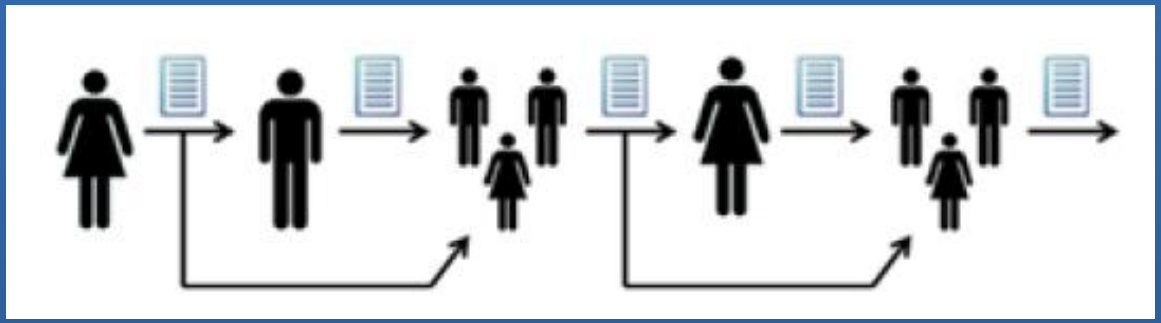
Lecture 1:
CS 6306 / INFO 6306:
Advanced Human Computation

Haym Hirsh
haym.hirsh@cornell.edu

Killer	whales	are	beautiful	animals	.	I	remember	seeing
these	huge	,	**smooth**	,	black	and	white	creatures	jumping
high	into	the	air	at	Sea	World	,	as	a	kid	.

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.



Human Computation

- New research area:
  - Most papers are less than 15 years old
  - (Ideas go back centuries)
- Definition?
  - Awkward with new fields

Human Computation

- Human computation is “a paradigm for utilizing human processing power to solve problems that computers cannot yet solve.”
- “In this paradigm, we treat human brains as processors in a distributed system, each performing a small part of a massive computation.”

Human Computation, Luis von Ahn,
Doctoral Dissertation, Department of Computer Science,
Carnegie Mellon University (2005).

Human Computation

- Human computation systems "can be defined as intelligent systems that organize humans to carry out the process of computation."

Human Computation, Law and von Ahn 2011

Human Computation

- Human computation:
  - The problems fit the general paradigm of computation, and as such might someday be solvable by computers.
  - The human participation is directed by the computational system or process

Quinn and Bederson, “Human Computation: A Survey and Taxonomy of a Growing Field”, CHI 2011

Human Computation

- Writing programs that turn to people as if they were subroutines to do things that we don’t know how to get computers to do (yet).

Human Computation

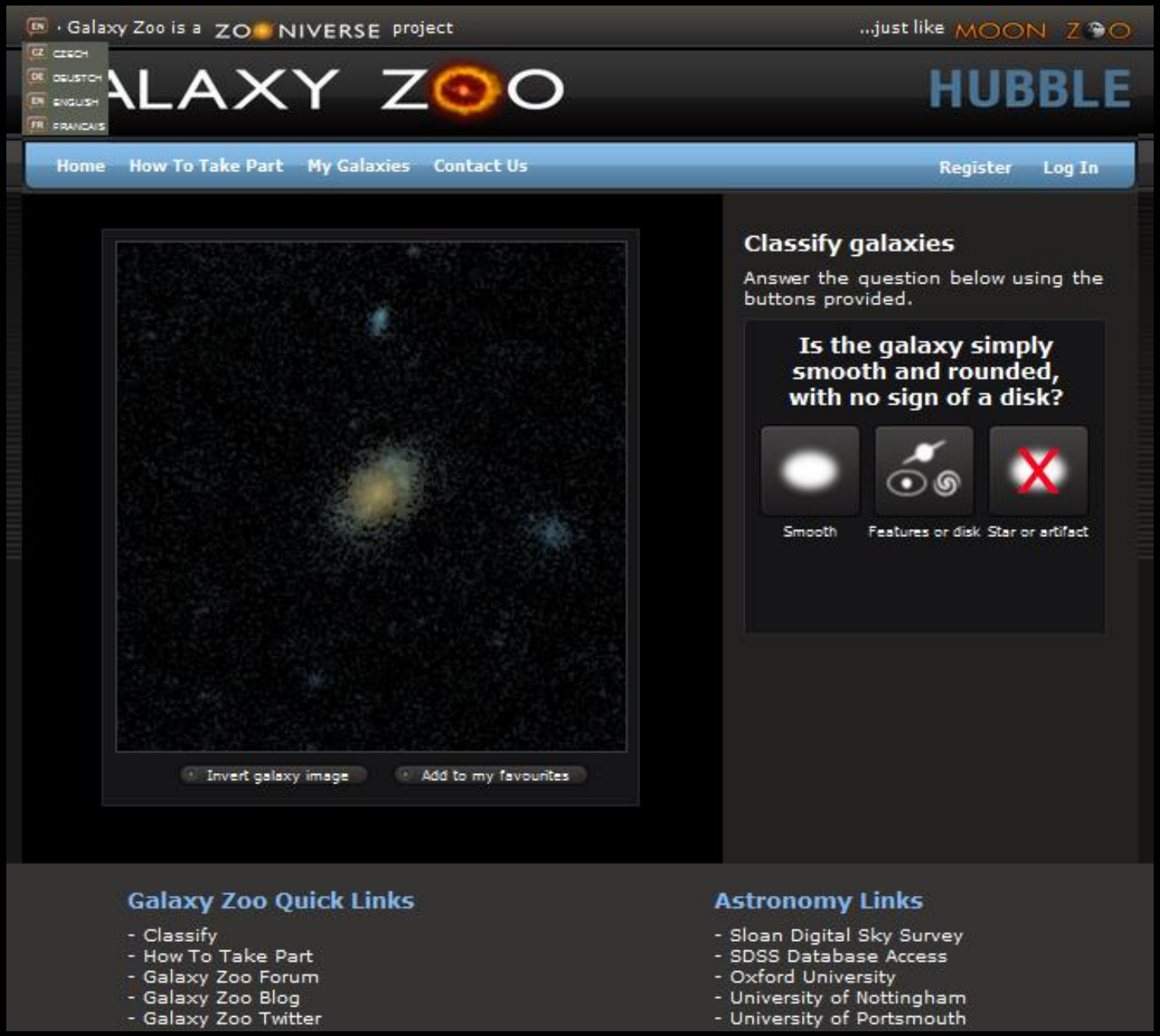
Thinking computationally about organized human labor

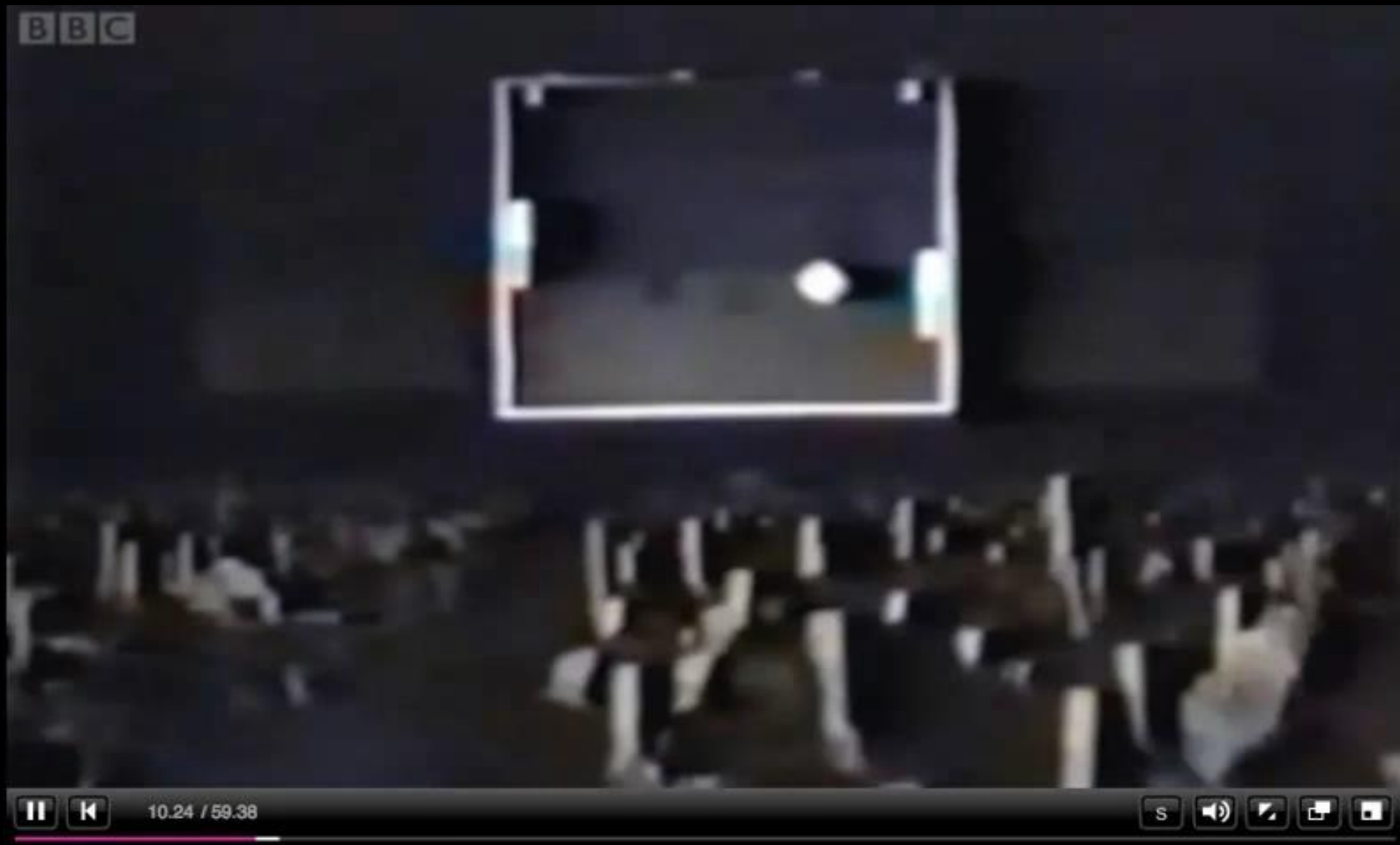
- Algorithms
- Abstractions
- Performance measures
  (correctness, accuracy, efficiency, cost, …)
- System building tools
- ….

