Lecture 24:
CS 5306 / INFO 5306:
Crowdsourcing and
Human Computation

Amazon Mechanical Turk Reimbursements: Use Form from Course Webpage Submit to: Janeen Orr, Gates 236

Payment/Reimbursement Cover Form
COE CIS FTC
Attach to: Invoices, receipts, etc.

Please Check the box below that is appropriate for the action you are requesting and include all necessary forms

- [x] Reimbursement (Non-travel)
- [ ] Direct Payment (attach invoice)
- [ ] Petty Cash Replenishment
- [ ] Speaker fees (to include Honorariums)*
- [ ] Wire Transfer (attach invoice)
- [ ] Awards/Prizes

**(If visiting speaker, lecturer, etc., obtain W9 (US vendors)/W8Ben (International); fax to 607-255-9786 and shred the original.)*

- [ ] *Completed W9/W8-BEN has been faxed to FTC* Date:
- [ ] Requesting FTC to obtain W9/W8-BEN (see contact information below)

Payee: ______ NetID: ______

Address: ______

account number (Required)	sub-account	object code	sub-object	Project	org ref ID	Amount
F563405						20.00

Alcohol? If Yes, provide unrestricted account:

account number (Required)	sub-account	object code	sub-object	Project	org ref ID	Amount

Business/Research Purpose: Be specific.

Who, What, When, Where, Why:
Fees generated from Mechanical Turk for final project for CS/INFO 5306

i.e.: payment for speaker, Dr. George Smith, MIT, who presented a colloquium at the Theory Seminar on 1/15/12. (flyer attached)

If **meal** - list individual attendees or group name: staple attendance sheet for group meetings (if available) Please provide **attendees** full name and affiliation

NOTE: please provide itemized meal receipts

Preparer: ______ NetID: ______ Date: ______
(Individual Submitting form)

Department/Unit use only
Account Authorization: ______ NetID: ______ Date: ______

Print this form and attach to credit card receipt/invoice/etc. and forward to your Department/Unit. Department/Unit will forward to FTC Representative, 484 Rhodes Hall.

Print Form

11/17/11

Looking Ahead

Looking Ahead

Populating the space of examples

Embracing Error to Enable Rapid Crowdsourcing

Ranjay Krishna[1], Kenji Hata[1], Stephanie Chen[1], Joshua Kravitz[1], David A. Shamma[2], Li Fei-Fei[1], Michael S. Bernstein[1]
Stanford University[1], Yahoo! Labs[2]
{ranjaykrishna, kenjihata, stephchen, kravitzj, feifeili, msb}@cs.stanford.edu, aymans@acm.org

ABSTRACT

Microtask crowdsourcing has enabled dataset advances in social science and machine learning, but existing crowdsourcing schemes are too expensive to scale up with the expanding volume of data. To scale and widen the applicability of crowdsourcing, we present a technique that produces extremely rapid judgments for binary and categorical labels. Rather than punishing all errors, which causes workers to proceed slowly and deliberately, our technique speeds up workers' judgments to the point where errors are acceptable and even expected. We demonstrate that it is possible to rectify these errors by randomizing task order and modeling response latency. We evaluate our technique on a breadth of common labeling tasks such as image verification, word similarity, sentiment analysis and topic classification. Where prior work typically achieves a 0.25× to 1× speedup over fixed majority vote, our approach often achieves an order of magnitude (10×) speedup.

Author Keywords

Human computation; Crowdsourcing; RSVP

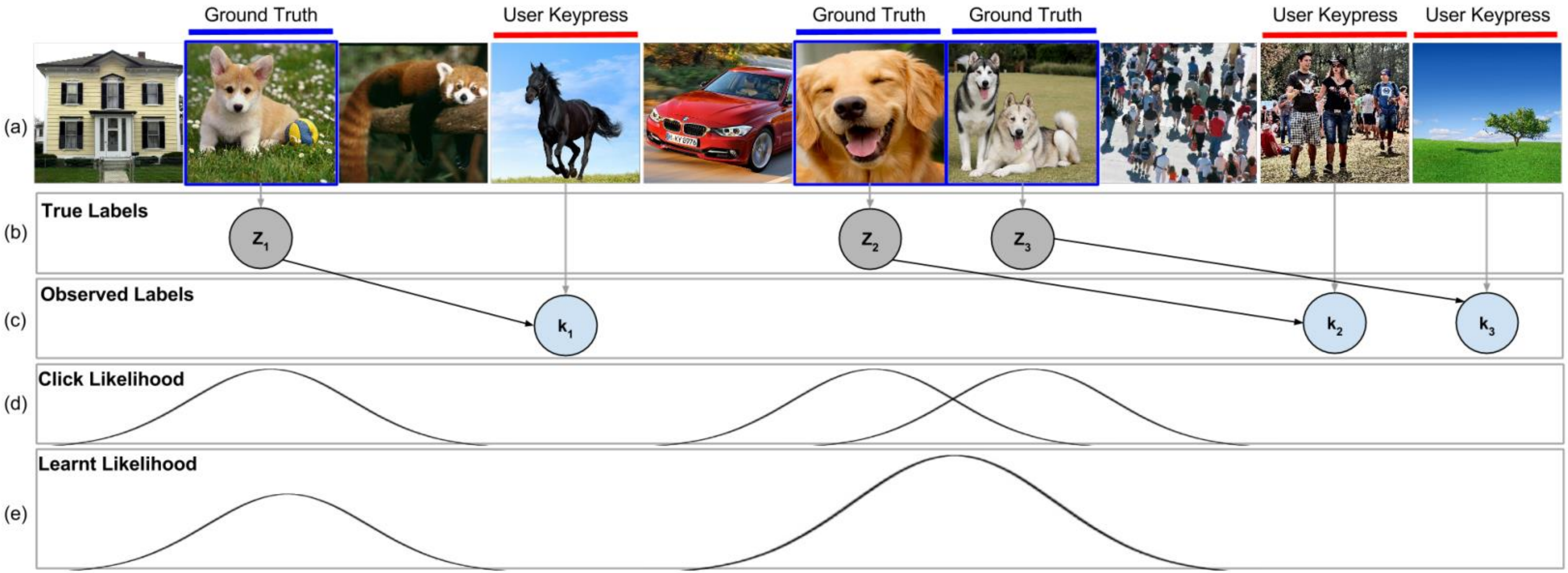
ACM Classification Keywords

H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

However, even microtask crowdsourcing can be insufficiently scalable, and it remains too expensive for use in the production of many industry-size datasets [24]. Cost is bound to the amount of work completed per minute of effort, and existing techniques for speeding up labeling (reducing the amount of required effort) are not scaling as quickly as the volume of data we are now producing that must be labeled [63]. To expand the applicability of crowdsourcing, the number of items annotated per minute of effort needs to increase substantially.

In this paper, we focus on one of the most common classes of crowdsourcing tasks [20]: binary annotation. These tasks are yes-or-no questions, typically identifying whether or not an input has a specific characteristic. Examples of these types of tasks are topic categorization (e.g., "Is this article about finance?") [52], image classification (e.g., "Is this a dog?") [13, 38, 36], audio styles [53] and emotion detection [36] in songs (e.g., "Is the music calm and soothing?"), word similarity (e.g., "Are *shipment* and *cargo* synonyms?") [42] and sentiment analysis (e.g., "Is this tweet positive?") [43].

Previous methods have sped up binary classification tasks by minimizing worker error. A central assumption behind this prior work has been that workers make errors because they are not trying hard enough (e.g., "a lack of expertise, dedication [or] interest" [54]). Platforms thus punish errors harshly, for example by denying payment. Current methods calculate the minimum redundancy necessary to be confident that



Ground Truth
User Keypress
Ground Truth
Ground Truth
User Keypress
User Keypress
(a)
(b)
True Labels
z_1
z_2
z_3
(c)
Observed Labels
k_1
k_2
k_3
(d)
Click Likelihood
(e)
Learnt Likelihood

The Knowledge Accelerator: Big Picture Thinking in Small Pieces

1st Author Name
Affiliation
City, Country
e-mail address

2nd Author Name
Affiliation
City, Country
e-mail address

3rd Author Name
Affiliation
City, Country
e-mail address

ABSTRACT

Crowdsourcing offers a powerful new paradigm for online work. However, real world tasks are often interdependent, requiring a big picture view of the difference pieces involved. Existing crowdsourcing approaches that support such tasks — ranging from Wikipedia to flash teams — are bottlenecked by relying on a small number of individuals to maintain the big picture. In this paper, we explore the idea that a computational system can scaffold an emerging interdependent, big picture view entirely through the small contributions of individuals, each of whom sees only a part of the whole. To investigate the viability, strengths, and weaknesses of this approach we instantiate the idea in a prototype system for accomplishing distributed information synthesis and evaluate its output across a variety of topics. We also contribute a set of design patterns that may be informative for other systems aimed at supporting big picture thinking in small pieces.

Author Keywords

information synthesis; crowdsourcing; crowd work; complex workflow; design patterns.

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

However, much work in the real world is not amenable to crowdsourcing because of the difficulty in decomposing tasks into small, independent units. As noted by many researchers [9, 38, 47, 48], decomposing tasks ranging from writing an article to creating an animated film often results in pieces that have complex dependencies on each other. Take for example the goal of writing even a simple article about growing tomatoes. At the lowest level, each sentence must be coherent and align with the other sentences in the paragraph. At a higher level, each paragraph within the article must fit together as well, and sections need to have proper transition and flow. Moving to even higher levels, the article must have an appropriate set of topics (e.g., appropriate soil, sunlight, watering, pruning) that are coherent and comprehensive. Information from different sources should be appropriately synthesized and cited while reducing redundancies and bias. Supporting this type of work requires having a big picture view of different pieces at different scales and ensuring they all fit together.

Accomplishing this big picture thinking through small tasks is challenging because it means that each person can only have a limited view of the bigger picture. As a result, many of the applications of crowdsourcing have been limited to simple tasks such as image labeling where each piece can be decomposed and processed independently. Those approaches that do crowdsource tasks requiring big picture thinking — such

How Do I Get My Tomato Plants To Produce More Tomatoes?

Contents

1. **Tomatos - Feeding**
2. Pruning Is Love
3. Maintenance And Harvesting
4. Tomatos - Proper Potting Procedure
5. Weather And Sunlight Conditions
6. Growing Tomatoes
7. Tomatos - Stakes And Support

Tomatos - Feeding

Producing better tomato plants is as simple as picking the perfect soil. There are many market soils or one can add a few things to their own soil. Extra nutrients go a long way in producing more tomatoes per plant.

Tomatoes are heavy feeders since they are smaller plants that depend on the bushy growth to support fruit production. They can benefit from some added nutrition even if you use the best soil. Cutting back on nitrogen will ensure a big, goregous pile of fruit coming your way in no time!

Tomatoes take up nutrients the best when the soil pH ranges from 6.2 to 6.8. They need a constant supply of major and minor plant nutrients. Following the rates on the fertilizer label, mix a balanced timed-release or organic fertilizer to the soil as you prepare planting holes.

Feeding tomatoes regularly is critical for a good yield. At the very least, you need a good liquid food that is high in potassium.

Any tomato feed from a garden cetner should do the job. If you want take it a step further, check out Sea Nymph's natural seaweed-based feed or BioBizz's BioGrow, which include molassess to feed the microbes in the soil. About half way through the season, I add a 1 inch (2.5 cm) layer of worm compost or local farm manure to the top of my containers. This adds extra nutriends and soil life.

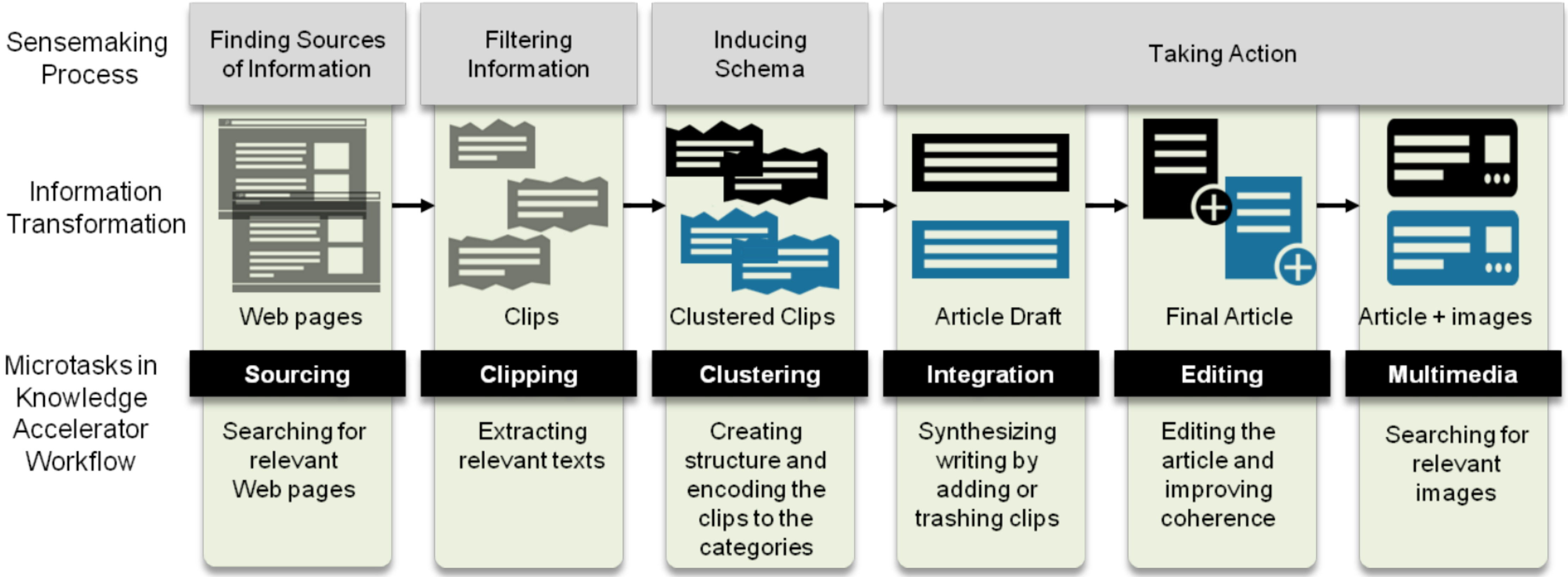
Amend your plant beds with your own or purchased compost; dry, timed-release fertilizer; and most importantly, worm castings. Add 5 cubic feet of Gardner & Bloome compost; 5 quarts of Gardner & Bloome 4-6-3 Tomato, Herb & Vegetable fertilizer; and a quart of 100% pure worm castings for every 50 square feet of garden space.

