venetis and Hector Garcia-Molina. Dynamic max algorithms in crowdsourcing environments. Technical report, Stanford University, August 2012.

[181] Petros Venetis, Hector Garcia-Molina, Kerui Huang, and Neoklis Polyzotis. Max algorithms in crowdsourcing environments. In *Proceedings of the 21st International World Wide Web Conference 2012*, pages 989–998, 2012.

[182] Luis von Ahn. *Human computation*. PhD thesis, Pittsburgh, PA, USA, 2005.

[183] Luis von Ahn, Manuel Blum, Nicholas J. Hopper, and John Langford. CAPTCHA: Using Hard AI Problems for Security. In *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 294–311, 2003.

[184] Luis von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 319–326, 2004.

[185] Luis von Ahn and Laura Dabbish. Designing games with a purpose. *Communications of the ACM*, 51(8):58–67, 2008.

[186] Luis von Ahn, Ruoran Liu, and Manuel Blum. Peekaboom: a game for locating objects in images. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 55–64, 2006.

[187] B. Walczak and D.L. Massart. Dealing with missing data: Part II. *Chemometrics and Intelligent Laboratory Systems*, 58(1):29 – 42, 2001.

[188] Daisy Zhe Wang, Eirinaios Michelakis, Minos Garofalakis, and Joseph M. Hellerstein. BayesStore: managing large, uncertain data repositories with probabilistic graphical models. *Proceedings of the VLDB Endowment*, 1(1):340–351, 2008.

[189] Jiannan Wang, Tim Kraska, Michael J. Franklin, and Jianhua Feng. CrowdER: Crowdsourcing Entity Resolution. *Proceedings of the VLDB Endowment*, 2012.

[190] Qinqin Wang, Patrick Cavanagh, and Marc Green. Familiarity and pop-out in visual search. *Perception & Psychophysics*, 56(5):495–500, 1994.

[191] P. Welinder and P. Perona. Online crowdsourcing: rating annotators and obtaining cost-effective labels. In *CVPR Conference*, 2010.

[192] Steven Euijong Whang, Peter Lofgren, and Hector Garcia-Molina. Question selection for crowd entity resolution. In *Proceedings of the VLDB Endowment*. VLDB Endowment, April 2013.

[193] Jacob Whitehill, Paul Ruvolo, Tingfan Wu, Jacob Bergsma, and Javier R. Movellan. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 2035–2043. 2009.

[194] Jennifer Widom. Trio: A system for integrated management of data, accuracy, and lineage. In *Proceedings of the Conference on Innovative Data Systems Research*, pages 262–276, 2005.

[195] Wikipedia. Citizen science — wikipedia, the free encyclopedia, 2013. [Online; accessed 22-July-2013].

[196] Wikipedia. Unstructured data — Wikipedia, the free encyclopedia, 2013. [Online; accessed 6-July-2013].

[197] Jeremy M. Wolfe and H. Pashler (Editor). *Attention*, chapter Visual Search, pages 13–73. University College London Press, 1998.

[198] Bianca Zadrozny, John Langford, and Naoki Abe. Cost-sensitive learning by cost-proportionate example weighting. In *Proceedings of the 3rd IEEE International Conference on Data Mining*, Washington, DC, USA, 2003.

[199] Omar Zaidan and Chris Callison-Burch. Feasibility of human-in-the-loop minimum error rate training. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, pages 52–61, 2009.

[200] Haoqi Zhang, Edith Law, Rob Miller, Krzysztof Gajos, David C. Parkes, and Eric Horvitz. Human computation tasks with global constraints. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 217–226, 2012.

[201] Yuchen Zhang, Xi Chen, Dengyong Zhou, and Michael I. Jordan. Spectral Methods meet EM: A Provably Optimal Algorithm for Crowdsourcing. *arXiv preprint arXiv:1406.3824*, 2014.

[202] Dengyong Zhou, Sumit Basu, Yi Mao, and John C. Platt. Learning from the wisdom of crowds by minimax entropy. In *Advances in Neural Information Processing Systems*, pages 2195–2203, 2012.

[203] Dengyong Zhou, Qiang Liu, John C. Platt, and Christopher Meek. Aggregating Ordinal Labels from Crowds by Minimax Conditional Entropy. In *Proceedings of the 31st International Conference on Machine Learning*, June 2014.

[204] Honglei Zhuang, Aditya G. Parameswaran, Dan Roth, and Jiawei Han. Debiasing crowdsourced batches. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*, pages 1593–1602, 2015.

[205] Paul C. Zikopoulos, Chris Eaton, Dirk deRoos, Thomas Deutsch, and George Lapis. *Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data*. McGraw-Hill Osborne Media, 1st edition, 2011.

[206] Matthew Zook, Mark Graham, Taylor Shelton, and Sean Gorman. Volunteered geographic information and crowdsourcing disaster relief: a case study of the haitian earthquake. *World Medical & Health Policy*, 2(2):7–33, 2010.

[207] Nathan Zukoff. Demographics of the Largest On-demand Workforce (Retrieved 16 March 2015), 2014. http://www.crowdflower.com/blog/2014/01/demographics-of-the-largest-on-demand-workforce.