WHEN COMPUTERS WERE HUMAN

David Alan Grier


· Galaxy Zoo is a ZOONIVERSE project
...just like MOON ZOO
GALAXY ZOO
HUBBLE
Home
The Story So Far
How To Take Part
Classify Galaxies
Explore Galaxies
The Science
FAQ
Forum
Blog
Contact Us
Pictures
Welcome to Galaxy Zoo, where you can help astronomers explore the Universe
Galaxy Zoo: Hubble uses gorgeous imagery of hundreds of thousands
Classifier Log In
Click here to log in


· Galaxy Zoo is a ZOONIVERSE project
..just like MOON ZOO
CZECH
DEUTSCH
ENGLISH
FRANCAIS
ALAXY ZOO
HUBBLE
Home
How To Take Part
My Galaxies
Contact Us
Register
Log In
Invert galaxy image
Add to my favourites
Classify galaxies
Answer the question below using the buttons provided.
Is the galaxy simply smooth and rounded, with no sign of a disk?
Smooth
Features or disk
Star or artifact
Galaxy Zoo Quick Links
- Classify
- How To Take Part
- Galaxy Zoo Forum
- Galaxy Zoo Blog
- Galaxy Zoo Twitter
Astronomy Links
- Sloan Digital Sky Survey
- SDSS Database Access
- Oxford University
- University of Nottingham
- University of Portsmouth


BBC
10.24 / 59.38

ARTICLE PREVIEW

view full access options

Journal home | Current issue | For authors | Subscribe | E-alert sign up | RSS feed

NATURE | LETTER

previous abstract next abstract

Predicting protein structures with a multiplayer online game

Seth Cooper, Firas Khatib, Adrien Treuille, Janos Barbero, Jeehyung Lee, Michael Beenen, Andrew Leaver-Fay, David Baker, Zoran Popović & Foldit players

Affiliations | Contributions | Corresponding authors

Nature **466**, 756–760 (05 August 2010) | doi:10.1038/nature09304
Received 22 January 2010 | Accepted 30 June 2010

People exert large amounts of problem-solving effort playing computer games. Simple image- and text-recognition tasks have been successfully 'crowd-sourced' through games[1, 2, 3], but it is not clear if more complex scientific problems can be solved with human-directed computing. Protein structure prediction is one such problem: locating the biologically relevant native conformation of a protein is a formidable computational challenge given

full text
日本語要約
print
email
download citation

Editor's summary

Many hands make light work

A natural polypeptide chain can fold into a native protein in microseconds, but predicting such stable three-dimensional structure from any given amino-acid sequence and first physical principles

ARTICLE PREVIEW
view full access options

Journal home | Current issue | For authors
Subscribe | E-alert sign up | RSS feed

NATURE | LETTER

previous abstract next abstract

Predicting protein structures with a multiplayer online game

Seth Cooper, Firas Khatib, Adrien Treuille, Janos Barbero, Jeehyung Lee, Michael Beenen, Andrew Leaver-Fay, David Baker, Zoran Popović & Foldit players

Affiliations | Contributions | Corresponding authors

Nature **466**, 756–760 (05 August 2010) | doi:10.1038/nature09304
Received 22 January 2010 | Accepted 30 June 2010

People exert large amounts of problem-solving effort playing computer games. Simple image- and text-recognition tasks have been successfully 'crowd-sourced' through games[1, 2, 3], but it is not clear if more complex scientific problems can be solved with human-directed computing. Protein structure prediction is one such problem: locating the biologically relevant native conformation of a protein is a formidable computational challenge given

full text
日本語要約
print
email
download citation

Editor's summary

Many hands make light work

A natural polypeptide chain can fold into a native protein in microseconds, but predicting such stable three-dimensional structure from any given amino-acid sequence and first physical principles

duolingo

Site language: English

Login

Learn a language for free. Forever.

Get started

Spanish French German Italian Portuguese Dutch Irish

reCAPTCHA™

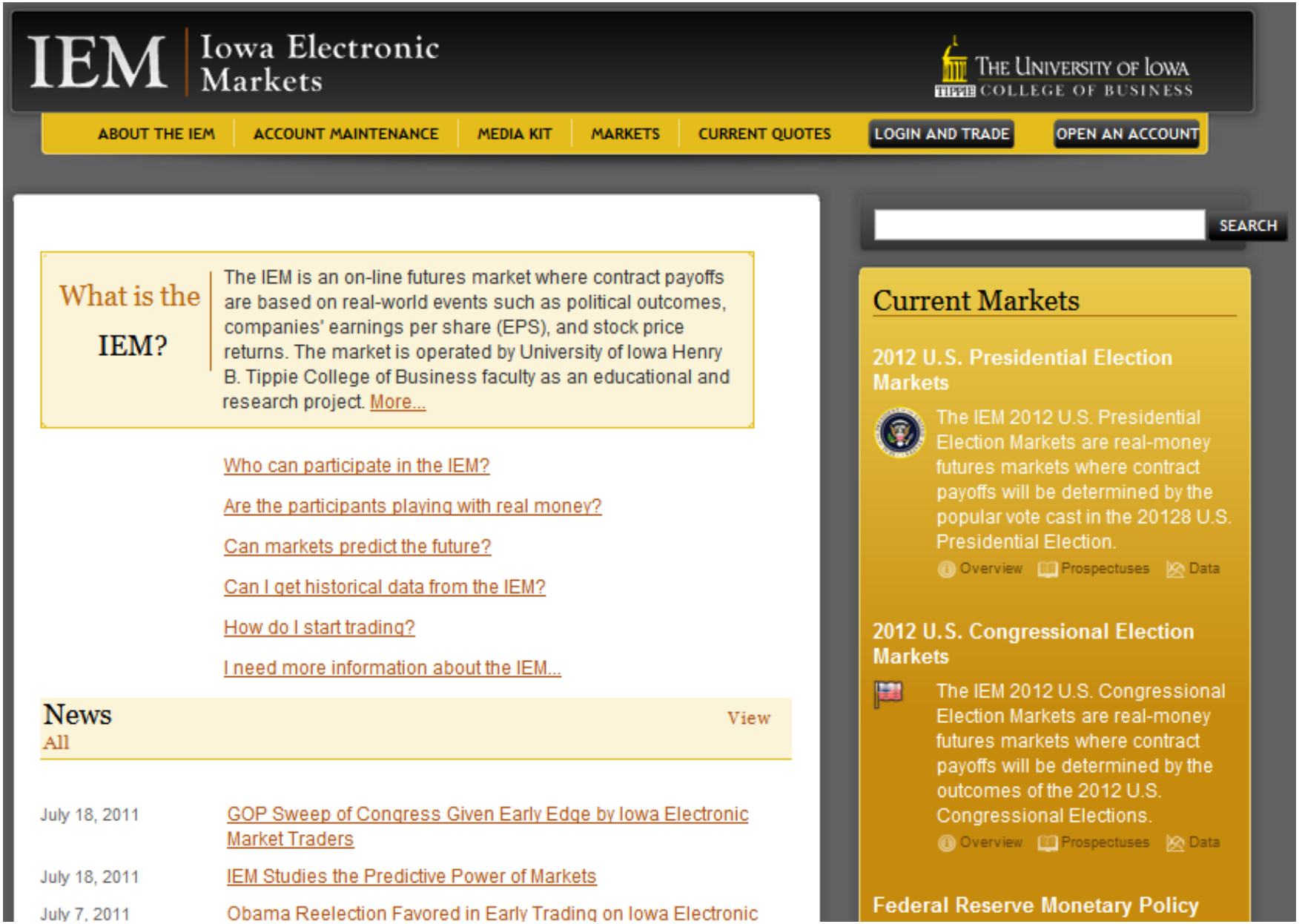
- WHAT IS reCAPTCHA
- GET reCAPTCHA
- PROTECT YOUR EMAIL
- MY ACCOUNT
- RESOURCES: DOCS & PLUGINS

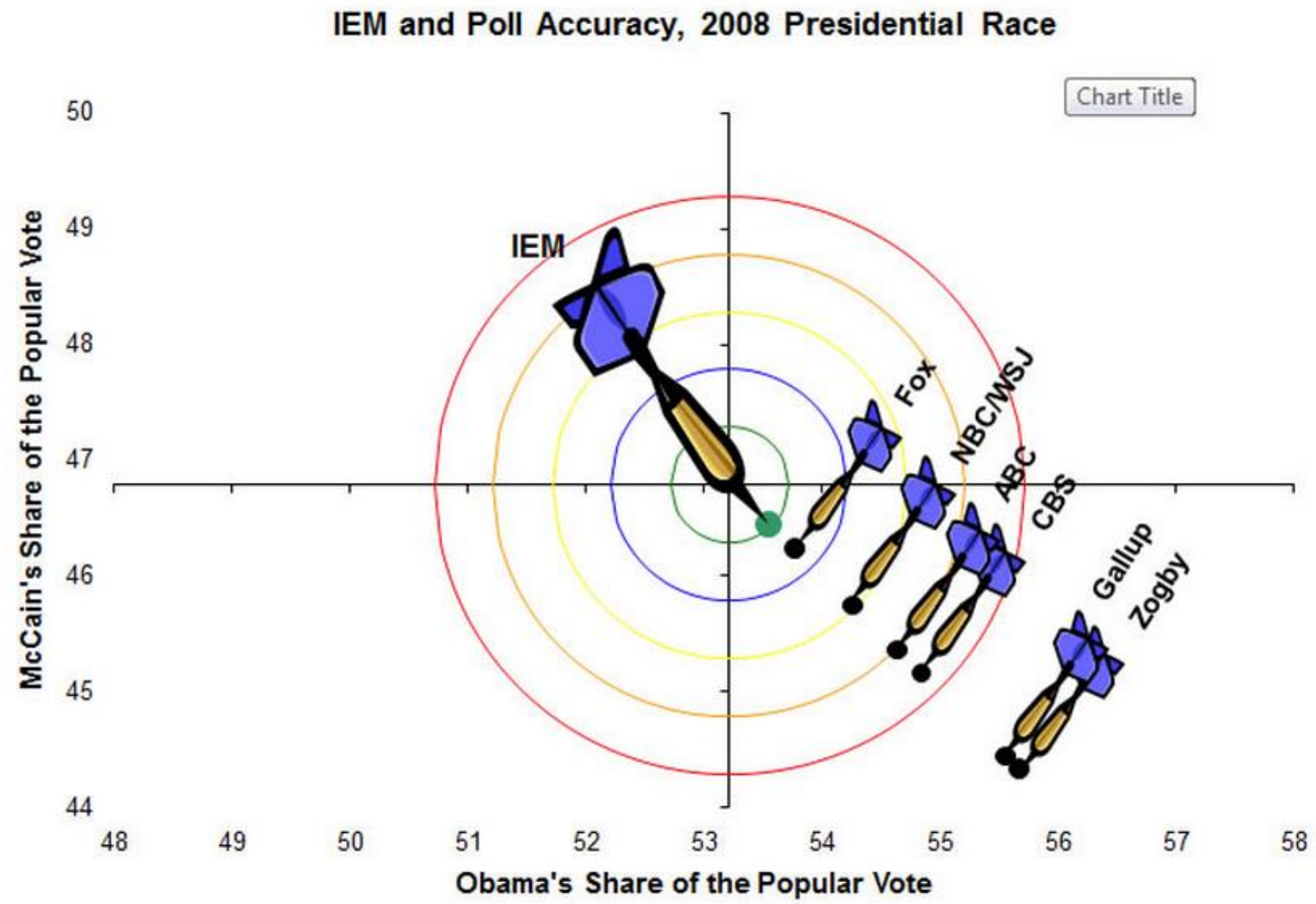
reCAPTCHA IS A FREE ANTI-BOT SERVICE THAT HELPS DIGITIZE BOOKS.