Tomatos - Feeding

Producing better tomato plants is as simple as picking the perfect soil.

References:

- Vertical veg man: how to grow tomatoes successfully (www.theguardian.com)
- Tomatoes..How To Get The Most From Your Plants In The Garden! (oldworldgardenfarms.com)
- Love Apple Farms (www.growbetterveggies.com)
- 10 Tips for Growing Great Tomatoes (gardening.about.com)


Sensemaking Process
Finding Sources of Information
Filtering Information
Inducing Schema
Taking Action
Information Transformation
Web pages
Clips
Clustered Clips
Article Draft
Final Article
Article + images
Microtasks in Knowledge Accelerator Workflow
Sourcing
Clipping
Clustering
Integration
Editing
Multimedia
Searching for relevant Web pages
Extracting relevant texts
Creating structure and encoding the clips to the categories
Synthesizing writing by adding or trashing clips
Editing the article and improving coherence
Searching for relevant images

Proceedings, The Third AAAI Conference on Human Computation and Crowdsourcing (HCOMP-15)

Crowdsourcing in the Field: A Case Study Using Local Crowds for Event Reporting

Elena Agapie[1], Jaime Teevan[2], Andrés Monroy-Hernández[2]
[1]Human Centered Design & Engineering, University of Washington
eagapie@uw.edu
[2]Microsoft Research
{teevan, amh}@microsoft.com

Abstract

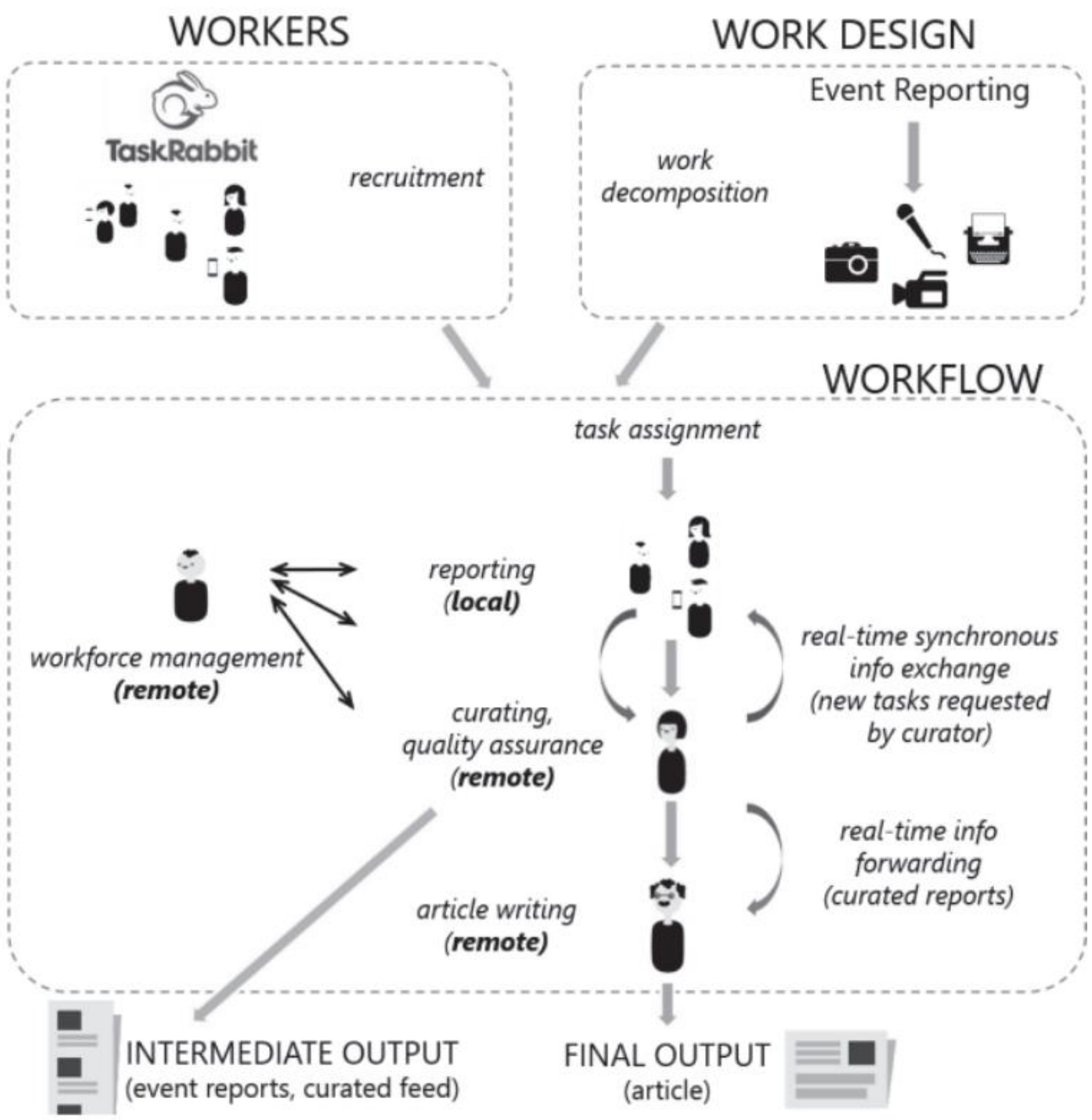
While crowd work typically involves tasks that performed at any time and anywhere, some tasks inherently require the physical presence of workers at a specific time and location. This paper presents a case study of a hybrid crowdsourcing process that involves the collaborative production of event reports using a combination of local and remote workers. The process extends human computation into the physical world by using local workers to collect information in person at events and remote workers to curate the collected information and generate event reports. We deployed the process at 11 events, employing 84 workers, and identified the challenges local workers face as constraints in mobility, time available to perform tasks, unpredictability of events, and interaction with others. We discuss issues related to collaboration with remote workers and bias in field reporting, and conduct a qualitative analysis to make design recommendations for extending human computation into the physical environment.

Introduction

The vast majority of crowd work involves tasks that

decline of news organizations (Snyder & Strömberg 2008, Pew 2014). We do this by producing crowd-based news reports (see example snippet in Figure 1) for 11 small, local events across the United States, including neighborhood festivals, craft fairs, public lectures, and town hall meetings. We employed 84 local and remote on-demand workers from TaskRabbit, oDesk (now UpWork), and event attendees as volunteers. Applying a general-purpose crowdsourcing framework by Kittur et al. (2013), we introduce and examine a crowd work process for local event reporting. We explain how we map each of the framework's four components—workers, work design, workflow, and output—to our local, situated context.

Our findings reveal four primary insights regarding hybrid local-and-remote crowdsourcing environments. (1) Local workers must overcome challenges posed by the physical environment. These include mobility constraints, time constraints, and event unpredictability (e.g., rescheduling of an event). Each constraint affects the worker's preparation time, the number of completed tasks, and their adaptability to changes in tasks and event


WORKERS
TaskRabbit
recruitment
WORK DESIGN
Event Reporting
work decomposition
WORKFLOW
task assignment
reporting (local)
workforce management (remote)
real-time synchronous info exchange (new tasks requested by curator)
curating, quality assurance (remote)
real-time info forwarding (curated reports)
article writing (remote)
INTERMEDIATE OUTPUT
(event reports, curated feed)
FINAL OUTPUT
(article)

Proceedings, The Third AAAI Conference on Human Computation and Crowdsourcing (HCOMP-15)

Guardian: A Crowd-Powered Spoken Dialog System for Web APIs

Ting-Hao (Kenneth) Huang
Carnegie Mellon University
Pittsburgh, PA USA
tinghaoh@cs.cmu.edu

Walter S. Lasecki
University of Michigan
Ann Arbor, MI USA
wlasecki@umich.edu

Jeffrey P. Bigham
Carnegie Mellon University
Pittsburgh, PA USA
jbigham@cs.cmu.edu

Abstract

Natural language dialog is an important and intuitive way for people to access information and services. However, current dialog systems are limited in scope, brittle to the richness of natural language, and expensive to produce. This paper introduces *Guardian*, a crowd-powered framework that wraps existing Web APIs into immediately usable spoken dialog systems. Guardian takes as input the Web API and desired task, and the crowd determines the parameters necessary to complete it, how to ask for them, and interprets the responses from the API. The system is structured so that, over time, it can learn to take over for the crowd. This hybrid systems approach will help make dialog systems both more general and more robust going forward.

Introduction

Conversational interaction allows users to access computer systems and satisfy their information needs in an intuitive and fluid manner, especially in mobile environments. Recently, spoken dialog systems (SDSs) have made great strides in achieving that goal. It is now possible to speak to computers on the phone via conversational assistants on mobile devices, *e.g. Siri*, and, increasingly, from wearable devices on which non-speech interaction is limited. However, *sourcing* to efficiently and cost-effectively enlarge the scope of existing spoken dialog systems. Furthermore, Guardian is structured so that, over time, an automated dialog system could be learned from the chat logs collected by our dialog system and gradually take over from the crowd.

Web-accessible APIs can be viewed as a gateway to the rich information stored on the Internet. The Web contains tens of thousands of APIs (many of which are free) that support access to myriad resources and services. As of April 2015, ProgrammableWeb[1] alone contains the description of more than 13,000 APIs in categories including travel (1,073), reference (1,342), news (1,277), weather (368), health (361), food (356), and many more. These Web APIs can encompass the common functions of popular existing SDSs, such as Siri, which is often used to send text messages, access weather reports, get directions, and find nearby restaurants. Therefore, if SDSs are able to exploit the rich information provided by the thousands of available APIs on the web, their scope would be significantly enlarged.

However, automatically incorporating Web APIs into an SDS is a non-trivial task. To be useful in an application like Siri, these APIs need to be manually wrapped into conversational templates. However, these templates are brittle because they only address a small subset of the many ways to ask for a particular piece of information. Even a topic

Augur: Mining Human Behaviors from Fiction to Power Interactive Systems

Ethan Fast, William McGrath, Pranav Rajpurkar, Michael S. Bernstein
Stanford University
{ethan.fast, wmcgrath, pranavsr, msb}@cs.stanford.edu

ABSTRACT

From smart homes that prepare coffee when we wake, to phones that know not to interrupt us during important conversations, our collective visions of HCI imagine a future in which computers understand a broad range of human behaviors. Today our systems fall short of these visions, however, because this range of behaviors is too large for designers or programmers to capture manually. In this paper, we instead demonstrate it is possible to mine a broad knowledge base of human behavior by analyzing more than one billion words of modern fiction. Our resulting knowledge base, Augur, trains vector models that can predict many thousands of user activities from surrounding objects in modern contexts: for example, whether a user may be eating food, meeting with a friend, or taking a selfie. Augur uses these predictions to identify actions that people commonly take on objects in the world and estimate a user's future activities given their current situation. We demonstrate Augur-powered, activity-based systems such as a phone that silences itself when the odds of you answering it are low, and a dynamic music player that adjusts to your present activity. A field deployment of an Augur-powered wearable camera resulted in 96% recall and 71% precision on its unsupervised predictions of common daily activities. A second evaluation where human judges rated the system's predictions over a broad set of input images found that 94% were rated sensible.

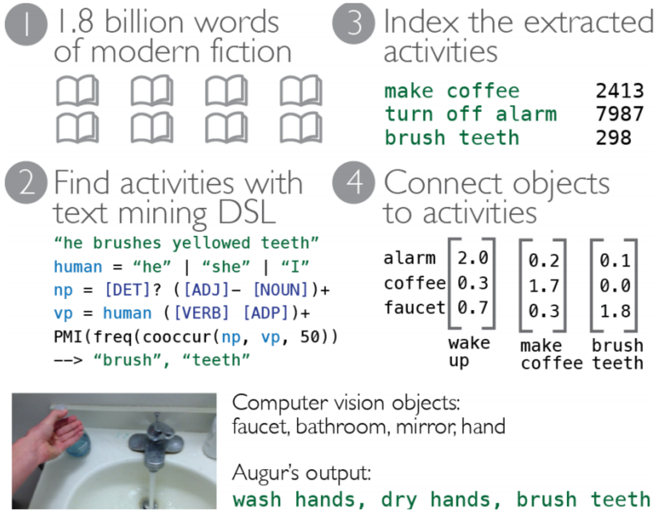


Figure 1. Augur mines human activities from a large dataset of modern fiction. Its statistical associations give applications an understanding of when each activity might be appropriate.

human life. Mark Weiser's first example scenario of ubiquitous computing, for instance, imagines a smart home that predicts its user may want coffee upon waking up [27]. Apple's Knowledge Navigator similarly knows not to let the user's phone ring during a conversation [3]. In science fiction, tech-

Alloy: Clustering with Crowds and Computation

Joseph Chee Chang, Aniket Kittur, Nathan Hahn
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213
{josephcc, nkittur, nhahn}@cs.cmu.edu

ABSTRACT

Crowdsourced clustering approaches present a promising way to harness deep semantic knowledge for clustering complex information. However, existing approaches have difficulties supporting the global context needed for workers to generate meaningful categories, and are costly because all items require human judgments. We introduce Alloy, a hybrid approach that combines the richness of human judgments with the power of machine algorithms. Alloy supports greater global context through a new "*sample and search*" crowd pattern which changes the crowd's task from classifying a fixed subset of items to actively sampling and querying the entire dataset. It also improves efficiency through a two phase process in which crowds provide examples to help a machine cluster the head of the distribution, then classify low-confidence examples in the tail. To accomplish this, Alloy introduces a modular "*cast and gather*" approach which leverages a machine learning backbone to stitch together different types of judgment tasks.