- LEARN HOW reCAPTCHA WORKS

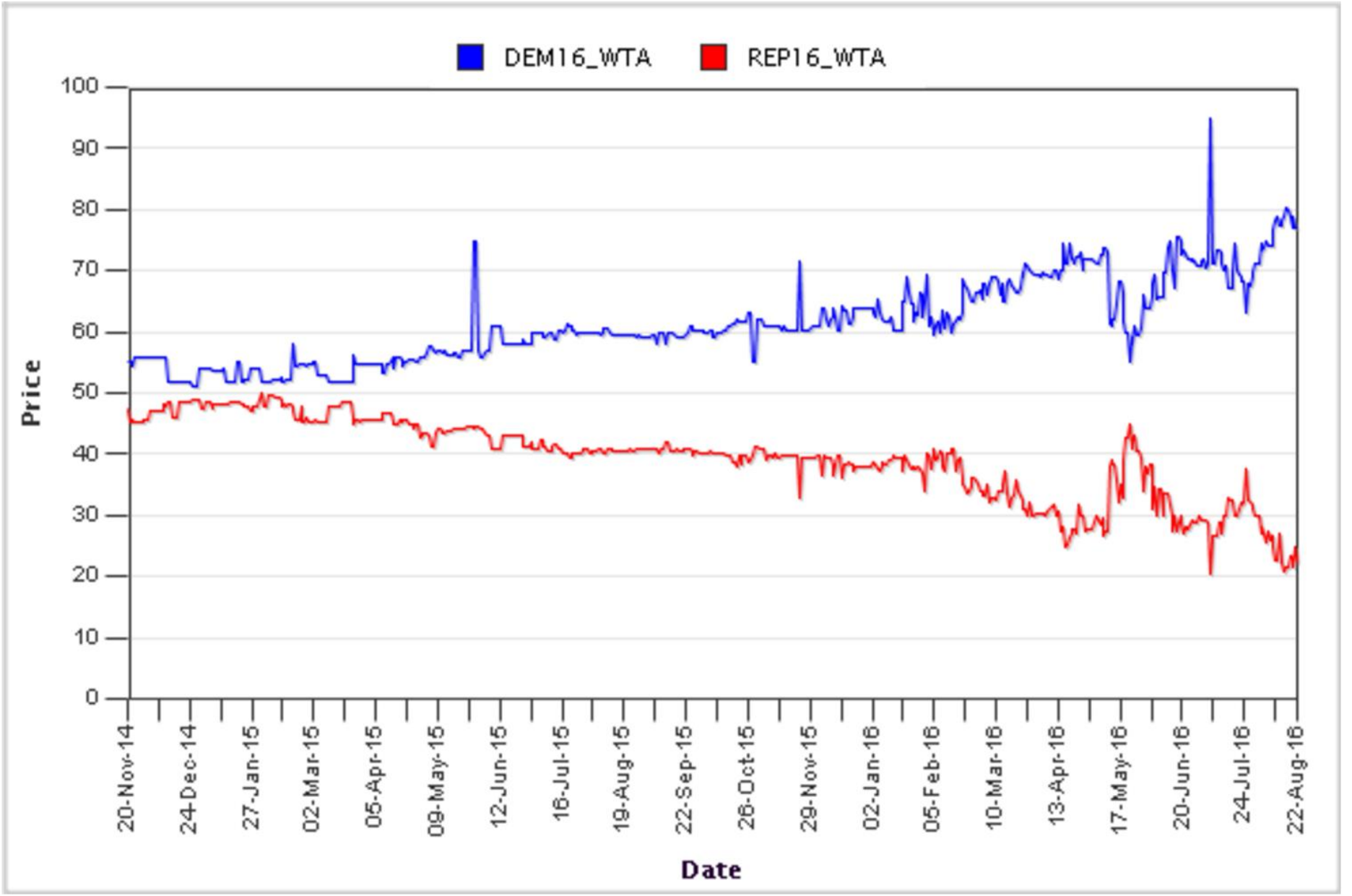
USE reCAPTCHA ON YOUR SITE

- STRONG SECURITY
- ACCESSIBLE TO BLIND USERS
- 30+ MILLION SERVED DAILY


IEM
Iowa Electronic Markets
The University of Iowa
College of Business
ABOUT THE IEM
ACCOUNT MAINTENANCE
MEDIA KIT
MARKETS
CURRENT QUOTES
LOGIN AND TRADE
OPEN AN ACCOUNT
SEARCH
What is the IEM?
The IEM is an on-line futures market where contract payoffs are based on real-world events such as political outcomes, companies' earnings per share (EPS), and stock price returns. The market is operated by University of Iowa Henry B. Tippie College of Business faculty as an educational and research project. More...
Who can participate in the IEM?
Are the participants playing with real money?
Can markets predict the future?
Can I get historical data from the IEM?
How do I start trading?
I need more information about the IEM...
News
All
View
July 18, 2011
GOP Sweep of Congress Given Early Edge by Iowa Electronic Market Traders
July 18, 2011
IEM Studies the Predictive Power of Markets
July 7, 2011
Obama Reelection Favored in Early Trading on Iowa Electronic
Current Markets
2012 U.S. Presidential Election Markets
The IEM 2012 U.S. Presidential Election Markets are real-money futures markets where contract payoffs will be determined by the popular vote cast in the 20128 U.S. Presidential Election.
Overview
Prospectuses
Data
2012 U.S. Congressional Election Markets
The IEM 2012 U.S. Congressional Election Markets are real-money futures markets where contract payoffs will be determined by the outcomes of the 2012 U.S. Congressional Elections.
Overview
Prospectuses
Data
Federal Reserve Monetary Policy


IEM and Poll Accuracy, 2008 Presidential Race
Chart Title
IEM
Fox
NBC/WSJ
ABC
CBS
Gallup
Zogby
McCain's Share of the Popular Vote
Obama's Share of the Popular Vote
50
49
48
47
46
45
44
48
49
50
51
52
53
54
55
56
57
58

Pres16_WTA
2016 US Presidential Election Winner Takes All Market




FEDERAL RESERVE NOTE
FL 39923504 C
L12
THE UNITED STATES OF AMERICA
100
FL 39923504 C
THIS NOTE IS LEGAL TENDER
FOR ALL DEBTS, PUBLIC AND PRIVATE
SERIES 2003 A
ONE HUNDRED DOLLARS
FRANKLIN

Leaderboard

10.05%

Display top 20 leaders.

Rank	Team Name	Best Score	% Improvement	Last Submit Time
1	BellKor's Pragmatic Chaos	0.8558	10.05	2009-06-26 18:42:37
Grand Prize - RMSE <= 0.8563				
2	PragmaticTheory	0.8582	9.80	2009-06-25 22:15:51
3	BellKor in BigChaos	0.8590	9.71	2009-05-13 08:14:09
4	Grand Prize Team	0.8593	9.68	2009-06-12 08:20:24
5	Dace	0.8604	9.56	2009-04-22 05:57:03
6	BigChaos	0.8613	9.47	2009-06-23 23:06:52

amazonmechanical turk
Artificial Artificial Intelligence

Your Account | HITs | Qualifications

Already have an account?
Sign in as a Worker | Requester

Introduction | Dashboard | Status | Account Settings

Mechanical Turk is a marketplace for work.

We give businesses and developers access to an on-demand, scalable workforce.
Workers select from thousands of tasks and work whenever it's convenient.

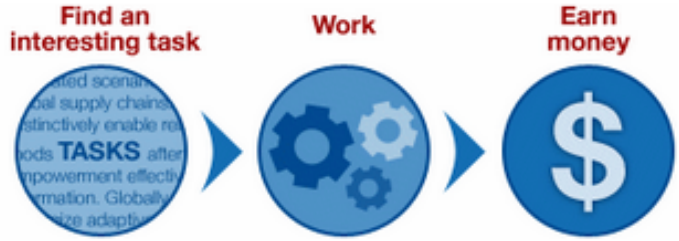
74,026 HITs available. View them now.

Make Money
by working on HITs

HITs - *Human Intelligence Tasks* - are individual tasks that you work on. Find HITs now.

As a Mechanical Turk Worker you:

- Can work from home
- Choose your own work hours
- Get paid for doing good work



Find HITs Now

or learn more about being a **Worker**

Get Results
from Mechanical Turk Workers

Ask workers to complete HITs - *Human Intelligence Tasks* - and get results using Mechanical Turk. Register Now

As a Mechanical Turk Requester you:

- Have access to a global, on-demand, 24 x 7 workforce
- Get thousands of HITs completed in minutes
- Pay only when you're satisfied with the results

Get Started

FAQ | Contact Us | Careers at Amazon | Developers | Press | Policies | Blog

An amazon.com company

Cheap and Fast — But is it Good?
Evaluating Non-Expert Annotations for Natural Language Tasks

Rion Snow[†] **Brendan O'Connor**[‡] **Daniel Jurafsky**[§] **Andrew Y. Ng**[†]

[†]Computer Science Dept.
Stanford University
Stanford, CA 94305
{rion,ang}@cs.stanford.edu

[‡]Dolores Labs, Inc.
832 Capp St.
San Francisco, CA 94110
brendano@doloreslabs.com

[§]Linguistics Dept.
Stanford University
Stanford, CA 94305
jurafsky@stanford.edu

Abstract

Human linguistic annotation is crucial for many natural language processing tasks but can be expensive and time-consuming. We explore the use of Amazon's Mechanical Turk system, a significantly cheaper and faster method for collecting annotations from a broad base of paid non-expert contributors over the Web. We investigate five tasks: affect recognition, word similarity, recognizing textual entailment, event temporal ordering, and word sense disambiguation. For all five, we show high agreement between Mechanical Turk non-expert annotations and existing gold standard labels provided by expert labelers. For the task of affect recognition, we also

and financial cost. Since the performance of many natural language processing tasks is limited by the amount and quality of data available to them (Banko and Brill, 2001), one promising alternative for some tasks is the collection of non-expert annotations.

In this work we explore the use of Amazon Mechanical Turk[1] (AMT) to determine whether non-expert labelers can provide reliable natural language annotations. We chose five natural language understanding tasks that we felt would be sufficiently natural and learnable for non-experts, and for which we had gold standard labels from expert labelers, as well as (in some cases) expert labeler agree-

Utility data annotation with Amazon Mechanical Turk

Alexander Sorokin, David Forsyth
University of Illinois at Urbana-Champaign
201 N Goodwin
Urbana,IL 61820
{sorokin2,daf}@uiuc.edu

Abstract

We show how to outsource data annotation to Amazon Mechanical Turk. Doing so has produced annotations in quite large numbers relatively cheaply. The quality is good, and can be checked and controlled. Annotations are produced quickly. We describe results for several different annotation problems. We describe some strategies for determining when the task is well specified and properly priced.

1. Introduction

Big annotated image datasets now play an important role in Computer Vision research. Many of them were built **in-house** ([18, 11, 12, 3, 13, 5] and many others). This consumes significant amounts of highly skilled labor, requires much management work, is expensive and creates a perception that annotation is difficult. Another successful strategy is to make the annotation process **completely public**

Exp	Task	img	labels	cost USD	time	effective pay/hr
1	1	170	510	$8	750m	$0.76
2	2	170	510	$8	380m	$0.77
3	3	305	915	$14	950m	$0.41[1]
4	4	305	915	$14	150m	$1.07
5	4	337	1011	$15	170m	$0.9
Total:		982	3861	$59		

Table 1. **Collected data.** In our five experiments we have collected **3861** labels for 982 distinct images for only **US $59**. In experiments 4 and 5 the throughput exceeds 300 annotations per hour even at low ($1/hour) hourly rate. We expect further increase in throughput as we increase the pay to effective market rate.

a researcher are: (1) define an annotation protocol and (2) determine what data needs to be annotated.

The annotation protocol should be implemented within an IFRAME of a web browser. We call the implementation

Get Another Label? Improving Data Quality and Data Mining Using Multiple, Noisy Labelers

Victor S. Sheng
ssheng@stern.nyu.edu

Foster Provost
fprovost@stern.nyu.edu

Panagiotis G. Ipeirotis
panos@stern.nyu.edu

Department of Information, Operations, and Management Sciences
Leonard N. Stern School of Business, New York University

ABSTRACT

This paper addresses the repeated acquisition of labels for data items when the labeling is imperfect. We examine the improvement (or lack thereof) in data quality via repeated labeling, and focus especially on the improvement of training labels for supervised induction. With the outsourcing of small tasks becoming easier, for example via Rent-A-Coder or Amazon's Mechanical Turk, it often is possible to obtain less-than-expert labeling at low cost. With low-cost labeling, preparing the unlabeled part of the data can become considerably more expensive than labeling. We present repeated-labeling strategies of increasing complexity, and show several main results. (i) Repeated-labeling can improve label quality and model quality, but not always. (ii) When labels are noisy, repeated labeling can be preferable to single labeling even in the traditional setting where labels are not particularly cheap. (iii) As soon as the cost of processing the unlabeled data is not free, even the simple strategy of labeling everything multiple times can give considerable advantage. (iv) Repeatedly labeling a

1. INTRODUCTION

There are various costs associated with the *preprocessing* stage of the KDD process, including costs of acquiring features, formulating data, cleaning data, obtaining expert labeling of data, and so on [31, 32]. For example, in order to build a model to recognize whether two products described on two web pages are the same, one must extract the product information from the pages, formulate features for comparing the two along relevant dimensions, and label product pairs as identical or not; this process involves costly manual intervention at several points. To build a model that recognizes whether an image contains an object of interest, one first needs to take pictures in appropriate contexts, sometimes at substantial cost.

This paper focuses on problems where it is possible to obtain certain (noisy) data values ("labels") relatively cheaply, from multiple sources ("labelers"). A main focus of this paper is the use of these values as training labels for supervised modeling.[1] For our two examples above, once we have constructed the unlabeled example, for relatively low cost one can obtain

Crowdsourcing User Studies With Mechanical Turk

Aniket Kittur, Ed H. Chi, Bongwon Suh
Palo Alto Research Center
3333 Coyote Hill Road, Palo Alto, CA 94304 USA
{nkittur, echi, suh}@parc.com

ABSTRACT
User studies are important for many aspects of the design process and involve techniques ranging from informal surveys to rigorous laboratory studies. However, the costs involved in engaging users often requires practitioners to trade off between sample size, time requirements, and monetary costs. Micro-task markets, such as Amazon's Mechanical Turk, offer a potential paradigm for engaging a large number of users for low time and monetary costs. Here we investigate the utility of a micro-task market for collecting user measurements, and discuss design considerations for developing remote micro user evaluation tasks. Although micro-task markets have great potential for rapidly collecting user measurements at low costs, we found that special care is needed in formulating tasks in order to harness the capabilities of the approach.

Author Keywords
Remote user study, Mechanical Turk, micro task, Wikipedia.

ACM Classification Keywords

others. Thus it is often not possible to acquire user input that is both low-cost and timely enough to impact development. The high costs of sampling additional users lead practitioners to trade off the number of participants with monetary and time costs [5].

Collecting input from only a small set of participants is problematic in many design situations. In usability testing, many issues and errors (even large ones) are not easily caught with a small number of participants [5]. In both prototyping and system validation, small samples often lead to a lack of statistical reliability, making it difficult to determine whether one approach is more effective than another. The lack of statistical rigor associated with small sample sizes is also problematic for both experimental and observational research.

These factors have led to new ways for practitioners to collect input from users on the Web, including tools for user surveys (e.g., surveymonkey.com, vividence.com), online experiments [3], and remote usability testing [2]. Such tools expand the potential user pool to anyone connected to the

Crowdsourcing Graphical Perception: Using Mechanical Turk to Assess Visualization Design

Jeffrey Heer and Michael Bostock
Computer Science Department
Stanford University
{jheer, mbostock}@cs.stanford.edu

ABSTRACT
Understanding perception is critical to effective visualization design. With its low cost and scalability, crowdsourcing presents an attractive option for evaluating the large design space of visualizations; however, it first requires validation. In this paper, we assess the viability of Amazon's Mechanical Turk as a platform for graphical perception experiments. We replicate previous studies of spatial encoding and luminance contrast and compare our results. We also conduct new experiments on rectangular area perception (as in treemaps or cartograms) and on chart size and gridline spacing. Our results demonstrate that crowdsourced perception experiments are viable and contribute new insights for visualization design. Lastly, we report cost and performance data from our experiments and distill recommendations for the design of crowdsourced studies.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces—Evaluation/Methodology

General Terms: Experimentation, Human Factors.

Keywords: Information visualization, graphical perception,

for ecological validity. Crowdsourced experiments may also substantially reduce both the cost and time to result.

Unfortunately, crowdsourcing introduces new concerns to be addressed before it is credible. Some concerns, such as ecological validity, subject motivation and expertise, apply to any study and have been previously investigated [13, 14, 23]; others, such as display configuration and viewing environment, are specific to visual perception. Crowdsourced perception experiments lack control over many experimental conditions, including display type and size, lighting, and subjects' viewing distance and angle. This loss of control inevitably limits the scope of experiments that reliably can be run. However, there likely remains a substantial subclass of perception experiments for which crowdsourcing can provide reliable empirical data to inform visualization design.

In this work, we investigate if crowdsourced experiments insensitive to environmental context are an adequate tool for graphical perception research. We assess the feasibility of using Amazon's Mechanical Turk to evaluate visualizations and then use these methods to gain new insights into visual-

Soylent: A Word Processor with a Crowd Inside

Michael S. Bernstein[1], *Greg Little*[1], *Robert C. Miller*[1],
Björn Hartmann[2], *Mark S. Ackerman*[3], *David R. Karger*[1], *David Crowell*[1], *Katrina Panovich*[1]

[1] MIT CSAIL
Cambridge, MA
{msbernst, glittle, rcm,
karger, dcrowell, kp}@csail.mit.edu

[2] Computer Science Division
University of California, Berkeley
Berkeley, CA
bjoern@cs.berkeley.edu

[3] Computer Science & Engineering
University of Michigan
Ann Arbor, MI
ackerm@umich.edu

ABSTRACT
This paper introduces architectural and interaction patterns for integrating crowdsourced human contributions directly into user interfaces. We focus on writing and editing, complex endeavors that span many levels of conceptual and pragmatic activity. Authoring tools offer help with pragmatics, but for higher-level help, writers commonly turn to other people. We thus present Soylent, a word processing interface that enables writers to call on Mechanical Turk workers to shorten, proofread, and otherwise edit parts of their documents on demand. To improve worker quality, we introduce the Find-Fix-Verify crowd programming pattern, which splits tasks into a series of generation and review stages. Evaluation studies demonstrate the feasibility of crowdsourced editing and investigate questions of reliability, cost, wait time, and work time for edits.

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

General terms: Design, Human Factors

Keywords: Outsourcing, Mechanical Turk, Crowdsourcing

INTRODUCTION
Word processing is a complex task that touches on many goals of human-computer interaction. It supports a deep cognitive activity – writing – and requires complicated manipulations. Writing is difficult: even experts routinely make style, grammar and spelling mistakes. Then, when a writer makes high-level decisions like changing a passage from past to present tense or fleshing out citation sketches into a true references section, she is faced with executing daunting numbers of nontrivial tasks across the entire document. Finally, when the document is a half-page over length, interactive software provides little support to help us trim those last few paragraphs. Good user interfaces aid these tasks; good artificial intelligence helps as well, but it is clear that we have far to go.