Author Keywords

Computer Supported Cooperative Work (CSCW); World Wide Web and Hypermedia; Database access / Information Retrieval; Empirical Methods, Quantitative

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI):

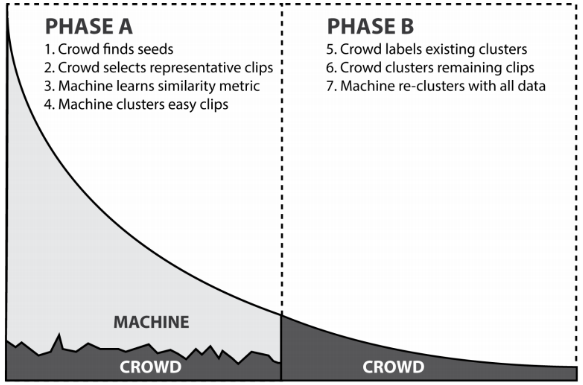


Figure 1. A conceptual overview of the system. In the first phase, crowd workers identify seed clips to train a machine learning model, which is used to classify the "head" of the distribution. In the second phase, crowd workers classify the more difficult items in the "tail". A machine learning backbone provides a consistent way to connect worker judgments in different phases.

Doing so involves complex cognitive processing requiring an understanding of how concepts are related to each other and learning the meaningful differences among them [2, 24, 29].

Computational tools such as machine learning have made great strides in automating the clustering process [4, 10, 6].

WearWrite: Crowd-Assisted Writing from Smartwatches

Michael Nebeling[1], **Alexandra To**[1], **Anhong Guo**[1], **Adrian A. de Freitas**[1],
Jaime Teevan[2], **Steven P. Dow**[1], **Jeffrey P. Bigham**[1]
[1] Human-Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA, USA
[2] Microsoft Research, Redmond, WA, USA
{ mnebelin, aato, anhongg, adefreit, spdow, jbigham }@cs.cmu.edu, teevan@microsoft.com

ABSTRACT
The physical constraints of smartwatches limit the range and complexity of tasks that can be completed. Despite interface improvements on smartwatches, the promise of enabling productive work remains largely unrealized. This paper presents *WearWrite*, a system that enables users to write documents from their smartwatches by leveraging a crowd to help translate their ideas into text. WearWrite users dictate tasks, respond to questions, and receive notifications of major edits on their watch. Using a dynamic task queue, the crowd receives tasks issued by the watch user and generic tasks from the system. In a week-long study with seven smartwatch users supported by approximately 29 crowd workers each, we validate that it is possible to manage the crowd writing process from a watch. Watch users captured new ideas as they came to mind and managed a crowd during spare moments while going about their daily routine. WearWrite represents a new approach to getting work done from wearables using the crowd.

Author Keywords
Smartwatches; Wearables; Crowdsourcing; Writing.

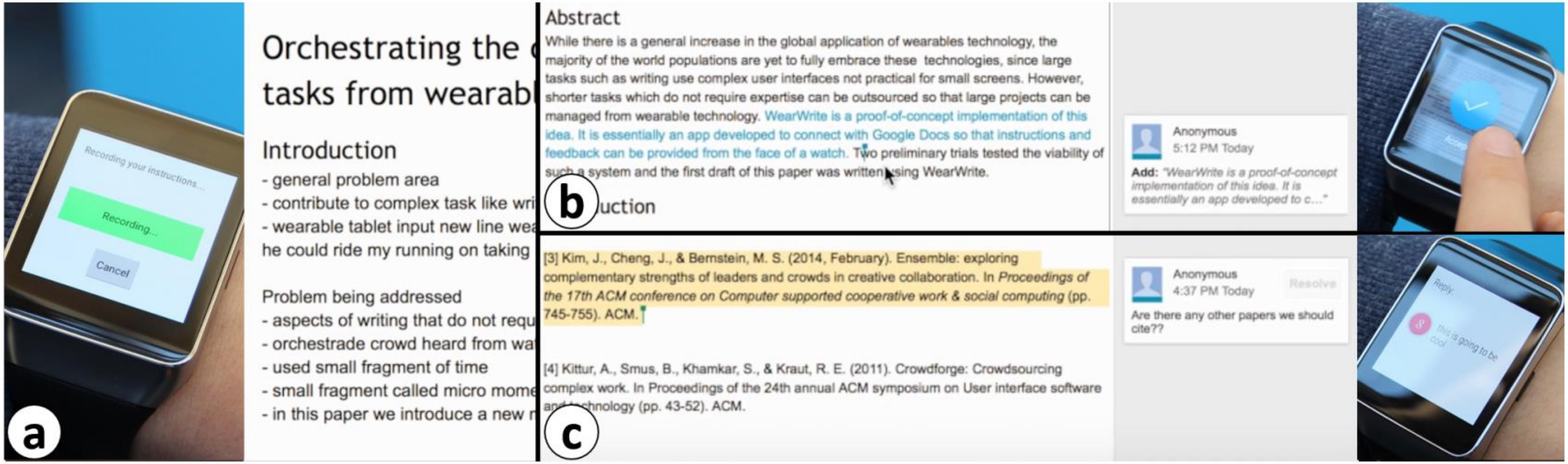
ACM Classification Keywords
H.5.m. Info. Interfaces and Presentation (e.g., HCI): Misc

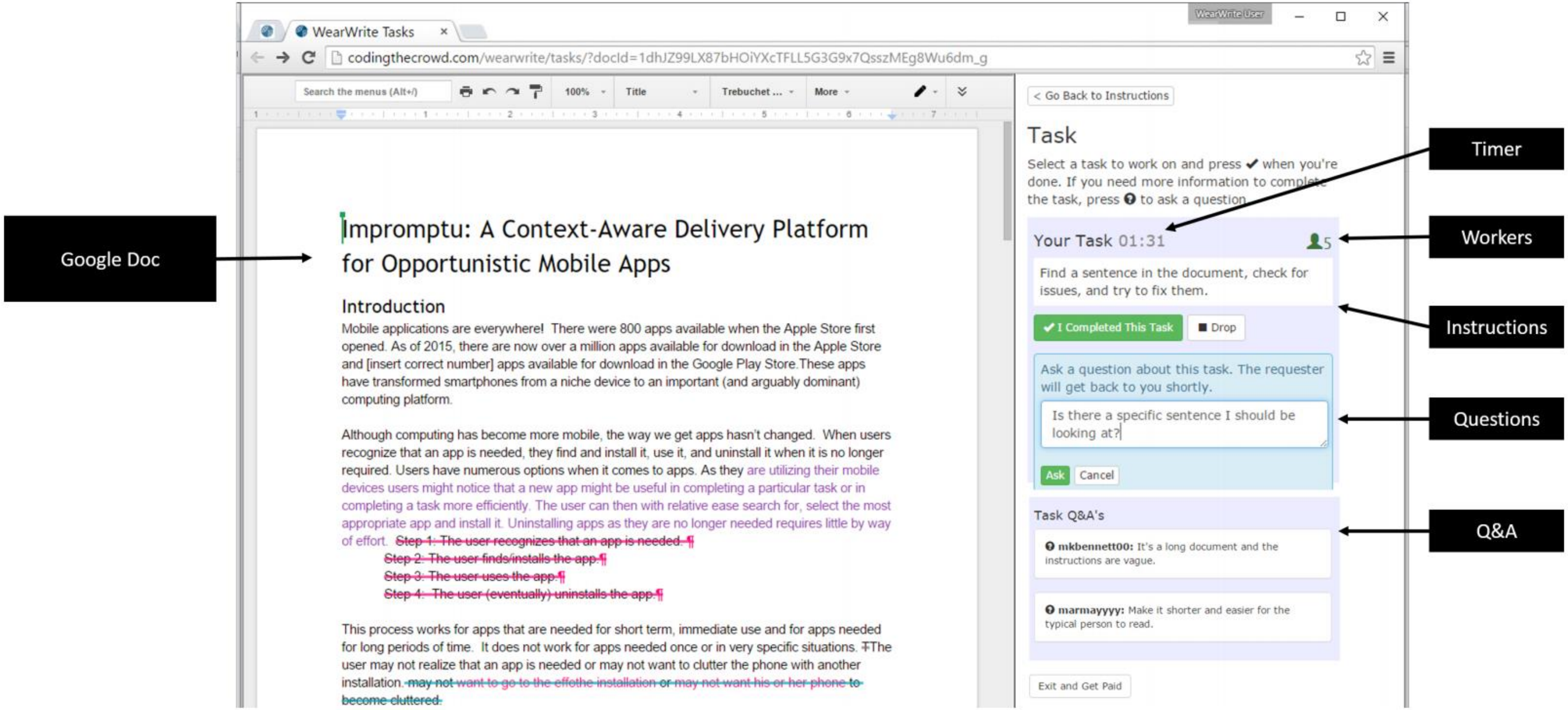
INTRODUCTION
Smartwatches provide immediate access to information from increases the range of possible interactions, limitations with input and output continue to inhibit people's ability to create new content. Touch-based text input from a watch remains much slower than it is from other types of devices, and text-entry alternatives like speech-to-text are error prone. Additionally, limited output on a watch makes it difficult for users to understand complex information and presents a challenge for interface designers who want to provide rich context.

We propose overcoming these limitations by using crowdsourcing. While using the crowd to complete complex tasks like writing is difficult, shepherding the crowd through the process by providing feedback along the way has been shown to result in higher-quality outcomes [10]. We hypothesized that a smartwatch could provide a sufficient and effective interface to orchestrate crowds to create new content, while crowdsourcing in turn could provide a mechanism to overcome limitations of the watch and enable a much wider range of smartwatch interactions than currently possible.

To study this, this paper presents *WearWrite*, a system that connects a smartwatch user as the domain expert of a particular piece of writing with a novice crowd of writers recruited on demand from Amazon Mechanical Turk. As shown in Figure 1, WearWrite consists of two key components:

Watch User Interface WearWrite provides the watch user with a lightweight notification-driven watch interface that


Recording your instructions...
Recording...
Cancel
Orchestrating the
tasks from wearabl
Introduction
- general problem area
- contribute to complex task like wri
- wearable tablet input new line wea
he could ride my running on taking
Problem being addressed
- aspects of writing that do not requ
- orchestrade crowd heard from wa
- used small fragment of time
- small fragment called micro mome
- in this paper we introduce a new
a
Abstract
While there is a general increase in the global application of wearables technology, the majority of the world populations are yet to fully embrace these technologies, since large tasks such as writing use complex user interfaces not practical for small screens. However, shorter tasks which do not require expertise can be outsourced so that large projects can be managed from wearable technology. WearWrite is a proof-of-concept implementation of this idea. It is essentially an app developed to connect with Google Docs so that instructions and feedback can be provided from the face of a watch. Two preliminary trials tested the viability of such a system and the first draft of this paper was written using WearWrite.
b
uction
Anonymous
5:12 PM Today
Add: "WearWrite is a proof-of-concept implementation of this idea. It is essentially an app developed to c..."
[3] Kim, J., Cheng, J., & Bernstein, M. S. (2014, February). Ensemble: exploring complementary strengths of leaders and crowds in creative collaboration. In Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing (pp. 745-755). ACM.
[4] Kittur, A., Smus, B., Khamkar, S., & Kraut, R. E. (2011). Crowdforge: Crowdsourcing complex work. In Proceedings of the 24th annual ACM symposium on User interface software and technology (pp. 43-52). ACM.
c
Anonymous
4:37 PM Today
Resolve
Are there any other papers we should cite??


Google Doc
Timer
Workers
Instructions
Questions
Q&A
WearWrite Tasks
codingthecrowd.com/wearwrite/tasks/?docId=1dhJZ99LX87bHOiYXcTFLL5G3G9x7QsszMEg8Wu6dm_g
Impromptu: A Context-Aware Delivery Platform for Opportunistic Mobile Apps
Introduction
Mobile applications are everywhere! There were 800 apps available when the Apple Store first opened. As of 2015, there are now over a million apps available for download in the Apple Store and [insert correct number] apps available for download in the Google Play Store.These apps have transformed smartphones from a niche device to an important (and arguably dominant) computing platform.
Although computing has become more mobile, the way we get apps hasn't changed. When users recognize that an app is needed, they find and install it, use it, and uninstall it when it is no longer required. Users have numerous options when it comes to apps. As they are utilizing their mobile devices users might notice that a new app might be useful in completing a particular task or in completing a task more efficiently. The user can then with relative ease search for, select the most appropriate app and install it. Uninstalling apps as they are no longer needed requires little by way of effort. Step 1: The user recognizes that an app is needed.
Step 2: The user finds/installs the app.
Step 3: The user uses the app.
Step 4: The user (eventually) uninstalls the app.
This process works for apps that are needed for short term, immediate use and for apps needed for long periods of time. It does not work for apps needed once or in very specific situations. TThe user may not realize that an app is needed or may not want to clutter the phone with another installation. may not want to go to the effothe installation or may not want his or her phone to become cluttered.
< Go Back to Instructions
Task
Select a task to work on and press ✔ when you're done. If you need more information to complete the task, press ❓ to ask a question.
Your Task 01:31
5
Find a sentence in the document, check for issues, and try to fix them.
I Completed This Task
Drop
Ask a question about this task. The requester will get back to you shortly.
Is there a specific sentence I should be looking at?
Ask
Cancel
Task Q&A's
mkbennett00: It's a long document and the instructions are vague.
marmayyyy: Make it shorter and easier for the typical person to read.
Exit and Get Paid

HapTurk: Crowdsourcing Affective Ratings for Vibrotactile Icons

Oliver S. Schneider, Hasti Seifi, Salma Kashani, Matthew Chun, and Karon E. MacLean
oschneid@cs.ubc.ca, seifi@cs.ubc.ca, salma@ece.ubc.ca, mchun345@cs.ubc.ca, maclean@cs.ubc.ca
Sensory, Perception, & Interaction (SPIN) Lab, University of British Columbia, Vancouver, Canada

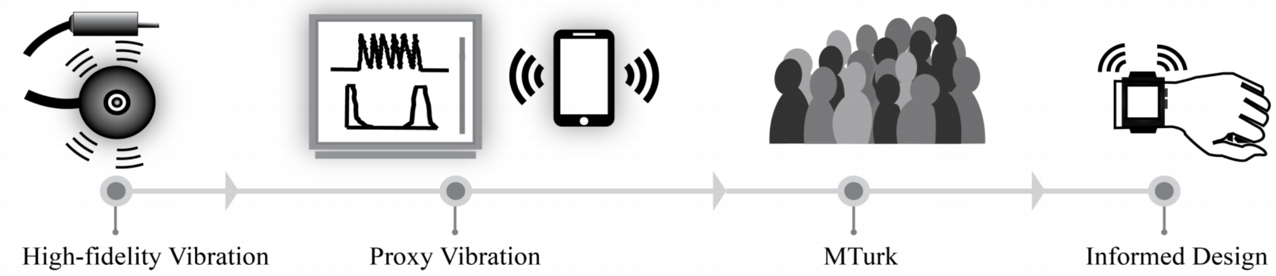


Figure 1: In HapTurk, we access large-scale feedback on informational effectiveness of high-fidelity vibrations after translating them into proxies of various modalities, rendering important characteristics in a crowdsource-friendly way.