In our everyday life, when we need help with complex cognition and manipulation tasks, we often turn to other people. We ask friends to answer questions that we cannot answer ourselves [8]; masses of volunteer editors flag spam edits on Wikipedia [13]. Writing is no exception [7]: we commonly recruit friends and colleagues to help us shape and polish our writing. But we cannot always rely on them: colleagues do not want to proofread every sentence we write, cut a few lines from every paragraph in a ten-page paper, or help us format thirty ACM-style references.

UIST'10

As a step toward integrating this human expertise permanently into our writing tools, we present *Soylent*, a word processing interface that utilizes crowd contributions to aid complex writing tasks ranging from error prevention and paragraph shortening to automation of tasks like citation searches and tense changes. We hypothesize that crowd workers with a basic knowledge of written English can support both novice and expert writers. These workers perform tasks that the writer might not, such as scrupulously scanning for text to cut, or updating a list of addresses to include a zip code. They can also solve problems that artificial intelligence cannot, for example flagging writing errors that the word processor does not catch.

Soylent aids the writing process by integrating paid crowd workers from Amazon's Mechanical Turk platform[1] into Microsoft Word. *Soylent is people*: its core algorithms involve calls to Mechanical Turk workers (Turkers). Soylent is comprised of three main components:

1) *Shortn*, a text shortening service that cuts selected text down to 85% of its original length typically without changing the meaning of the text or introducing errors.
2) *Crowdproof*, a human-powered spelling and grammar checker that finds problems Word misses, explains the problems, and suggests fixes.
3) *The Human Macro*, an interface for offloading arbitrary word processing tasks such as formatting citations or finding appropriate figures.

The main contribution of this paper is *the idea of embedding paid crowd workers in an interactive user interface to support complex cognition and manipulation tasks on demand.* These crowd workers do tasks that computers cannot reliably do automatically and the user cannot easily script. This paper contributes the design of one such system, an implementation embedded in Microsoft Word, and a programming pattern that increases the reliability of paid crowd workers on complex tasks. We expand these contributions with feasibility studies of the performance, cost, and time delay of our three main components and a discus-

[1] http://www.mturk.com

VizWiz: Nearly Real-time Answers to Visual Questions

Jeffrey P. Bigham[†], *Chandrika Jayant*[ϕ], *Hanjie Ji*[†], *Greg Little*[§], *Andrew Miller*[γ], *Robert C. Miller*[§], *Robin Miller*[†], *Aubrey Tatarowicz*[§], *Brandyn White*[‡], *Samuel White*[†], *and Tom Yeh*[‡]

[†]University of Rochester Computer Science
Rochester, NY 14627 USA
{jbigham, hji, rmill13, swhite24}@cs.rochester.edu

[‡]University of Maryland Computer Science
College Park, MD 20742 USA
{bwhite, tomyeh}@umiacs.umd.edu

[§]MIT CSAIL
Cambridge, MA 02139 USA
{altat, glittle, rcm}@mit.edu

[γ]University of Central Florida CS
Orlando, FL 32816 USA
amiller@ucf.edu

[ϕ]University of Washington CSE
Seattle, WA 98195 USA
cjayant@cs.washington.edu

ABSTRACT

The lack of access to visual information like text labels, icons, and colors can cause frustration and decrease independence for blind people. Current access technology uses automatic approaches to address some problems in this space, but the technology is error-prone, limited in scope, and quite expensive. In this paper, we introduce *VizWiz*, a talking application for mobile phones that offers a new alternative to answering visual questions in nearly real-time—asking multiple people on the web. To support answering questions quickly, we introduce a general approach for intelligently recruiting human workers in advance called *quikTurkit* so that workers are available when new questions arrive. A field deployment with 11 blind participants illustrates that blind people can effectively use VizWiz to cheaply answer questions in their everyday lives, highlighting issues that automatic approaches will need to address to be useful. Finally, we illustrate the potential of using VizWiz as part of the participatory design of advanced tools by using it to build and evaluate VizWiz::LocateIt, an interactive mobile tool that helps blind people solve general visual search problems.

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

General terms: Human Factors, Design, Experimentation

Keywords: Real-Time Human Computation, Non-Visual Interfaces, Blind Users

INTRODUCTION

Our environment often assumes the ability to see. Food products otherwise indistinguishable are labeled with their contents, color relays semantics, and currency denominations are differentiable only by the writing on them[1]. A quick visual scan helps stop minor problems from turning into big frustrations—a sighted person can tell in a glance if their clothes match before an important job interview, spot an empty picnic table at the park, or locate the restroom at the other end of the room without having to ask. Blind people often have effective, albeit inefficient, work-arounds that render individual problems into mere nuisances. Collectively, however, small problems can lead to decreased independence.

Talking mobile devices from both research and industry have been designed to help blind people solve visual problems in their everyday lives, but current automatic approaches are not yet up to the task. Products designed for blind people are specialized for a few functions, are prone to errors, and are usually quite expensive. As an example, the popular Kurzweil knfbReader software ($1000 USD) uses optical character recognition (OCR) to convert text to speech in pictures taken by users on their mobile devices [16]. When it works, this product offers the independence of reading printed material anywhere, but unfortunately, OCR cannot yet reliably identify the text in many real-world situations, such as the graphic labels on many products, a hand-written menu in a coffee shop, or even the street name on a street sign. Other popular products identify colors and read barcodes with similar performance (and prices). Filling the remaining void are a large number of human workers, volunteers, and friends who help blind people address remaining visual problems.

In this paper we introduce *VizWiz*, a project aimed at enabling blind people to recruit remote sighted workers to help them with visual problems in nearly real-time. Blind people use VizWiz on their existing camera phones. Users take a picture with their phone, speak a question, and then receive multiple spoken answers. Currently, answers are provided by workers on Amazon Mechanical Turk [1]. Prior work has demonstrated that such services need to work quickly [22], and so we have developed an approach (and accompanying implementation) called quikTurkit that provides a layer of abstraction on top of Mechanical Turk to intelligently recruit multiple workers before they are needed. In a field deployment, users had to wait just over 2 minutes to get their first answer on average, but wait times decreased sharply when questions and photos were easy for workers to understand. Answers were returned at an average cost per question of only $0.07 USD for 3.3 answers. Given that many tools in this domain cost upwards of $1000 USD (the equivalent of

[1]Many currencies other than the US Dollar are tactually distinguishable.

UIST'10, October 3-6, 2010, New York City, USA.

PlateMate: Crowdsourcing Nutrition Analysis from Food Photographs

Jon Noronha, Eric Hysen, Haoqi Zhang, Krzysztof Z. Gajos
Harvard School of Engineering and Applied Sciences
33 Oxford St., Cambridge, MA 02138, USA
{noronha,hysen,hqz,kgajos}@seas.harvard.edu

ABSTRACT
We introduce PlateMate, a system that allows users to take photos of their meals and receive estimates of food intake and composition. Accurate awareness of this information can help people monitor their progress towards dieting goals, but current methods for food logging via self-reporting, expert observation, or algorithmic analysis are time-consuming, expensive, or inaccurate. PlateMate crowdsources nutritional analysis from photographs using Amazon Mechanical Turk, automatically coordinating untrained workers to estimate a meal's calories, fat, carbohydrates, and protein. We present the Management framework for crowdsourcing complex tasks, which supports PlateMate's nutrition analysis workflow. Results of our evaluations show that PlateMate is nearly as accurate as a trained dietitian and easier to use for most users than traditional self-reporting.

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

General terms: Design, Human Factors

Keywords: Human computation, Crowdsourcing, Mechanical Turk, Nutrition, Remote Food Photography

INTRODUCTION
The majority of Americans perceive healthy eating as complicated [5]. Seeking comprehensible and actionable advice, Americans spend over $40 billion each year on diets and self-help books [18], but achieve little success: the majority eventually regain any lost weight and more [13].

There are many factors that may impact successful long-term change in eating habits. Our work is based on the observation that food intake monitoring is a popular component of many diets. For people who make a commitment to changing their eating habits, accurate logs of what they eat may help in monitoring progress toward set goals [11]. Currently, food logging is typically done by hand using paper diaries, spreadsheets, or a growing number of specialized applications. This process is both time-consuming and error-prone [17, 6]. Nutritionists have explored alternative methods such as daily interviews with trained experts. While these methods improve accuracy, they are costly and still require substantial time investment.

Our work is inspired by the *Remote Food Photography Method* (RFPM) [16], a novel approach from the nutrition literature. Rather than remembering foods or writing down records, users take two photographs of each meal: one at the beginning of the meal and one at the end documenting the leftovers. These images are analyzed by a third party, making logging easier and discouraging self-deception. The challenge is in finding a qualified third party without prohibitive costs. Expert nutritionists are scarce and costly, limiting the system to wealthy users or patients with particular conditions.

To make accurate food logging easier and more affordable, we introduce *PlateMate*, a system for crowdsourcing nutritional analysis (calories, fat, carbohydrates, and protein) from photographs of meals using Amazon Mechanical Turk. Complex tasks like this are hard problems for crowdsourcing, as workers may vary drastically in experience and reliability. To achieve accurate estimates, we propose a workflow in which the overall problem is decomposed into small, manageable, and verifiable steps. PlateMate uses this workflow to assign tasks to contributors, to validate and combine results, and to appropriately route tasks for further processing.

This paper makes three main contributions:

1. We present PlateMate, an end-to-end system for crowdsourced nutrition analysis from food photographs.
2. We discuss the results of a two-part evaluation, which suggests PlateMate can be as accurate as experts and self-report methods, and more usable than manual logging for everyday use.
3. We introduce the Management framework—inspired by the structure of human organizations, it provides effective support for managing crowdsourcing of complex heterogeneous tasks.

PlateMate implements the first step in the Remote Food Photography Method. In the last section we suggest how it can be extended to also support the second step: the analysis of photographs of food waste.

UIST'11, October 16-19, 2011, Santa Barbara, CA, USA.

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.

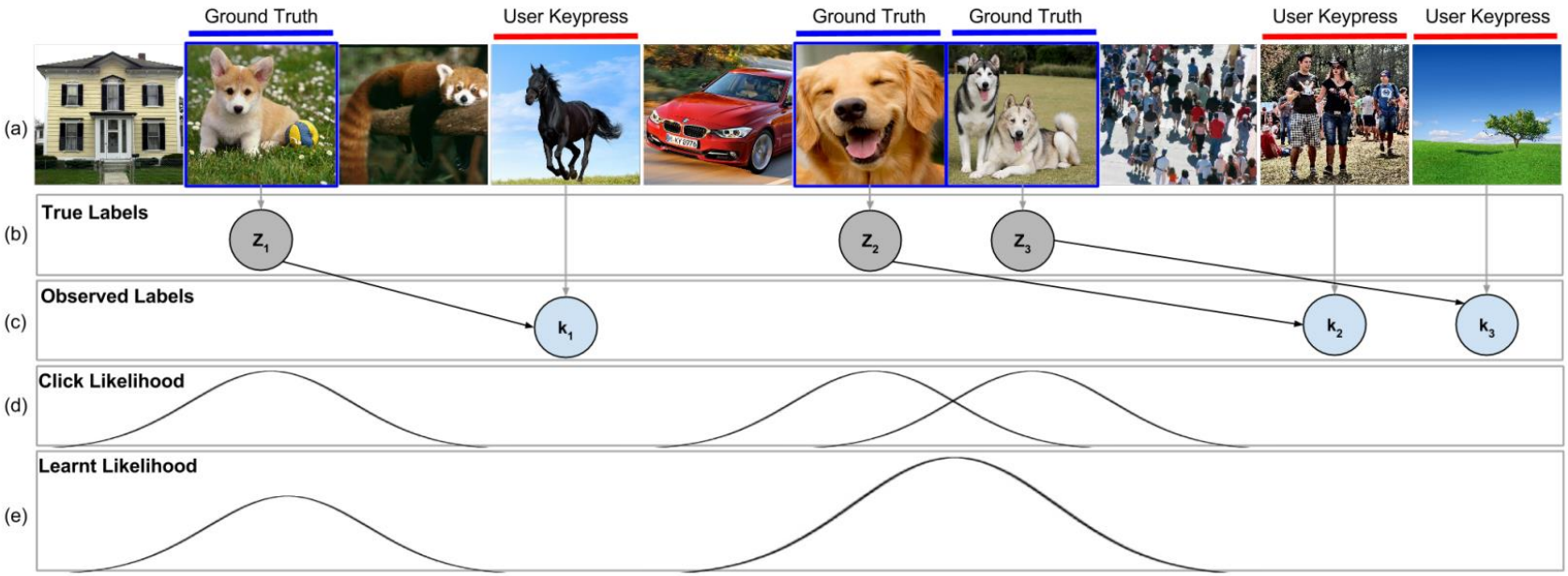
Author Keywords
Human computation; Crowdsourcing; RSVP

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

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

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):

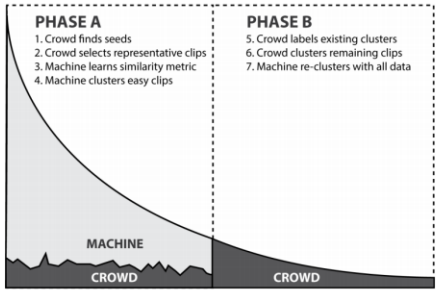


Figure 1. A conceptual overview of the system. In the first phase, crowd workers identify seed clips to train a machine learning model, which is used to classify the "head" of the distribution. In the second phase, crowd workers classify the more difficult items in the "tail". A machine learning backbone 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, 61,

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 allows the user to track and approve completed crowd

Fine-grained Categorization and Dataset Bootstrapping using Deep Metric Learning with Humans in the Loop

Yin Cui[1,2] Feng Zhou[3] Yuanqing Lin[3] Serge Belongie[1,2]

[1]Department of Computer Science, Cornell University [2]Cornell Tech [3]NEC Labs America

[1,2]{ycui, sjb}@cs.cornell.edu [3]{feng, ylin}@nec-labs.com

Abstract

Existing fine-grained visual categorization methods often suffer from three challenges: lack of training data, large number of fine-grained categories, and high intra-class vs. low inter-class variance. In this work we propose a generic iterative framework for fine-grained categorization and dataset bootstrapping that handles these three challenges. Using deep metric learning with humans in the loop, we learn a low dimensional feature embedding with anchor points on manifolds for each category. These anchor points capture intra-class variances and remain discriminative between classes. In each round, images with high confidence scores from our model are sent to humans for labeling. By comparing with exemplar images, labelers mark each candidate image as either a "true positive" or a "false positive." True positives are added into our current dataset and false positives are regarded as "hard negatives" for our metric learning model. Then the model is retrained with an expanded dataset and hard negatives for the next round. To demonstrate the effectiveness of the proposed framework, we bootstrap a fine-grained flower dataset with 620 categories from Instagram images. The proposed deep metric learning scheme is evaluated on both our dataset and the CUB-200-2001 Birds dataset. Experimental evaluations show significant performance gain using dataset bootstrapping and demonstrate state-of-the-art results achieved by the proposed deep metric learning methods.

1. Introduction

Fine-grained visual categorization (FGVC) has received increased interest from the computer vision community in recent years. By definition, FGVC, as a sub-field of object recognition, aims to distinguish subordinate categories within an entry-level category. For example, in fine-grained flower categorization [33, 34, 3], we want to identify the species of a flower in an image, such as "nelumbo nucifera (lotus flower)," "tulip" or "cherry blossom." Other examples include classifying different types of plants [28], birds [7, 6], dogs [24], insects [30], galaxies [13, 11]; recognizing brand, model and year of cars [26, 46, 48]; and face identification [39, 36].

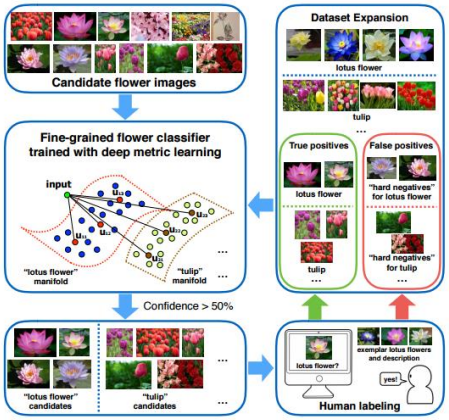


Figure 1. Overview of the proposed framework. Using deep metric learning with humans in the loop, we learn a low dimensional feature embedding for each category that can be used for fine-grained visual categorization and iterative dataset bootstrapping.

Most existing FGVC methods fall into a classical two-step scheme: feature extraction followed by classification [1, 5, 8, 35]. Since these two steps are independent, the performance of the whole system is often suboptimal compared with an end-to-end system using Convolutional Neural Networks (CNN) that can be globally optimized via back-propagation [6, 50, 25, 32]. Therefore, in this work, we focus on developing an end-to-end CNN-based method for FGVC. However, compared with general purpose visual categorization, there are three main challenges arising when

A Human Computation Framework for Boosting Combinatorial Solvers

Ronan Le Bras, Yexiang Xue, Richard Bernstein, Carla P. Gomes, Bart Selman

Computer Science Department
Cornell University
Ithaca, NY 14853

Abstract

We propose a general framework for boosting combinatorial solvers through human computation. Our framework combines insights from human workers with the power of combinatorial optimization. The combinatorial solver is also used to guide requests for the workers, and thereby obtain the most useful human feedback quickly. Our approach also incorporates a problem decomposition approach with a general strategy for discarding incorrect human input. We apply this framework in the domain of materials discovery, and demonstrate a speedup of over an order of magnitude.

Introduction

The past decade has witnessed the rapid emergence of the field of human computation, along with numerous successful applications. Human computation is motivated by problems for which automated algorithms cannot yet exceed human performance (Von Ahn 2005). Indeed, some tasks are naturally and truly easy for humans, while they remain surprisingly challenging for machines. These problems typically involve a perceptual or cognitive component. For example, successful applications with a strong visual recognition component include the ESP game (Von Ahn and Dabbish 2004), Peekaboom (Von Ahn, Liu, and Blum 2006), and Eyespy (Bell et al. 2009), while TagATune (Law and Von Ahn 2009) and the Listen Game (Turnbull et al. 2007) make extensive use of the human ability to perceive and recognize sounds. In addition, human computation applications might exploit the implicit, background or commonsense knowledge of humans, as it is the case for Verbosity (Von Ahn, Kedia, and Blum 2006) and the Common Consensus system (Lieberman, Smith, and Teeters 2007). Recent developments have also demonstrated how to successfully exploit the wisdom of crowds by combining user annotations for image labeling (Welinder et al. 2010; Zhou et al. 2012) or clustering tasks (Gomes et al. 2011; Yi et al. 2012; Chen et al. 2010).

The work described in this paper is motivated by application domains that involve visual and audio tasks. In particular, we focus on a central problem in combinatorial materials discovery. New materials will help us address some of the key challenges our society faces today, in terms of finding a path towards a sustainable planet (White 2012; Patel 2011). In combinatorial materials discovery, scientists experimentally explore large numbers of combinations of different elements with the hope of finding new compounds with interesting properties, e.g., for efficient fuel cells or solar cell arrays. We are collaborating with two teams of materials scientists, the Energy Materials Center at Cornell (emc2) and the Joint Center for Artificial Photosynthesis (JCAP) at Caltech. An overall goal is to develop the capability of analyzing data from over one million new materials samples per day. Automated data analysis tools, boosted with a human computation component, will be key to the success of this project.

We consider a central task in combinatorial materials discovery, namely the problem of identifying the crystalline phases of inorganic compounds based on an analysis of high-intensity X-ray patterns. In our approach, we integrate a state-of-the-art optimization framework based on constraint reasoning with a human computation component. This hybrid framework reduces our analysis time by orders of magnitude compared to running the constraint solver by itself. The human input also helps us improve the quality of solutions. For the human computation component, we developed a relatively simple and appealing visual representation of the X-ray images using heat maps (i.e. color-coded graphical representations of 2-D real-valued data matrices), which allows us to decompose the problem into manageable Human Intelligence Tasks (HITs), involving the identification of simple visual patterns, requiring no prior knowledge about materials science.