ABSTRACT

Vibrotactile (VT) display is becoming a standard component of informative user experience, where notifications and feedback must convey information eyes-free. However, effective design is hindered by incomplete understanding of relevant perceptual qualities. To access evaluation streamlining now common in visual design, we introduce *proxy modalities* as a way to crowdsource VT sensations by reliably communicating high-level features through a crowd-accessible channel. We investigate two proxy modalities to represent a high-fidelity tactor: a new VT visualization, and low-fidelity vibratory translations playable on commodity smartphones. We

INTRODUCTION

In modern handheld and wearable devices, vibrotactile (VT) feedback can provide unintrusive, potentially meaningful cues through wearables in on-the-go contexts [8]. With consumer wearables like Pebble and the Apple Watch featuring high-fidelity actuators, VT feedback is becoming standard in more user tools. Today, VT designers seek to provide sensations with various perceptual and emotional connotations to support the growing use cases for VT feedback (everyday apps, games, etc.). Although low-level design guidelines exist and are helpful for addressing perceptual requirements [5,6,22,32,47], higher-level concerns and design approaches

Crowd-Designed Motivation: Motivational Messages for Exercise Adherence Based on Behavior Change Theory

Roelof A. J. de Vries[1], Khiet P. Truong[1], Sigrid Kwint[2], Constance H.C. Drossaert[2], Vanessa Evers[1]
[1]Human Media Interaction, University of Twente
{r.a.j.devries, k.p.truong, v.evers}@utwente.nl
[2]Psychology, Health and Technology, University of Twente
s.j.m.kwint@student.utwente.nl, c.h.c.drossaert@utwente.nl

ABSTRACT
Developing motivational technology to support long-term behavior change is a challenge. A solution is to incorporate insights from behavior change theory and design technology to tailor to individual users. We carried out two studies to investigate whether the processes of change, from the Transtheoretical Model, can be effectively represented by motivational text messages. We crowdsourced peer-designed text messages and coded them into categories based on the processes of change. We evaluated whether people perceived messages tailored to their stage of change as motivating. We found that crowdsourcing is an effective method to design motivational messages. Our results indicate that different messages are perceived as motivating depending on the stage of behavior change a person is in. However, while motivational messages related to later stages of change were perceived as motivational for those stages, the motivational messages related to earlier stages of change were not. This indicates that a person's stage of change may not be the (only) key factor that determines behavior change. More individual factors need to be considered to design effective motivational technology.

Author Keywords
Crowdsourcing; motivational messages; exercise adherence;

their exercising behavior [10]. The potential benefits of motivational applications in healthcare and well-being are tremendous because of the large number of people who can be reached through mobile technology. This has motivated many researchers to design and develop motivational technology for health.

Design challenges
Challenges arise for HCI researchers aiming to design technology that motivates people to adopt more healthy behaviors long-term. One of these challenges is the evaluation of motivational technology. Although many HCI researchers aim to promote long-term behavior change through their technology, evaluations confirming long-term behavior change are rarely carried out [17]. Practical limitations (e.g., how to track a large number of subjects for more than a few months) and conceptual definitions (e.g., how to measure behavior change, how many months can be considered 'long-term') are some of the aspects researchers struggle with. Another challenge is that there is no method available to translate behavior change theories and models into concrete interaction designs to be used in practice. Also a challenge is to increase effectiveness of the motivational technology [33]. One way to increase effectiveness is to go beyond a one-size-fits-all motivational

Stage of change scenario	**User designed text message**
Precontemplation: “Consider a middle-aged person, with a steady personal life and solid friend foundation. This person lacks regular exercise in his/her daily life and is unwilling to consider starting with this, at least not within the next 6 months.”	“You would really feel better if you started exercising regularly” **Self-reevaluation** “Do you really like being a muffin top?” **Dramatic relief** “Maybe you can get one of your friends to work out with you” **Helping relationships**
Maintenance: “Consider a middle-aged person, with a steady personal life and solid friend foundation. This person participates in regular exercise in his/her daily life, and has been doing so for an extended period of time. This person has been active for more than 6 months.”	“Keep it up!” **Self-liberation** “You have really lost a lot of weight! Keep up the good work on your exercise!” **Reinforcement management** “You can spend your time on more exercises that will remove all your stress” **Counter-conditioning**

A Crowd-Powered System for Fashion Similarity Search

THEODOROS SEMERTZIDIS, Aristotle University of Thessaloniki
JASMINKO NOVAK, European Institute for Participatory Media, University of Applied Sciences Stralsund
MICHALIS LAZARIDIS, Centre for Research and Technology Hellas
MARK MELENHORST, Technical University of Delft
ISABEL MICHEEL, European Institute for Participatory Media
DIMITRIOS MICHALOPOULOS, Centre for Research and Technology Hellas
MARTIN BÖCKLE, European Institute for Participatory Media
MICHAEL G. STRINTZIS, Aristotle University of Thessaloniki
PETROS DARAS, Centre for Research and Technology Hellas

Driven by the needs of customers and industry, online fashion search and analytics are recently gaining much attention. As fashion is mostly expressed by visual content, the analysis of fashion images in online social networks is a rich source of possible insights on evolving trends and customer preferences. Although a plethora of visual content is available, the modeling of clothes' physics and movement, the implicit semantics in fashion designs, and the subjectivity of their interpretation pose difficulties to fully automated solutions for fashion search and analysis. In this article, we present the design and evaluation of a crowd-powered system for fashion similarity search from Twitter, supporting trend analysis for fashion professionals. The system enables fashion similarity search based on specific human-based similarity criteria. This is achieved by implementing a novel machine–crowd workflow that supports complex tasks requiring highly subjective judgments where multiple true solutions may coexist. We discuss how this leads to a novel class of crowd-powered systems for which the output of the crowd is not used to verify the automatic analysis but is the desired outcome. Finally, we show how this kind of crowd involvement enables a novel kind of similarity search and represents a crucial factor for the acceptance of system results by the end user.

Categories and Subject Descriptors: H.3.3 [**Information Search and Retrieval**]: Clustering

General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Real-time crowd, social multimedia retrieval, similarity dimensions, fashion search, crowdsourcing, interactive systems, human–machine cooperation

This work was partially supported by the EC FP7 funded project CUBRIK, ICT- 287704 (www.cubrikproject.eu)

Looking Ahead

Understanding people and crowd work

Pay It Backward: Per-Task Payments on Crowdsourcing Platforms Reduce Productivity

Kazushi Ikeda
KDDI R&D Laboratories, Inc.
Fujimino, Japan
kz-ikeda@kddilabs.jp

Michael S. Bernstein
Stanford University
Stanford, USA
msb@cs.stanford.edu

ABSTRACT
Paid crowdsourcing marketplaces have gained popularity by using piecework, or payment for each microtask, to incentivize workers. This norm has remained relatively unchallenged. In this paper, we ask: is the pay-per-task method the right one? We draw on behavioral economic research to examine whether payment in bulk after every ten tasks, saving money via coupons instead of earning money, or material goods rather than money will increase the number of completed tasks. We perform a twenty-day, between-subjects field experiment (N=300) on a mobile crowdsourcing application and measure how often workers responded to a task notification to fill out a short survey under each incentive condition. Task completion rates increased when paying in bulk after ten tasks: doing so increased the odds of a response by 1.4x, translating into 8% more tasks through that single intervention. Payment with coupons instead of money produced a small negative effect on task completion rates. Material goods were the most robust to decreasing participation over time.

Author Keywords
Crowdsourcing; incentives; motivation; crowd work.

ACM Classification Keywords
H.5.3 Information interfaces and presentation (e.g., HCI): Group and Organization Interfaces.

INTRODUCTION

[3] and Clickworker adopting the same model. Researchers have largely focused their investigations within this model, for example finding that higher payment leads workers to complete more tasks [26] and that incentivizing agreement with other workers has no impact on accuracy [34].

In this paper, we ask: is piecework payment a well-grounded approach? We draw on the behavioral economics literature to suggest several alternatives. First, *goal-setting* experiments have established that specific, ambitious goals lead to higher task performance than general, easy goals [24], in part because people wish to avoid losing funds or effort that they have already invested [6]. Second, not all payments must come as earnings: coupons can also attract participation [18]. So, discounting the cost of a necessity such as a monthly phone bill may be a viable alternative to traditional payment schemes. Third, material goods (e.g., catalog gifts) can also function as incentives [31]. Might crowdsourcing markets offer material goods as an alternative to cash? These investigations can challenge the default design of crowdsourcing marketplaces.

The present study aims to examine whether these alternate incentive approaches implied by economic and psychological theory result in higher performance. It measures the likelihood of responding to time-limited mobile crowdsourcing tasks under incentive conditions that (1) reward per task vs. reward in bulk only after ten tasks

Using Crowd Sourcing to Measure the Effects of System Response Delays on User Engagement

Brandon Taylor
Carnegie Mellon University
bttaylor@cs.cmu.edu

Anind K. Dey
Carnegie Mellon University
anind@cs.cmu.edu

Daniel Siewiorek
Carnegie Mellon University
dps@cs.cmu.edu

Asim Smailagic
Carnegie Mellon University
asim@cs.cmu.edu

ABSTRACT

It is well established that delays in system response time negatively impact productivity, error rates and user satisfaction. What is less clear is the degree to which these effects deter users from engaging with a system. Usability guidelines provide rough response time targets for minimizing these effects across various types of interactions. However, developers faced with technical limitations or cost constraints that prevent them from meeting such targets are given no data with which to estimate the impact that system response delays will have on user engagement. In this work, we demonstrate a methodology for using crowd sourcing platforms to examine (1) the relative impacts of different delay types and (2) the effects of marginal changes in system response times. We compare two common network delay types, those caused by limited bandwidth (increased download times) and those caused by network latency (lag in responsiveness), and present how these delays reduce engagement in the context of a crowd sourced image classification task. Furthermore, we model how financial incentives interact with system response delays to impact user engagement. Finally, we show how such models can be used to optimize the cost of system design choices.

Author Keywords

System Response Delays; Usability; Crowd Sourcing

delays are often an unpredictable side effect produced by a combination of limited computational resources [4], greedy application processes [30], and the physical limits of information transfer.

Understanding how users are affected by system response delays has long been a topic of interest in HCI. Most work related to system response delays can be traced back to Robert Miller's seminal work on the matter in 1968 [21]. Following Miller's suggestions, human factors research put many of his assertions to the test and gained empirical results correlating system response delays with increases in error rates and decreases in productivity and user satisfaction [6,8,9,10,16,32,33].

Over the years, the findings from these studies have been codified into rules of thumb for interaction designers. Usability textbooks vary, but response time guidelines generally offer the advice that one or two seconds "seems appropriate for many tasks" [31], whereas response times under 0.1 second will seem as though "the system is reacting instantaneously" [24]. While these texts do make it clear that these heuristics do not apply in all cases, they do not offer much advice for determining specific system response time requirements for an application beyond "measuring productivity, errors, and the cost of providing short response times" [31].

Is there a Doctor in the Crowd? Diagnosis Needed! (for less than $5)

James Cheng[1], Monisha Manoharan[2], Matthew Lease[1], and Yan Zhang[1]
[1]School of Information and [2]Department of Computer Science
University of Texas at Austin, USA

Abstract
We investigate the feasibility of crowd-based medical diagnosis by posting medical cases on a variety of crowdsourcing platforms: general and specialized volunteer question answering sites, and pay-based Mechanical Turk (MTurk) and oDesk. To assess the crowd's ability to diagnose cases of varying difficulty, three sets of medical cases are considered. While volunteer channels proved ineffective for us, we discuss design limitations and opportunities for improvement. In contrast, Mechanical Turk workers without medical training not only correctly diagnosed easy cases, but also a previously unsolved case from *CrowdMed* involving extensive patient details. Likely due to varying expertise, MTurk workers and oDesk health professionals also differed in their willingness to provide uncertain diagnoses, diagnosis rationales, and reliance on personal experience with a disease to diagnose it.

Keywords: crowdsourcing; human computation; Mechanical Turk; medical and health support

Research Data: Available online at: `https://www.ischool.utexas.edu/~ml/data`

Contact: Matthew Lease <ml@utexas.edu>

1 Introduction

Medical diagnosis is a central component of physician competence, with mastery as a major objective of medical education (Elstein, 1978; Norman, 2005). It is widely recognized that diagnosing is highly individual, dependent on physicians' perceived difficulty of a particular case and their relevant domain knowledge (Elstein, 2002). As a result, health consumers in practice often choose to seek additional opinions from other providers.

Health 2.0 (or *participatory health*) is a growing movement toward use of Internet technologies for personalized health-care (Swan, 2012). Specialized social networks for consumer health have become particularly popular, allowing people to ask questions about their health problems and receive feedback from others with similar symptoms or medical specialists. As a result, the pool from which consumers can seek "second opinions" has been drastically expanded. At present, 35% of adults in the U.S.A. have gone online to

PLOS | ONE

RESEARCH ARTICLE

An Experimental Study of Team Size and Performance on a Complex Task

Andrew Mao[1] *, Winter Mason[2], Siddharth Suri[1], Duncan J. Watts[1]

1 Microsoft Research, 641 Avenue of the Americas, New York, NY 10011, United States of America,
2 Facebook Inc., 1299 Pennsylvania Avenue, NW, Washington, DC 20004, United States of America

* mao@microsoft.com

OPEN ACCESS

Citation: Mao A, Mason W, Suri S, Watts DJ (2016) An Experimental Study of Team Size and Performance on a Complex Task. PLoS ONE 11(4): e0153048. doi:10.1371/journal.pone.0153048

Editor: Luís A. Nunes Amaral, Northwestern University, UNITED STATES

Received: December 3, 2015

Accepted: March 21, 2016

Abstract

The relationship between team size and productivity is a question of broad relevance across economics, psychology, and management science. For complex tasks, however, where both the potential benefits and costs of coordinated work increase with the number of workers, neither theoretical arguments nor empirical evidence consistently favor larger vs. smaller teams. Experimental findings, meanwhile, have relied on small groups and highly stylized tasks, hence are hard to generalize to realistic settings. Here we narrow the gap between real-world task complexity and experimental control, reporting results from an online experiment in which 47 teams of size ranging from $n = 1$ to 32 collaborated on a realistic *crisis mapping* task. We find that individuals in teams exerted lower overall effort than independent workers, in part by allocating their effort to less demanding (and less productive) sub-tasks; however, we also find that individuals in teams collaborated more with increasing team size. Directly comparing these competing effects, we find that the largest teams outperformed an equivalent number of independent workers, suggesting that gains to collaboration dominated losses to effort. Importantly, these teams also performed comparably to a field deployment of crisis mappers, suggesting that experiments of the type

"Why would anybody do this?": Older Adults' Understanding of and Experiences with Crowd Work

Robin Brewer
Northwestern University
Evanston, IL, USA
rnbrewer@u.northwestern.edu

Meredith Ringel Morris
Microsoft Research
Redmond, WA, USA
merrie@microsoft.com

Anne Marie Piper
Northwestern University
Evanston, IL, USA
ampiper@northwestern.edu

ABSTRACT
Diversifying participation in crowd work can benefit the worker and requester. Increasing numbers of older adults are online, but little is known about their awareness of or how they engage in mainstream crowd work. Through an online survey with 505 seniors, we found that most have never heard of crowd work but would be motivated to complete tasks by earning money or working on interesting or stimulating tasks. We follow up results from the survey with interviews and observations of 14 older adults completing crowd work tasks. While our survey data suggests that financial incentives are encouraging, in-depth interviews reveal that a combination of personal and social incentives may be stronger drivers of participation, but only if older adults can overcome accessibility issues and understand the purpose of crowd work. This paper contributes insights into how crowdsourcing sites could better engage seniors and other users.

Author Keywords
Crowdsourcing; motivation; older adults; online work.

ACM Classification Keywords
H.5.m. Information interfaces and presentation (HCI): Misc.

INTRODUCTION
Crowd work is an emerging form of digital labor in which many workers complete small *microtasks* online for various requesters. Crowd work arrangements are established work tend to be in young or middle adulthood [13,20,35]. Indeed, most research on crowd work focuses on this demographic, resulting in a gap in our understanding of why older people may or may not engage in crowd work. On Amazon's Mechanical Turk, a popular crowdsourcing platform for non-experts, the average worker age is 30 years old [35]. Additionally, of a 1000 participant sample in a study which sought to understand the demographics on AMT, only 2.8% were over the age of 60 [20]. While nearly 60% of older adults in the U.S. are now online [39], only 5% of workers are over 60 years old on CrowdFlower, whereas more than 50% of its contributors are under 30 years old [13]. Although prior work examined specialized crowdsourcing platforms for older adults (e.g., [25]), it is important to understand why older adults are participating so minimally on non-specialized (or general-purpose) crowd work platforms that dominate the marketplace (e.g., AMT). Towards this end, the present paper contributes new insights on issues of inclusion, accessibility, and older adults' values around participating in crowd work.

Engaging older adults in crowd work may hold value both from a worker perspective and from a platform/requestor perspective. Recent work by Zyskowski et al. [45] suggests that people with disabilities (a group that tends to include many older adults) find crowd work an appealing form of employment due the scheduling and location flexibility it provides. Understanding how to help under-represented

The Communication Network Within the Crowd

Ming Yin[*]
Harvard University
mingyin@fas.harvard.edu

Mary L. Gray
Microsoft Research
mlg@microsoft.com

Siddharth Suri
Microsoft Research
suri@microsoft.com

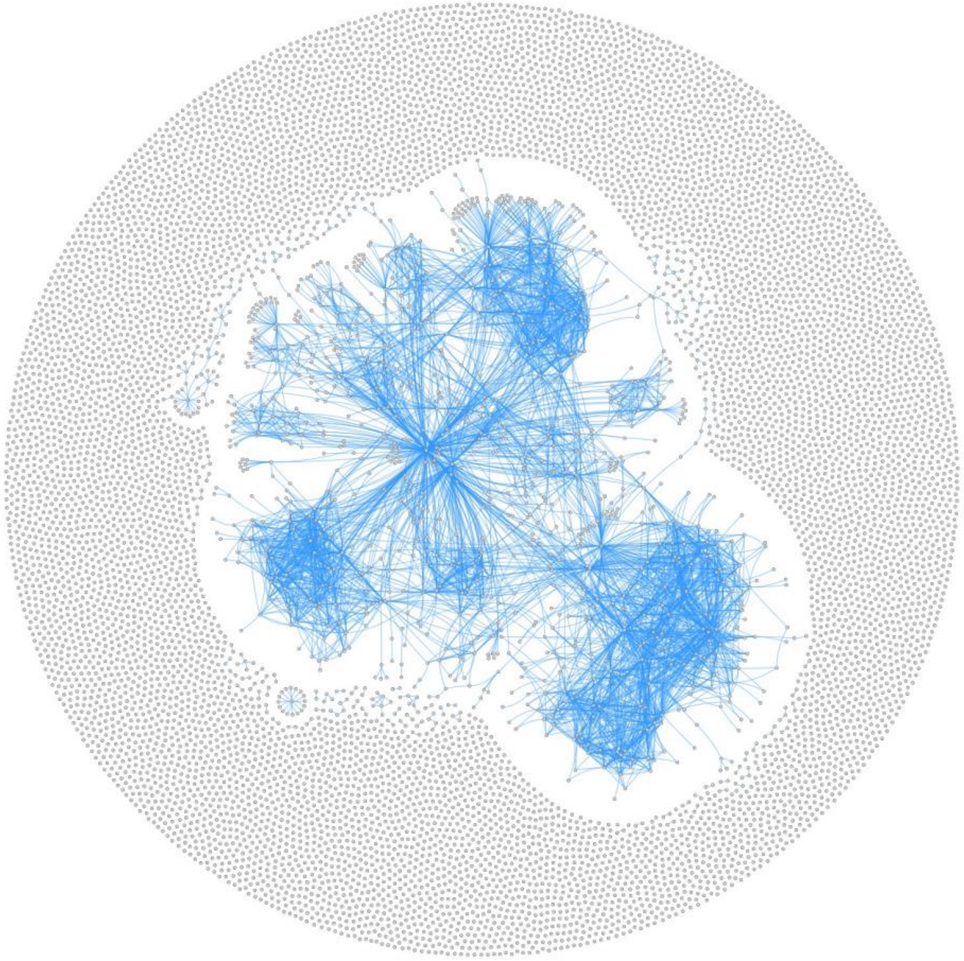
Jennifer Wortman Vaughan
Microsoft Research
jenn@microsoft.com

ABSTRACT

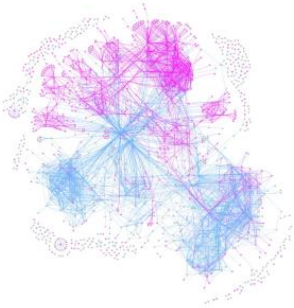
Since its inception, crowdsourcing has been considered a black-box approach to solicit labor from a crowd of workers. Furthermore, the "crowd" has been viewed as a group of independent workers dispersed all over the world. Recent studies based on in-person interviews have opened up the black box and shown that the crowd is not a collection of independent workers, but instead that workers communicate and collaborate with each other. Put another way, prior work has shown the existence of edges between workers. We build on and extend this discovery by mapping the entire communication network of workers on Amazon Mechanical Turk, a leading crowdsourcing platform. We execute a task in which over 10,000 workers from across the globe self-report their communication links to other workers, thereby mapping the communication network among workers. Our results suggest that while a large percentage of workers indeed appear to be independent, there is a rich network topology over the rest of the population. That is, there is a substantial communication network *within* the crowd. We further examine how online forum usage relates to network topology, how workers communicate with each other via this network, how workers' experience levels relate to their network positions, and how U.S. workers differ from international workers in their network characteristics. We conclude by discussing the implications of our findings for requesters, workers, and platform providers like Amazon

questers use an API provided by the platform to post tasks for workers to complete. Workers can then browse available tasks and choose which tasks to perform in exchange for prespecified payments. The platform's API defines the communication protocol between a requester and the workers. By design, it hides personal attributes of workers, such as age, gender, and ethnicity, from the requester. Similarly, the API hides *social* characteristics of workers such as how many friends they have who also do crowdwork or if they are currently working on a task with other workers. In fact, Mechanical Turk does not ask workers about their personal characteristics and does not have access to social characteristics or any details of worker communication that takes place off the platform. Without this information, requesters may come to view workers as simply a black box method to accomplish tasks. In light of this, it is not surprising that crowds are often seen as groups of independent workers, with little attention paid to the connections between them.

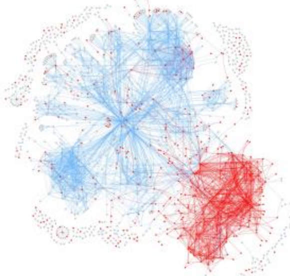
This notion of crowds as independent workers was recently dispelled by Gray et al. [6], who opened up the black box and showed that workers are *not* independent but rather connected through social ties. Through a mix of ethnographic fieldwork, in-person interviews, surveys, and large scale data analyses of four different crowdsourcing platforms, they showed that workers collaborate with one another to meet social and technical needs left wanting by the crowdsourcing platforms studied. More specifically, they showed



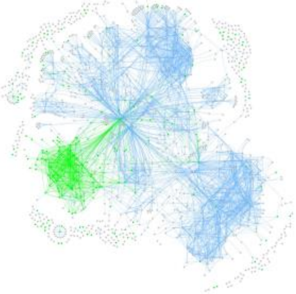
(a) The communication network



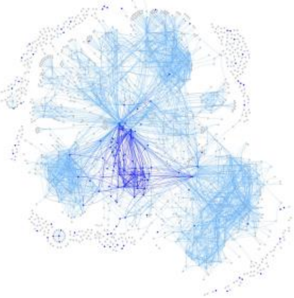
(b) Reddit HWTF



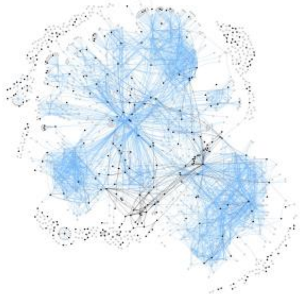
(c) MTurkGrind



(d) TurkerNation



(e) Facebook



(f) MTurkForum

Taking a HIT: Designing around Rejection, Mistrust, Risk, and Workers' Experiences in Amazon Mechanical Turk

Brian McInnis[1], Dan Cosley[1], Chaebong Nam[2], Gilly Leshed[1]
Information Science[1], Law School[2], Cornell University
{bjm277, drc44, cn277, gl87}@cornell.edu

ABSTRACT
Online crowd labor markets often address issues of risk and mistrust between employers and employees from the employers' perspective, but less often from that of employees. Based on 437 comments posted by crowd workers (Turkers) on the Amazon Mechanical Turk (AMT) participation agreement, we identified *work rejection* as a major risk that Turkers experience. Unfair rejections can result from poorly-designed tasks, unclear instructions, technical errors, and malicious Requesters. Because the AMT policy and platform provide little recourse to Turkers, they adopt strategies to minimize risk: avoiding new and known bad Requesters, sharing information with other Turkers, and choosing low-risk tasks. Through a series of ideas inspired by these findings—including notifying Turkers and Requesters of a broken task, returning rejected work to Turkers for repair, and providing collective dispute resolution mechanisms—we argue that making reducing risk and building trust a first-class design goal can lead to solutions that improve outcomes around rejected work for all parties in online labor markets.

Author Keywords
Crowdsourcing; trust; risk management; design; rejection

ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION

ambiguous [4, 16, 41]. These conditions raise concerns about fairness [27] and abuse [44].