This work is part of our broader research agenda focused on *harnessing human insights to solve hard combinatorial problems.* Our work is close in spirit to the seminal FoldIt project for protein folding (Cooper et al. 2010). In FoldIt, human gamers are the main driving force for finding new protein folds, complemented with a limited amount of local computation (e.g., "shaking" of structures). We are proposing a framework that provides a much *tighter integration of a combinatorial solver with human insights.* Our approach takes advantage of the dramatic improvements in combinatorial solvers in recent years, allowing us to handle practical instances with millions of variables and constraints. Our objective is also to minimize the amount of required user in-

Avicaching: A Two Stage Game for Bias Reduction in Citizen Science

Yexiang Xue
Computer Science Dept
Cornell University
yexiang@cs.cornell.edu

Ian Davies, Daniel Fink, Christopher Wood
Cornell Lab of Ornithology
{id99, daniel.fink, chris.wood}@cornell.edu

Carla P. Gomes
Computer Science Dept
Cornell University
gomes@cs.cornell.edu

ABSTRACT

Citizen science projects have been very successful at collecting rich datasets for different applications. However, the data collected by the citizen scientists are often biased, more aligned with the citizens' preferences rather than scientific objectives. We introduce a novel two-stage game for reducing data-bias in citizen science in which the game *organizer*, a citizen-science program, incentivizes the *agents*, the citizen scientists, to visit under-sampled areas. We provide a novel way of encoding this two-stage game as a single optimization problem, cleverly folding (an approximation of) the agents' problems into the organizer's problem. We present several new algorithms to solve this optimization problem as well as a new structural SVM approach to learn the parameters that capture the agents' behaviors, under different incentive schemes. We apply our methodology to *eBird*, a well-established citizen-science program for collecting bird observations, as a game called *Avicaching*. We deployed Avicaching in two New York counties (March 2015), with a great response from the birding community, surpassing the expectations of the eBird organizers and bird-conservation experts. The field results show that the Avicaching incentives are remarkably effective at encouraging the bird watchers to explore under-sampled areas and hence alleviate the eBird's data bias problem.

Keywords

Two-Stage Game, Bilevel Optimization, Structural SVM, Citizen Science

1. INTRODUCTION

Over the past decade, along with the emergence of the *big data* era, the data collection process for scientific discovery has evolved dramatically. One effective way of collecting large datasets is to engage the public through citizen science projects, such as *Zooniverse*, *Cicada Hunt* and *eBird* [24, 42, 35]. The success of these projects relies on the ability to tap into the intrinsic motivations of the volunteers to make participation enjoyable [5]. Thus in order to engage large groups of participants, citizen science projects often have few restrictions, leaving many decisions about where, when, and how to collect data up to the participants. As a result, the data collected by volunteers are often biased, more aligned with their preferences, rather than providing systematic observations across various experimental settings. Moreover, since participants volunteer their effort, personal convenience is an important factor that often determines how data are collected. For spatial data, this means more searches occur in areas close to urban areas and roads (Fig. 1).

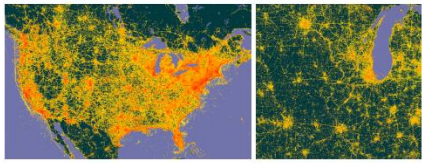


Figure 1: Highly biased distribution of *eBird* observations until 2014. (Left) continental US (Right) Zoom in Midwest US. Submissions coincide with urban areas.

We provide a general methodology to mitigate the data bias problem, as a two-stage game in which the game organizer, e.g., a citizen-science program, provides incentives to the agents, the citizen scientists, to perform more crucial scientific tasks. We apply it to *eBird*, a well-established citizen-science program for collecting bird observations, as a game called *Avicaching*.

Our proposed two-stage game is related to the Principal-Agent framework, originally studied in economics [31], and more recently in computer science [1, 17, 14], and to the Stackelberg games [13, 12, 28, 15], which also involves e.g., a *principal* or a *leader* and *agents* or *followers*. These games have been studied under different assumptions regarding the agents' preferences and computational abilities [18, 8]. In crowdsourcing, there has been related work on mechanisms to improve the crowd performance [29, 23, 21, 22, 34, 26, 37, 7]. Notable works include using incentives to promote exploration activities [16], and steering user participation with badges [3]. [32, 10, 9] discuss the optimal reward allocation to reduce the empirical risk of machine learning models.

In our two-stage game setting, the *agents* are citizen scientists maximizing their intrinsic utilities, as well as the incentives distributed by the game organizer, subject to a budget constraint. The organizer corresponds to an organization with notable influence on the citizen scientists. The

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), J. Thangarajah, K. Tuyls, C. Jonker, S. Marsella (eds.), May 9–13, 2016, Singapore.*

OpenSurfaces: A Richly Annotated Catalog of Surface Appearance

Sean Bell Paul Upchurch Noah Snavely Kavita Bala
Cornell University

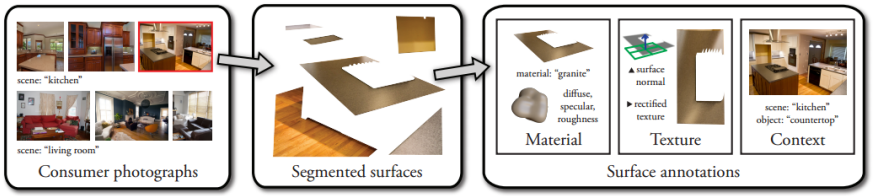


Figure 1: *We present OpenSurfaces, a large database of annotated surfaces created from real-world consumer photographs. Our annotation pipeline draws on crowdsourcing to segment surfaces from photos, and then annotates them with rich surface appearance properties, including material, texture, and contextual information.*

Abstract

The appearance of surfaces in real-world scenes is determined by the materials, textures, and context in which the surfaces appear. However, the datasets we have for visualizing and modeling rich surface appearance in context, in applications such as home remodeling, are quite limited. To help address this need, we present OpenSurfaces, a rich, labeled database consisting of thousands of examples of surfaces segmented from consumer photographs of interiors, and annotated with material parameters (reflectance, material names), texture information (surface normals, rectified textures), and contextual information (scene category, and object names).

Retrieving usable surface information from uncalibrated Internet photo collections is challenging. We use human annotations and present a new methodology for segmenting and annotating materials in Internet photo collections suitable for crowdsourcing (e.g., through Amazon's Mechanical Turk). Because of the noise and variability inherent in Internet photos and novice annotators, designing this annotation engine was a key challenge; we present a multi-stage set of annotation tasks with quality checks and validation. We demonstrate the use of this database in proof-of-concept applications including surface retexturing and material and image browsing, and discuss future uses. OpenSurfaces is a public resource available at http://opensurfaces.cs.cornell.edu/.

CR Categories: I.4.6 [Image Processing and Computer Vision]: Scene Analysis—Photometry, Shading I.4.6 [Image Processing and Computer Vision]: Feature Measurement—Texture;

Keywords: materials, reflectance, textures, crowdsourcing

Links: DL PDF WEB

1 Introduction

The rich appearance of objects and surfaces in real-world scenes is determined by the materials, textures, shape, and context in which the surfaces appear. An everyday room, such as a kitchen, can include a wide range of surfaces, including granite countertops, shiny hardwood floors, brushed metal appliances, and many others. Much of the perceived appeal of such scenes depends on the kinds of materials used, individually and as an ensemble. Thus, many users—ranging from homeowners, to interior designers, to 3D modelers—expend significant effort in the design, visualization, and simulation of realistic materials and textures for real or rendered scenes.

However, the tools and data that we have for exploring and applying materials and textures for everyday problems are currently quite limited. For instance, consider a homeowner planning a kitchen renovation, who would like to create a scrapbook of kitchen photographs from which to draw inspiration for materials, find appliances with a certain look, visualize paint samples, etc. Even simply finding a set of good kitchen photos to look at can be a time-consuming process. Interior design websites, such as Houzz, are starting to provide a forum where people share photos of interior scenes, tag elements such as countertops with brand names, and ask and answer questions about material design. Their popularity indicates the demand and need for better tools. For example, people want to:

- Search for examples of materials or textures that meet certain criteria (e.g. "show me kitchens that use light-colored, shiny wood floors")
- Find materials that go well with a given material ("what do people with black granite countertops tend to use for their kitchen cabinets?")
- Retexture a surface in a photo with a new material ("what would my tiled kitchen look like with a hardwood floor?")
- Edit the material parameters of a surface in a photograph, ("how would my wood table look with fresh varnish?")
- Automatically recognize materials in a photograph, or find where one could buy the materials online (search-by-texture).

To support these kinds of tasks, we present OpenSurfaces, a large, rich database of annotated surfaces (including material, texture and context information), collected from real-world photographs via crowdsourcing. As shown in Figure 1, each surface is segmented from an input Internet photograph and labeled with material information (a named category, e.g., "wood" or "metal", and reflectance

Interactive Consensus Agreement Games For Labeling Images

Paul Upchurch and **Daniel Sedra** and **Andrew Mullen** and **Haym Hirsh** and **Kavita Bala**
Computing and Information Science, Cornell University

Abstract

Scene understanding algorithms in computer vision are improving dramatically by training deep convolutional neural networks on millions of accurately annotated images. Collecting large-scale datasets for this kind of training is challenging, and the learning algorithms are only as good as the data they train on. Training annotations are often obtained by taking the majority label from independent crowdsourced workers using platforms such as Amazon Mechanical Turk. However, the accuracy of the resulting annotations can vary, with the hardest-to-annotate samples having prohibitively low accuracy.

Our insight is that in cases where independent worker annotations are poor more accurate results can be obtained by having workers collaborate. This paper introduces consensus agreement games, a novel method for assigning annotations to images by the agreement of multiple consensuses of small cliques of workers. We demonstrate that this approach reduces error by 37.8% on two different datasets at a cost of $0.10 or $0.17 per annotation. The higher cost is justified because our method does not need to be run on the entire dataset. Ultimately, our method enables us to more accurately annotate images and build more challenging training datasets for learning algorithms.

Introduction

Creating large-scale image datasets has proved crucial to enabling breakthrough performance on computer vision tasks (Krizhevsky, Sutskever, and Hinton 2012). A significant barrier to the creation of such datasets has been the human labor required to accurately annotate large collections of images. Increasingly such datasets have been labeled through innovations in the area of crowdsourcing and human computation, whereby the efforts of large numbers of often inexpert Internet-based workers are used to yield data of surprising accuracy (Deng et al. 2009; Kanefsky, Barlow, and Gulick 2001; Raddick et al. 2007; Russell et al. 2008; Sorokin and Forsyth 2008; Von Ahn and Dabbish 2004; Westphal et al. 2005). The introduction of the Amazon Mechanical Turk crowdsourcing platform in 2006 in particular quickly led to its adoption in various image recognition tasks (Barr and Cabrera 2006; Douglis 2007; Sorokin and Forsyth 2008; Spain and Perona 2008; Deng et al. 2009).

The most common approach seeks labels for each item from multiple workers and assigns as the item's label the "majority vote" among those provided by the workers (Sheng, Provost, and Ipeirotis 2008; Snow et al. 2008; Sorokin and Forsyth 2008). Even as increasingly sophisticated approaches have been developed for aggregating the labels of independent workers there can still be significant variability in the quality of such data. Many samples receive very low agreement when labeled by multiple MTurk workers. For example, (Bell et al. 2013) collected approximately five labels per sample, and for those with 60% agreement (3 out of 5 agreement, after removing votes from low-quality workers) many of these *low-agreement samples* were mislabeled, making them unsuitable for training a high-accuracy model. As a result, (Bell et al. 2015) only used samples with at least 80% agreement (*high-agreement samples*).