These concerns are exacerbated by AMT's *hands-off* approach to the labor market. AMT's participation agreement[1] classifies Turkers as independent contractors, free to accept any task they qualify for (§3b). At the same time, Requesters have the right to reject a Turker's completed work without payment (§3a) while AMT, providing only the venue for an exchange (§2), is not involved in resolving any labor disputes (§3f). When a Turker's work is rejected, the result is lost pay, time, and reputation, and AMT's stance gives workers little recourse. These policies, and other aspects of the AMT platform we detail below, make the practice of crowd working risky.

In this paper, we focus on how Turkers manage the risks of rejected work. Based on 1,092 comments collected during an experiment asking Turkers to comment on Turker-relevant aspects of the AMT participation agreement, we identified 437 that dealt with challenges, experiences, and practices around the risk of work rejection. Although respondents realize that some work is legitimately rejected, many rejections are seen as unfair. Problems with task clarity, design, and implementation can lead to rejections; many rejections include little rationale; some rejections seem arbitrary or malicious. No matter what the reason, Requesters are often non-responsive to Turkers who question the rejections—a position they can adopt because

Policy and Internet

Accounting for Market Frictions and Power Asymmetries in Online Labor Markets

Sara Constance Kingsley,* Mary L. Gray and Siddharth Suri

Amazon Mechanical Turk (AMT) is an online labor market that defines itself as "a marketplace for work that requires human intelligence." Early advocates and developers of crowdsourcing platforms argued that crowdsourcing tasks are designed so people of any skill level can do this labor online. However, as the popularity of crowdsourcing work has grown, the crowdsourcing literature has identified a peculiar issue: that work quality of workers is not responsive to changes in price. This means that unlike what economic theory would predict, paying crowdworkers higher wages does not lead to higher quality work. This has led some to believe that platforms, like AMT, attract poor quality workers. This article examines different market dynamics that might, unwittingly, contribute to the inefficiencies in the market that generate poor work quality. We argue that the cultural logics and socioeconomic values embedded in AMT's platform design generate a greater amount of market power for requesters (those posting tasks) than for individuals doing tasks for pay (crowdworkers). We attribute the uneven distribution of market power among participants to labor market frictions, primarily characterized by uncompetitive wage posting and incomplete information. Finally, recommendations are made for how to tackle these frictions when contemplating the design of an online labor market.

KEY WORDS: market friction, crowdsourcing, quality, Amazon Mechanical Turk, imperfect competition, online labor markets

Introduction

Crowdsourcing is task-orientated labor distributed online through an open

Intervention Strategies for Increasing Engagement in Crowdsourcing: Platform, Predictions, and Experiments

Avi Segal and **Ya'akov (Kobi) Gal**
Ben-Gurion University of the Negev, Israel

Ece Kamar and **Eric Horvitz**
Microsoft Research, Redmond WA

Alex Bowyer and **Grant Miller**
University of Oxford, U.K.

Abstract

Volunteer-based crowdsourcing depend critically on maintaining the engagement of participants. We explore a methodology for extending engagement in citizen science by combining machine learning with intervention design. We first present a platform for using real-time predictions about forthcoming disengagement to guide interventions. Then we discuss a set of experiments with delivering different messages to users based on the proximity to the predicted time of disengagement. The messages address motivational factors that were found in prior studies to influence users' engagements. We evaluate this approach on Galaxy Zoo, one of the largest citizen science application on the web, where we traced the behavior and contributions of thousands of users who received intervention messages over a period of a few months. We found sensitivity of the amount of user contributions to both the timing and nature of the message. Specifically, we found that a message emphasizing the helpfulness of individual users significantly increased users' contributions when delivered according to predicted times of disengagement, but

Paid crowdsourcing platforms such as Amazon mechanical Turk or CrowdFlower incentivize workers to remain active and to make high quality contributions with monetary incentives [Ipeirotis, 2010; Horton and Chilton, 2010; Tran-Thanh *et al.*, 2015]. In contrast, volunteer-based platforms rely on intrinsically motivated participants who do not get paid for their contributions. While some volunteers may become recurrent contributors, the vast majority of volunteer participants are "dabblers" [Eveleigh *et al.*, 2014], who make a small number of contributions before disengaging and never returning. Despite this casual and non-committed participation pattern, dabblers contribute a substantial amount of the overall effort in these platforms. Even a small increase in the contribution rates of these users can lead to a significant improvement in productivity for the platform.

We report on a study of intervention strategies aimed at increasing engagement in volunteer-based crowdsourcing. We employ machine learning to build predictive models that can be used in real time to identify users who are at risk for disengaging soon from the system. We describe a real-time platform that uses predictions of disengagement to guide the delivery of messages designed to increase users' motivation to contribute. The approach addresses two key challenges of intervention design: First, we evaluate different intervention strategies with respect to balancing the potential disruption

PLOS | ONE

RESEARCH ARTICLE

Quantifying the Search Behaviour of Different Demographics Using Google Correlate

Adrian Letchford*, Tobias Preis, Helen Susannah Moat

Data Science Lab, Behavioural Science, Warwick Business School, University of Warwick, CV4 7AL, Coventry, United Kingdom

* Adrian.Letchford@wbs.ac.uk

CrossMark
click for updates

Citation: Letchford A, Preis T, Moat HS (2016) Quantifying the Search Behaviour of Different Demographics Using Google Correlate. PLoS ONE 11(2): e0149025. doi:10.1371/journal.pone.0149025

Editor: Matjaz Perc, University of Maribor, SLOVENIA

Received: November 4, 2015

Accepted: January 26, 2016

Abstract

Vast records of our everyday interests and concerns are being generated by our frequent interactions with the Internet. Here, we investigate how the searches of *Google* users vary across U.S. states with different birth rates and infant mortality rates. We find that users in states with higher birth rates search for more information about pregnancy, while those in states with lower birth rates search for more information about cats. Similarly, we find that users in states with higher infant mortality rates search for more information about credit, loans and diseases. Our results provide evidence that Internet search data could offer new insight into the concerns of different demographics.

Introduction

Atelier: Repurposing Expert Crowdsourcing Tasks as Micro-internships

Ryo Suzuki[1], **Niloufar Salehi**[2], **Michelle S. Lam**[2], **Juan C. Marroquin**[2], **Michael S. Bernstein**[2]
[1]University of Colorado Boulder, [2]Stanford University
ryo.suzuki@colorado.edu, niloufar@cs.stanford.edu, {mlam4, juanm95} @stanford.edu, msb@cs.stanford.edu

ABSTRACT

Expert crowdsourcing marketplaces have untapped potential to empower workers' career and skill development. Currently, many workers cannot afford to invest the time and sacrifice the earnings required to learn a new skill, and a lack of experience makes it difficult to get job offers even if they do. In this paper, we seek to lower the threshold to skill development by repurposing existing tasks on the marketplace as mentored, paid, real-world work experiences, which we refer to as micro-internships. We instantiate this idea in Atelier, a micro-internship platform that connects crowd interns with crowd mentors. Atelier guides mentor–intern pairs to break down expert crowdsourcing tasks into milestones, review intermediate output, and problem-solve together. We conducted a field experiment comparing Atelier's mentorship model to a non-mentored alternative on a real-world programming crowdsourcing task, finding that Atelier helped interns maintain forward progress and absorb best practices.

Author Keywords

crowdsourcing; crowd work; micro-internships

ACM Classification Keywords

H.5.3. Group and Organization Interfaces

INTRODUCTION

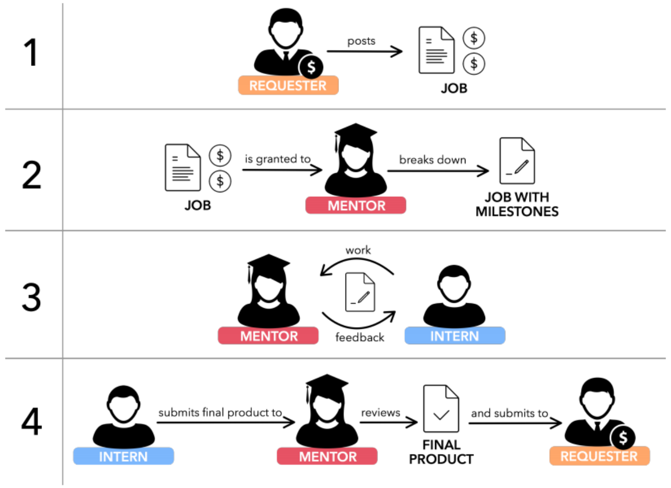


Figure 1. Atelier connects crowd workers (interns) with others on the crowdsourcing marketplace who have experience with a skill (mentors). It then facilitates a short micro-internship as the intern completes a real-world task from the marketplace under the tutelage of the mentor. (1) Requesters post a task, then (2) choose a mentor who breaks down the task into milestones. (3) Intern works on the job while receiving feedback and guidance from the mentor, and finally (4) submits the final product to the mentor, who reviews it and submits it to the requester.

marketplaces as places to seek temporary jobs for their pre-existing skills rather than as venues for long-term career de-

Toward a Learning Science for Complex Crowdsourcing Tasks

Shayan Doroudi[1], **Ece Kamar**[2], **Emma Brunskill**[1], **Eric Horvitz**[2]
[1]Carnegie Mellon University
Pittsburgh, PA
{shayand, ebrun}@cs.cmu.edu
[2]Microsoft Research
Redmond, WA
{eckamar, horvitz}@microsoft.com

ABSTRACT
We explore how crowdworkers can be trained to tackle complex crowdsourcing tasks. We are particularly interested in training novice workers to perform well on solving tasks in situations where the space of strategies is large and workers need to discover and try different strategies to be successful. In a first experiment, we perform a comparison of five different training strategies. For complex web search challenges, we show that providing expert examples is an effective form of training, surpassing other forms of training in nearly all measures of interest. However, such training relies on access to domain expertise, which may be expensive or lacking. Therefore, in a second experiment we study the feasibility of training workers in the absence of domain expertise. We show that having workers validate the work of their peer workers can be even more effective than having them review expert examples if we only present solutions filtered by a threshold length. The results suggest that crowdsourced solutions of peer workers may be harnessed in an automated training pipeline.

Author Keywords
crowdsourcing; worker training; worked examples; peer review; education; web search

ACM Classification Keywords
H.5.m. Information Interfaces and Presentation (e.g. HCI):

solve [13, 5, 3, 14, 30]. However, such task decomposition requires the careful design and engineering of task-specific workflows. We investigate the less-studied case of crowdsourcing tasks that cannot be decomposed in a straightforward manner. Specifically, we consider the class of **complex problem solving tasks** that satisfy the following three properties: (1) there is a large space of potential strategies that workers can use to solve the tasks, (2) workers have the capacity to solve the tasks by discovering and trying different strategies, and yet (3) a significant proportion of unskilled workers are unable to solve these tasks with high accuracy. We address the prospect of extending the reach of crowdsourcing to these complex problem solving tasks by exploring methods for training unskilled workers to solve these tasks with higher accuracy.

Little is known about how to optimally train crowdworkers to perform complex tasks in a cost-effective way. Experts may be unavailable or unwilling to invest time into training crowdworkers and, in many cases, requesters themselves do not understand how to solve their complex tasks let alone how to train others to solve them. Furthermore, there may be a large continuum of possible strategies for solving these problems, with different strategies being optimal in different instances of the task. The strategies used to solve the task may also need to change over time (e.g, to detect web spam, workers need to adapt to adversarial shifts in spammer strategies over time). As such, it can be unwieldy, if not impossible, to write

Learning from the Crowd: Observational Learning in Crowdsourcing Communities

Lena Mamykina
Columbia University
New York, NY, USA
lena.mamykina@dbmi.columbia.edu

Thomas N. Smyth, Jill P. Dimond
Sassafras Tech Collective
Ann Arbor, MI, USA
{jill, tom}@sassafras.coop

Krzysztof Z. Gajos
Harvard SEAS
Cambridge, MA, USA
kgajos@eecs.harvard.edu

ABSTRACT

Crowd work provides solutions to complex problems effectively, efficiently, and at low cost. Previous research showed that feedback, particularly correctness feedback can help crowd workers improve their performance; yet such feedback, particularly when generated by experts, is costly and difficult to scale. In our research we investigate approaches to facilitating continuous observational learning in crowdsourcing communities. In a study conducted with workers on Amazon Mechanical Turk, we asked workers to complete a set of tasks identifying nutritional composition of different meals. We examined workers' accuracy gains after being exposed to expert-generated feedback and to two types of peer-generated feedback: direct accuracy assessment with explanations of errors, and a comparison with solutions generated by other workers. The study further confirmed that expert-generated feedback is a powerful mechanism for facilitating learning and leads to significant gains in accuracy. However, the study also showed that comparing one's own solutions with a variety of solutions suggested by others and their comparative frequencies leads to significant gains in accuracy. This solution is particularly attractive because of its low cost, minimal impact on time and cost of job completion, and high potential for adoption by a variety of crowdsourcing platforms.

Author Keywords

computing are beyond doubt: it provides solutions to complex problems effectively, efficiently, and at low cost. Crowd computing is particularly effective for completing tasks that require human perception, judgment and common sense. Such tasks are frequently beyond the reach of computers, yet they can be solved with little effort by people. Crowd computing is less commonly used for tasks that require special knowledge and skills, such as visual design, coding and programming, and nutritional assessment of meals. Tasks like these typically require both domain and discipline-specific knowledge, as well as awareness of social norms, practices, and conventions related to these disciplines. One approach to enabling crowdsourcing for these tasks is through expert-based communities, such as 99design.com that focuses on graphic design. However, these specialized communities might present high entry barriers for crowd workers. An attractive alternative to searching for existing expertise is to develop mechanisms for training crowd workers on the job and helping them acquire the necessary knowledge and skills. This approach would benefit the requesters, who could receive higher quality solutions. In addition, it would benefit the workers and allow them to acquire and develop new skills, grow expertise and, potentially, advance their careers [14].

For crowdsourcing tasks that rely on general human abilities and common sense (such as writing product

Modeling Temporal Crowd Work Quality with Limited Supervision

Hyun Joon Jung and Matthew Lease
{hyunJoon | ml}@utexas.edu
School of Information
University of Texas at Austin, USA

Abstract

While recent work has shown that a worker's performance can be more accurately modeled by temporal correlation in task performance, a fundamental challenge remains in the need for expert gold labels to evaluate a worker's performance. To solve this problem, we explore two methods of utilizing limited gold labels, initial training and periodic updating. Furthermore, we present a novel way of learning a prediction model in the absence of gold labels with uncertainty-aware learning and soft-label updating. Our experiment with a real crowdsourcing dataset demonstrates that periodic updating tends to show better performance than initial training when the number of gold labels are very limited (< 25).

Keywords: *crowdsourcing, human computation, prediction, uncertainty-aware learning, time-series modeling*

Introduction

While crowdsourcing offers a cost-efficient and scalable method to collect human labels via the Internet, the quality of work performed by the crowd can greatly vary across individuals, which risks compromising overall data quality. As in traditional employment, a common management strategy is to evaluate the performance of each worker on a regular basis, enabling use of various carrots (e.g., performance-all examples have known gold labels readily available to immediately evaluate each worker response as it arrives. Of course, if we already had gold labels in-hand for all examples, there would be no need for collecting additional labels from the crowd.

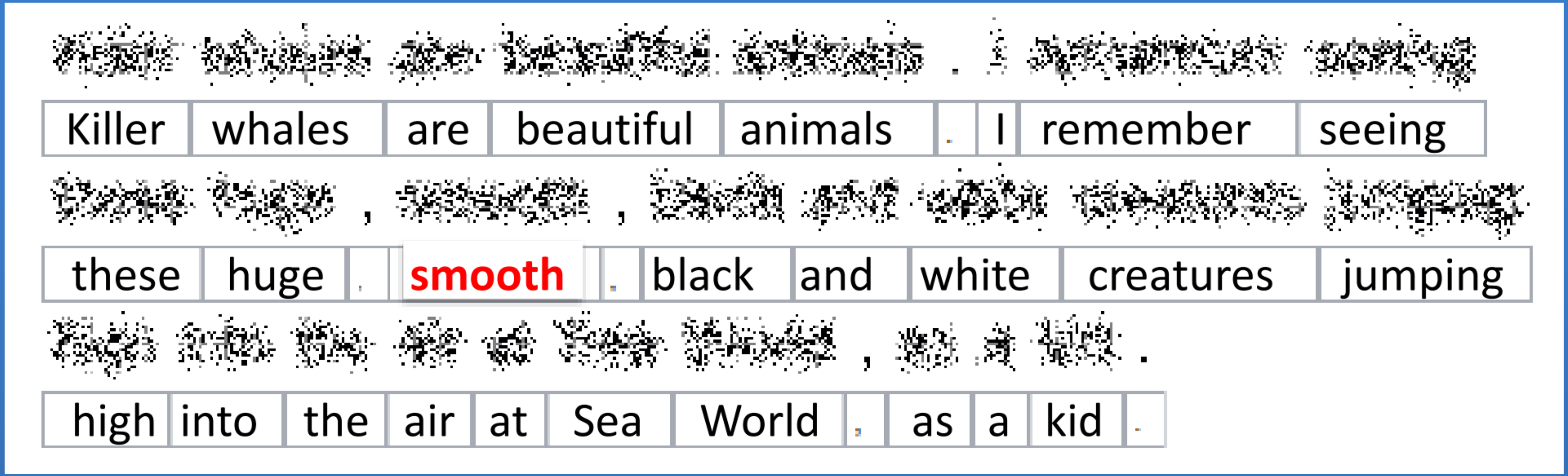
A common alternative strategy is to ask multiple workers to answer the same question, aggregate responses, and then evaluate each individual's agreement with the aggregate. This poses a fundamental tradeoff in *plurality*: asking more workers to answer the same question increases aggregate accuracy at the cost of increased redundancy. Also, unlike use of expert gold, it cannot safeguard against systematic crowd bias or crowd collusion. Most pertinent in this work, this strategy is difficult to employ in an *online* setting because it is unrealistic to assume that all workers assigned a given example will label it at the same time, or that a worker would happily wait for all others to complete the task before anyone could proceed to the next task (Jung 2014).

We consider how to best estimate a temporal model of worker performance when supervision is more realistically limited. Intuitively, if we have only a smaller sample of gold questions with which to check worker correctness, our estimate of worker accuracy will have larger variance (i.e., increased uncertainty).

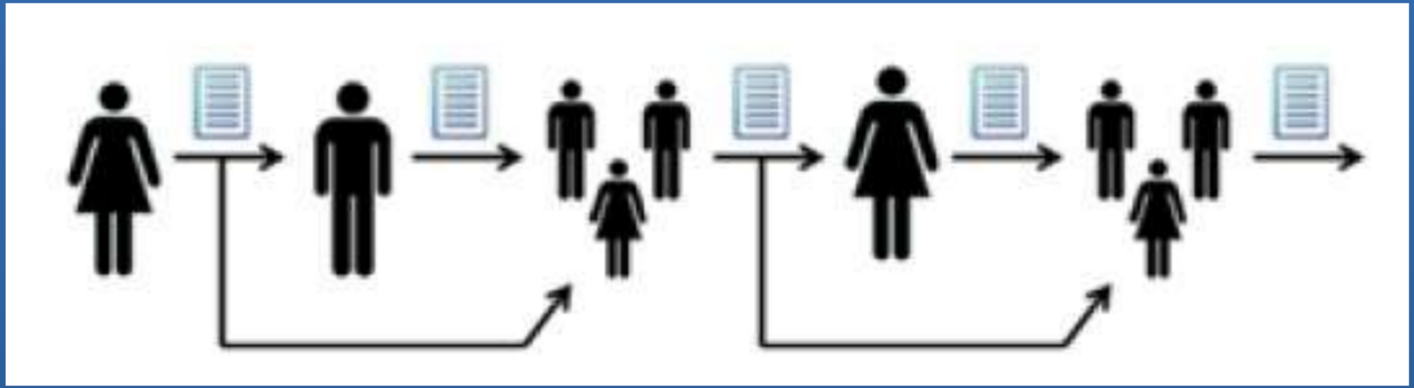
Methodology To solve this problem, we explore how to

Looking Ahead

Multi-layer crowdsourcing and human computation



Greg Little, Lydia B. Chilton, Max Goldman, and Robert C. Miller. "Exploring iterative and parallel human computation processes." In *Proceedings of the ACM SIGKDD workshop on human computation*, pp. 68-76. ACM, 2010.



Customer Reviews

★★★★☆ 18

3.8 out of 5 stars

5 star		28%
4 star		39%
3 star		22%
2 star		11%
1 star		0%

Share your thoughts with other customers

Write a customer review

See all 18 customer reviews ›

Top Customer Reviews

★★★☆☆ **Read the 1/5 about deliberation, leave the rest.**

By Review Guy on June 13, 2007

Format: Hardcover | Verified Purchase

In the 1960's, legal scholars discovered what the rest of us always knew: that pure legal scholarship is really, really boring. Law and economics demonstrated that a multidisciplinary approach could breath fresh life into the corpse of law. Then, suddenly, all the rock star law professors were interdisciplinarians. And along with this devaluation of pure legal thought came a general loss of intellectual rigor. By the 1990's, celebrity law professors were becoming like journalists with really good grades, each writing outside of his or her area of competence with an astonishing self-confidence. Richard Posner, who was on relatively solid ground in economics, crowned himself an expert on military intelligence. Lawrence Lessig wrote a whole series of books without any thesis or logical argument. And this new breed of scholar seemed to be in a race to publish as much as possible as quickly as possible, without regard for quality.

I have always thought that Cass Sunstein epitomizes the worst of this trend. He seems to rush a book into print every six months, and with each new work drifts further and further away from "law." But after hearing him on Russ Roberts' fantastic EconTalk podcast, I was genuinely dying to read this book. The topics chosen are all fascinating, and no one has really treated them all under one roof before.

The problem is that, once again, Sunstein has given short shrift to these topics. All of them, with the exception of group deliberation, has been covered better elsewhere. Where Sunstein is not stealing the limelight from people like Robin Hanson (prediction markets) he is rehashing the pop science books of people like James Surowieki (statistical group judgments). Read more ›

Comment | 43 people found this helpful. Was this review helpful to you? Yes No Report abuse

★★★★☆ **Complements Wikinomics, Solid but Incomplete**

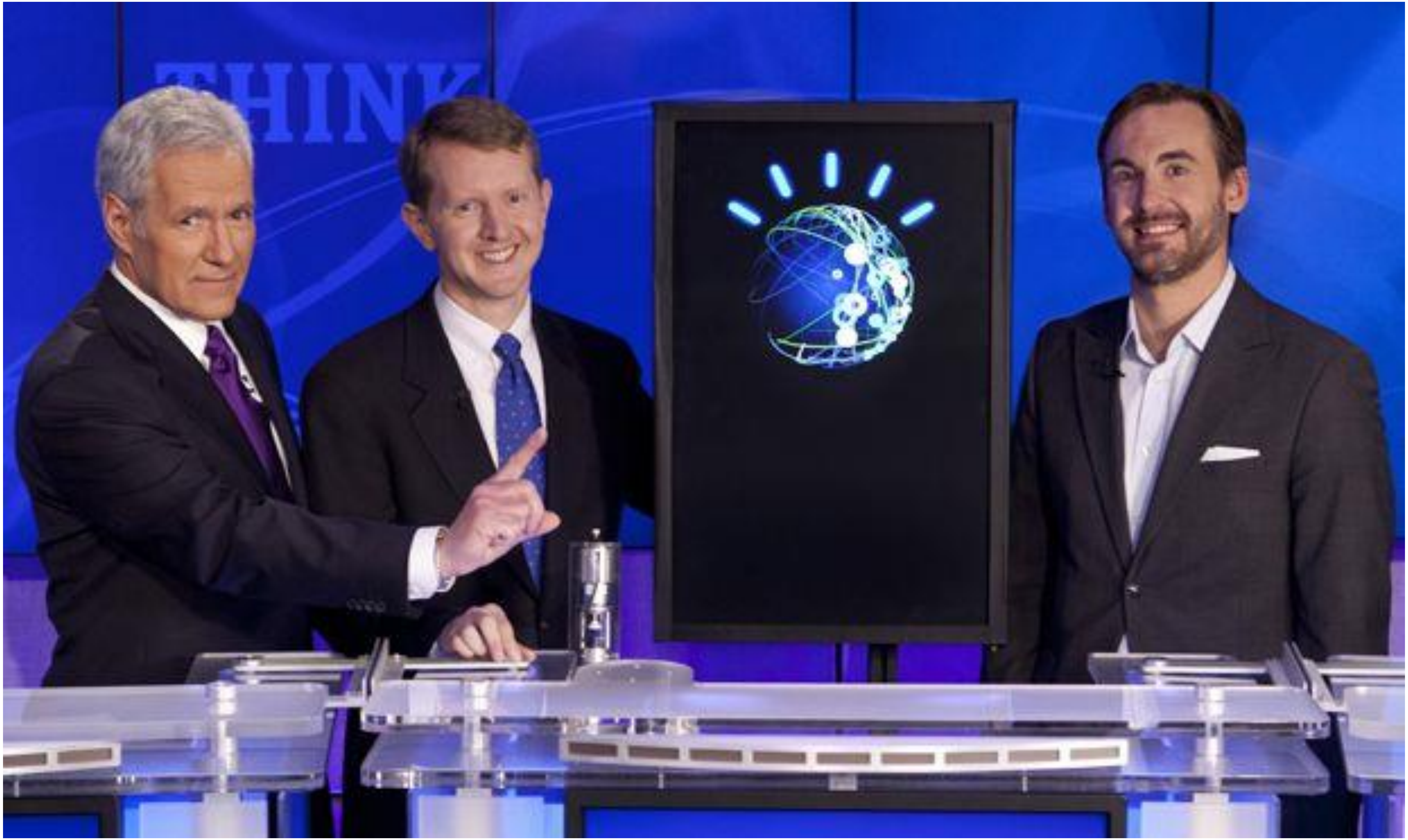
By Robert David STEELE Vivas HALL OF FAME TOP 1000 REVIEWER on January 17, 2007

Format: Hardcover | Verified Purchase

I was initially disappointed, but adjusted my expectations when I reminded myself that the author is at root a lawyer. The bottom line on this book is that it provided a very educated and well-footnoted discourse the nature and prospects for group deliberation, but there are three *huge* missing pieces:

1) Education as the necessary continuous foundation for deliberation

2) Collective Intelligence as an emerging discipline (see the Innovators spread sheet at Earth Intelligence Network); and


THINK

Looking Ahead

Programming models and tools

TurKit: Human Computation Algorithms on Mechanical Turk

Greg Little[1], *Lydia B. Chilton*[2], *Max Goldman*[1], *Robert C. Miller*[1]

[1]MIT CSAIL
{glittle, maxg, rcm}@mit.edu

[2]University of Washington
hmslydia@cs.washington.edu

ABSTRACT
Mechanical Turk provides an on-demand source of human computation. This provides a tremendous opportunity to explore algorithms which incorporate human computation as a function call. However, various systems challenges make this difficult in practice, and most uses of Mechanical Turk post large numbers of independent tasks. TurKit is a toolkit for prototyping and exploring truly algorithmic human computation, while maintaining a straight-forward imperative programming style. We present the crash-and-rerun programming model that makes TurKit possible, along with a variety of applications for human computation algorithms. We also present a couple case studies of TurKit used for real experiments outside our lab.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Prototyping.

General terms: Algorithms, Design, Experimentation

Keywords: Human computation, Mechanical Turk, toolkit

INTRODUCTION
Amazon's Mechanical Turk (MTurk) is a popular web ser-

```
ideas = []
for (var i = 0; i < 5; i++) {
    idea = mturk.prompt(
        "What's fun to see in New York City?
         Ideas so far: " + ideas.join(", "))
    ideas.push(idea)
}

ideas.sort(function (a, b) {
    v = mturk.vote("Which is better?", [a, b])
    return v == a ? -1 : 1
})
```

Figure 1: Naturally, a programmer wants to write an algorithm to help them visit New York City. TurKit lets them use Mechanical Turk as a function call to generate ideas and compare them.

general, this paper considers *human computation algorithms*, where an algorithm coordinates the contributions of humans toward some goal.

CrowdDB: Answering Queries with Crowdsourcing

Michael J. Franklin
AMPLab, UC Berkeley
franklin@cs.berkeley.edu

Donald Kossmann
Systems Group, ETH Zurich
donaldk@inf.ethz.ch

Tim Kraska
AMPLab, UC Berkeley
kraska@cs.berkeley.edu

Sukriti Ramesh
Systems Group, ETH Zurich
ramess@student.ethz.ch

Reynold Xin
AMPLab, UC Berkeley
rxin@cs.berkeley.edu

ABSTRACT

Some queries cannot be answered by machines only. Processing such queries requires human input for providing information that is missing from the database, for performing computationally difficult functions, and for matching, ranking, or aggregating results based on fuzzy criteria. CrowdDB uses human input via crowdsourcing to process queries that neither database systems nor search engines can adequately answer. It uses SQL both as a language for posing complex queries and as a way to model data. While CrowdDB leverages many aspects of traditional database systems, there are also important differences. Conceptually, a major change is that the traditional closed-world assumption for query processing does not hold for human input. From an implementation perspective, human-oriented query operators are needed to solicit, integrate and cleanse crowdsourced data. Furthermore, performance and cost depend on a number of new factors including worker affinity, training, fatigue, motivation and location. We describe the design of CrowdDB, report on an initial set of experiments using Amazon Mechanical Turk, and outline important avenues for future work in the development of crowdsourced query processing systems.

assumptions about the correctness, completeness and unambiguity of the data they store. When these assumptions fail to hold, relational systems will return incorrect or incomplete answers to user questions, if they return any answers at all.

1.1 Power to the People

One obvious situation where existing systems produce wrong answers is when they are missing information required for answering the question. For example, the query:

```
SELECT market_capitalization FROM company
WHERE name = "I.B.M.";
```

will return an empty answer if the company table instance in the database at that time does not contain a record for "I.B.M.". Of course, in reality, there are many reasons why such a record may be missing. For example, a data entry mistake may have omitted the I.B.M. record or the record may have been inadvertently deleted. Another possibility is that the record was entered incorrectly, say, as "I.B.N."

Traditional systems can erroneously return an empty result even

```
CREATE TABLE picture (
  p IMAGE,
  subject STRING
);
SELECT p FROM picture
WHERE subject = "Golden Gate Bridge"
ORDER BY CROWDORDER(p,
"Which picture visualizes better %subject");
```

PPLib: Toward the Automated Generation of Crowd Computing Programs Using Process Recombination and Auto-Experimentation

PATRICK M. DE BOER and ABRAHAM BERNSTEIN, University of Zurich

Crowdsourcing is increasingly being adopted to solve simple tasks such as image labeling and object tagging, as well as more complex tasks, where crowd workers collaborate in processes with interdependent steps. For the whole range of complexity, research has yielded numerous patterns for coordinating crowd workers in order to optimize crowd accuracy, efficiency, and cost. Process designers, however, often don't know which pattern to apply to a problem at hand when designing new applications for crowdsourcing.

In this article, we propose to solve this problem by systematically exploring the design space of complex crowdsourced tasks via automated recombination and auto-experimentation for an issue at hand. Specifically, we propose an approach to finding the optimal process for a given problem by defining the deep structure of the problem in terms of its abstract operators, generating all possible alternatives via the (re)combination of the abstract deep structure with concrete implementations from a Process Repository, and then establishing the best alternative via auto-experimentation.

To evaluate our approach, we implemented PPLib (pronounced "People Lib"), a program library that allows for the automated recombination of known processes stored in an easily extensible Process Repository. We evaluated our work by generating and running a plethora of process candidates in two scenarios on Amazon's Mechanical Turk followed by a meta-evaluation, where we looked at the differences between the two evaluations. Our first scenario addressed the problem of text translation, where our automatic recombination produced multiple processes whose performance almost matched the benchmark established by an expert translation. In our second evaluation, we focused on text shortening; we automatically generated 41 crowd process candidates, among them variations of the well-established Find-Fix-Verify process. While Find-Fix-Verify performed well in this setting, our recombination engine produced five processes that repeatedly yielded better results. We close the article by comparing the two settings where the Recombinator was used, and empirically show that the individual processes performed differently in the two settings, which led us to contend that there is no unifying formula, hence emphasizing the necessity for recombination.

CCS Concepts: • **Human-centered computing** → **Human computer interaction (HCI)**; **Computer supported cooperative work**; Collaborative content creation; Open source software

Additional Key Words and Phrases: Human computation algorithms

ACM Reference Format:
Patrick M. de Boer and Abraham Bernstein. 2016. PPLib: Toward the automated generation of crowd computing programs using process recombination and auto-experimentation. ACM Trans. Intell. Syst. Technol. 7, 4, Article 49 (April 2016), 20 pages.
DOI: http://dx.doi.org/10.1145/2897367

Modeling, Enacting, and Integrating Custom Crowdsourcing Processes

STEFANO TRANQUILLINI, FLORIAN DANIEL, PAVEL KUCHERBAEV,
and FABIO CASATI, University of Trento, Italy

Crowdsourcing (CS) is the outsourcing of a unit of work to a crowd of people via an open call for contributions. Thanks to the availability of online CS platforms, such as Amazon Mechanical Turk or CrowdFlower, the practice has experienced a tremendous growth over the past few years and demonstrated its viability in a variety of fields, such as data collection and analysis or human computation. Yet it is also increasingly struggling with the inherent limitations of these platforms: each platform has its own *logic* of how to crowdsource work (e.g., marketplace or contest), there is only very little support for *structured* work (work that requires the coordination of multiple tasks), and it is hard to *integrate* crowdsourced tasks into state-of-the-art business process management (BPM) or information systems.

We attack these three shortcomings by (1) developing a flexible CS platform (we call it *Crowd Computer*, or CC) that allows one to program custom CS logics for individual and structured tasks, (2) devising a BPMN–based modeling language that allows one to program CC intuitively, (3) equipping the language with a dedicated visual editor, and (4) implementing CC on top of standard BPM technology that can easily be integrated into existing software and processes. We demonstrate the effectiveness of the approach with a case study on the crowd-based mining of mashup model patterns.

Categories and Subject Descriptors: H.3.5 [**Information Systems**]: Online Information Services—*Web-based services*; H.4.1 [**Information Systems**]: Office Automation—*Workflow management*; H.1.2 [**Information Systems**]: User/Machine Systems—*Human information processing*

General Terms: Design, Languages, Human Factors

Additional Key Words and Phrases: Crowdsourcing, processes, tactics, Crowd Computer, BPMN4Crowd

ACM Reference Format:
Stefano Tranquillini, Florian Daniel, Pavel Kucherbaev, and Fabio Casati. 2015. Modeling, enacting, and integrating custom crowdsourcing processes. ACM Trans. Web 9, 2, Article 7 (May 2015), 43 pages.
DOI: http://dx.doi.org/10.1145/2746353

1. INTRODUCTION

Crowdsourcing (CS) is a relatively new approach to execute work that requires human capabilities. Howe [2008], who coined the term, defines *crowdsourcing* generically as "the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call." In principle, work could therefore be outsourced in a variety of ways, such as by temporarily recruiting volunteers to help complete a given job or by distribut-

ReLauncher: Crowdsourcing Micro-Tasks Runtime Controller

Pavel Kucherbaev, Florian Daniel, Stefano Tranquillini, Maurizio Marchese
University of Trento - DISI
Via Sommarive 9, 38123 Povo (TN), Italy
{lastname}@disi.unitn.it

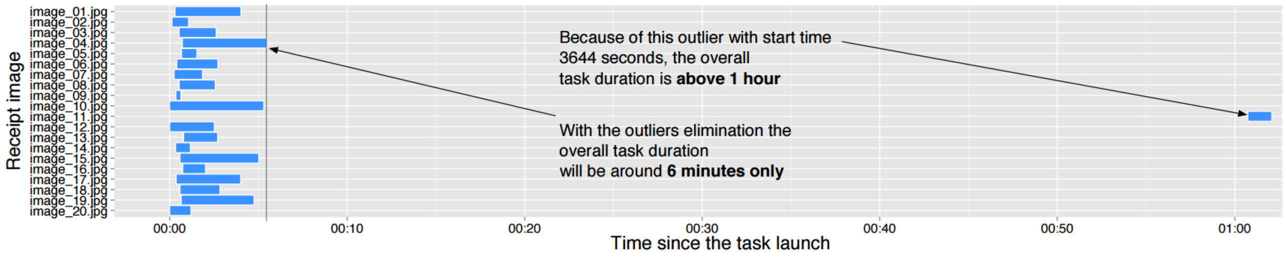


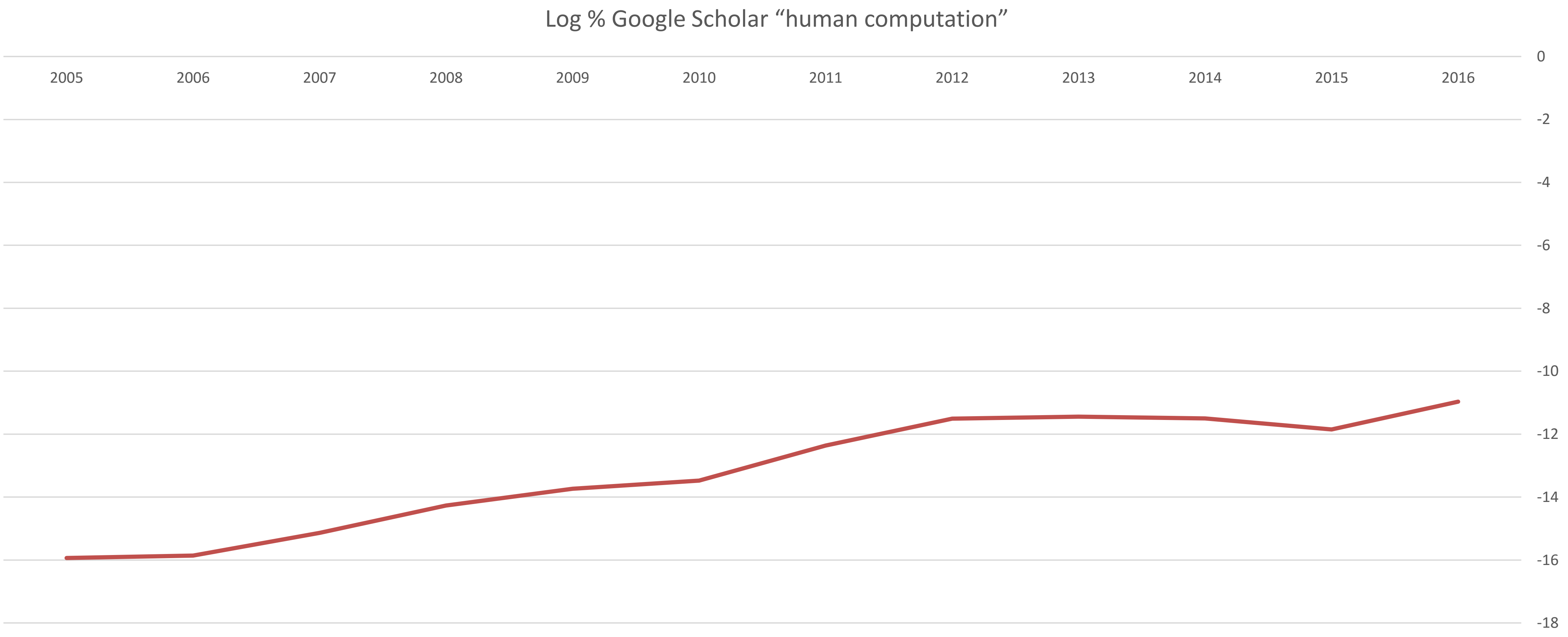
Figure 1. The timeline of the receipt transcription task launched on CrowdFlower with 20 data units (receipt photos). Each data unit is assigned to a single worker. The assignment on the very right for the unit "image_11.jpg" was finished only in 1 hour after the launch of the task, because the crowdsourcing platform kept it reserved for a certain time for a worker who left it without finishing it.

ABSTRACT

Task execution timeliness, i.e., the completion of a task within a given time frame, is a known open issue in crowdsourcing. While running tasks on crowdsourcing platforms a requester experiences long tails in execution caused by abandoned assignments (those left by workers unfinished), which become available for other workers only after some expiration time (e.g., 30 minutes in CrowdFlower). These abandoned assignments result in significant delays and a poor predictability of the overall task execution time. In this paper, we propose an approach and an implementation called ReLauncher to identify such abandoned assignments and relaunch them

INTRODUCTION

Crowdsourcing is the outsourcing of a unit of work to a crowd of people via an open call for contributions [8]. Thanks to the availability of online crowdsourcing platforms, such as Amazon Mechanical Turk (MTurk), CrowdFlower and many others [12, 14], the practice has experienced a tremendous growth over the last few years [11]. Crowdsourcing demonstrated its viability in a variety of different fields, such as data collection and analysis or human computation – all practices that use so-called *micro-tasks*, which ask workers to complete simple assignments (e.g., label an image or translate a sentence) in exchange for an optional reward (e.g., few cents or


Log % Google Scholar “human computation”
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
0
-2
-4
-6
-8
-10
-12
-14
-16
-18