Relying solely on high-agreement data can bias the data to easy-to-classify cases, which may not mirror the diversity of cases to which the trained model will be applied, negatively impacting the quality of the learned model. Further, low-agreement samples can represent cases that fall near decision boundaries, reflecting data that can be particularly valuable for improving model accuracy (Lewis and Gale 1994).

Ultimately, the problem is that MTurk workers have a high error rate on low-agreement data. If the *MTurk error rate* is the fraction of mislabed samples (compared to, for example, expert labelers or some other appropriate notion of ground truth), our goal is to reduce it so that low-agreement samples become more accurate, and thereby more useful for training computer vision models.

Reducing the MTurk error rate is not easy. The key characteristic of low-agreement data is that *independent* workers cannot agree on the label. Getting more answers from independent workers or encouraging them with agreement incentives does not get us better answers (as shown in the Experiments section). Instead, we take an approach where labels are assigned through a collaborative process involving multiple workers. We find our inspiration in two previous works. First, the graph consensus-finding work of (Judd, Kearns, and Vorobeychik 2010; Kearns 2012) showed that a network of workers can collectively reach consensus even when interactions are highly constrained. Next, the ESP Game (Von Ahn and Dabbish 2004) showed how to obtain labels from non-communicative workers by seeking agreement around a shared image. In this paper, we show how to label images by casting it as a graph consensus problem which seeks agreement between multiple, independent consensus-finding cliques of workers. We find this pattern to be effective on difficult-to-label images.

Human Computation
AAAI Technical Report WS-12-08

Diamonds From the Rough: Improving Drawing, Painting, and Singing via Crowdsourcing

Yotam Gingold
Departments of Computer Science
Columbia University & Rutgers University
yotam@yotamgingold.com

Etienne Vouga and **Eitan Grinspun**
Department of Computer Science
Columbia University
New York, NY
evouga,eitan@cs.columbia.edu

Haym Hirsh
Department of Computer Science
Rutgers University
Piscataway, NJ
hirsh@cs.rutgers.edu

Abstract

It is well established that in certain domains, noisy inputs can be reliably combined to obtain a better answer than any individual. It is now possible to consider the crowdsourcing of physical actions, commonly used for creative expressions such as drawing, shading, and singing. We provide algorithms for converting low-quality input obtained from the physical actions of a crowd into high-quality output. The inputs take the form of line drawings, shaded images, and songs. We investigate single-individual crowds (multiple inputs from a single human) and multiple-individual crowds.

Introduction

The *wisdom of crowds* (Surowiecki 2004) suggests that it can be advantageous to aggregate information from many "low-quality" sources rather than relying on information from a single "high-quality" source. There may be several advantages: it may be difficult or impossible to access a high-quality source; it may be cheaper to obtain information from many low-quality sources; perhaps most surprising, *aggregation may consistently produce higher-quality output*. Galton (1907) presented one of the earliest examples of this surprising result, when he calculated the median of a crowd's estimates of the weight of a bull and found it to be within 1% of the truth.

We propose to draw on the wisdom of crowds to produce a single higher-quality output from a set of lower-quality inputs. We consider the scenario where many individuals contribute a single input, as well as the scenario where a single individual contributes many inputs. We focus on *creative tasks* such as drawing, painting, and singing.

Our approach may be framed in terms of crowdsourcing and aggregation. Technology makes it possible to crowdsource physical actions, e.g., using a touch-screen or microphone. To harness this data, we must address the question of *how to meaningfully aggregate* creative works. Unlike many examples of the wisdom of crowds, our input and output data are more complex than a single number or a vote from among a small finite set of choices.

Yu and Nickerson (2011) employed genetic algorithms and tournament selection to iteratively aggregate and improve the quality of a set of drawings; the algorithm assumes that a human is able combine the best aspects of two creative pieces. By contrast, we consider settings in which this assumption does not hold.

We treat the case of *inherently low-quality* (ILQ) input. We assume that the initial human input is "as good as can be expected" for the available input hardware and software, and for the skill, level of focus, and allotted time of participating humans.

ILQ input can arise from multiple trials by single individuals (Vul and Pashler 2008), such as when a person with limited fine motor coordination makes repeated attempts to draw, write, or sign their name; the limitation may be due to disease (e.g., Parkinson's) or simply due to the limited form factor of the input device (finger-writing on a small screen). In another variation, the input may be reasonable, but an even better output is desired, such as when an average person sings or draws, but wishes they could do so better.

ILQ input can also arise from single trials across multiple individuals. For example, can we produce a great painting, if the humans and tools at our disposition limit us to only mediocre paintings? Even when we have humans and tools capable of painting expertly, economic conditions might favor participation of multiple less-skilled participants. Under a tight deadline, there may not be sufficient time for an expert to produce a great piece, but there may be sufficient time for a multitude of participants to produce mediocre pieces, or ILQ.

To explore this setting, we consider crowdsourcing and aggregation to produce better drawings, paintings, and songs from ILQ. We first analyze "smiley faces" sketched many times by the same individuals, we then aggregate similar paintings created by many individuals, and finally we analyze the same song sung many times by the same individuals.

Related Work

Crowdsourcing has been applied to algorithms and data collection in a variety of domains, including databases (Franklin et al. 2011), natural language processing (Snow et al. 2008), song identification (Huq, Cartwright, and Pardo 2010), and computer vision.

The problem of aggregating input from many (human) sources has been studied in the literature. This includes collaborative filtering (Goldberg et al. 1992; Adomavicius

Behavioral Mechanism Design: Optimal Crowdsourcing Contracts and Prospect Theory

David Easley, Cornell University
Arpita Ghosh, Cornell University

Incentive design is more likely to elicit desired outcomes when it is derived based on accurate models of agent behavior. A substantial literature in behavioral economics, however, demonstrates that individuals systematically and consistently deviate from the standard economic model—expected utility theory—for decision-making under uncertainty, which is at the core of the equilibrium analysis necessary to facilitate mechanism design. Can these behavioral biases—as modeled by *prospect theory* [Kahneman and Tversky 1979]—in agents' decision-making make a difference to the optimal design of incentives in these environments? In this paper, we explore this question in the context of markets for online labor and crowdsourcing where workers make strategic choices about whether to undertake a task, but do not strategize over quality conditional on participation. We ask what kind of incentive scheme—amongst a broad class of contracts, including those observed on major crowdsourcing platforms such as fixed prices or base payments with bonuses (as on MTurk or oDesk), or open-entry contests (as on platforms like Kaggle or Topcoder)—a principal might want to employ, and how the answer to this question depends on whether workers behave according to expected utility or prospect theory preferences.

We first show that with expected utility agents, the optimal contract—for any increasing objective of the principal—always takes the form of an *output-independent* fixed payment to some optimally chosen number of agents. In contrast, when agents behave according to prospect theory preferences, we show that a winner-take-all *contest* can dominate the fixed-payment contract, for large enough total payments, under a certain condition on the preference functions; we show that this condition is satisfied for the parameters given by the literature on econometric estimation of the prospect theory model [Tversky and Kahneman 1992; Bruhin et al. 2010]. Since these estimates are based on fitting the prospect theory model to extensive experimental data, this result provides a strong affirmative answer to our question for 'real' population preferences: a principal might indeed choose a fundamentally different kind of mechanism—an output-contingent contest versus a 'safe output-independent scheme—and do better as a result, if he accounts for deviations from the standard economic models of decision-making that are typically used in theoretical design.

Categories and Subject Descriptors: J.4 [**Computer Applications**]: Social & Behavioral Sciences.

Additional Key Words and Phrases: Contracts; Contests; Crowdsourcing; Mechanism design; Behavioral; Game theory.

1. INTRODUCTION

The vast range of systems with outcomes that depend on the choices made by economic agents has led to a rich and large literature on mechanism design, which regards designing incentives so that agents make choices resulting in 'good' outcomes. Incentives are more likely to elicit desired outcomes when they are derived based on accurate models of agent behavior. A growing literature, however, suggests that people do not quite behave like the standard economic agents in the mechanism design literature. Can such differences have significant consequences for the optimal design of incentive mechanisms?

Decision making under uncertainty: Prospect theory. A particular instance of such a difference involves behavioral biases in decision-making under uncertainty. Many, if not most, environments to which the mechanism design literature has been applied involve risky choice—agents who must

Authors addresses: D. Easley, Departments of Economics and Information Science, Cornell University, email: dae3@cornell.edu; A. Ghosh, Department of Information Science, Cornell University, email: ag865@cornell.edu.

EC'15, June 15–19, 2015, Portland, OR, USA. ACM 978-1-4503-3410-5/15/06 ...$15.00.

http://dx.doi.org/10.1145/http://dx.doi.org/10.1145/http://dx.doi.org/10.1145/2764468.2764513

In Search of a Gold Standard in Studies of Deception

Stephanie Gokhman[1], **Jeff Hancock**[1,3], **Poornima Prabhu**[2], **Myle Ott**[2], **Claire Cardie**[2,3]
Departments of Communication[1], Computer Science[2], and Information Science[3]
Cornell University, Ithaca, NY 14853
{sbg94,jth34,pmp67,mao37,ctc9}@cornell.edu

Abstract

In this study, we explore several popular techniques for obtaining corpora for deception research. Through a survey of traditional as well as non-gold standard creation approaches, we identify advantages and limitations of these techniques for web-based deception detection and offer crowdsourcing as a novel avenue toward achieving a gold standard corpus. Through an in-depth case study of online hotel reviews, we demonstrate the implementation of this crowdsourcing technique and illustrate its applicability to a broad array of online reviews.

1 Introduction

Leading deception researchers have recently argued that verbal cues are the most promising indicators for detecting deception (Vrij, 2008) while lamenting the fact that the majority of previous research has focused on nonverbal cues. At the same time, increasing amounts of language are being digitized and stored on computers and the Internet — from email, Twitter and online dating profiles to legal testimony and corporate communication. With the recent advances in natural language processing that have enhanced our ability to analyze language, researchers now have an opportunity to similarly advance our understanding of deception.

One of the crucial components of this enterprise, as recognized by the call for papers for the present workshop, is the need to develop corpora for developing and testing models of deception. To date there has not been any systematic approach for corpus creation within the deception field. In the present study, we first provide an overview of traditional approaches for this task (Section 2) and discuss recent deception detection methods that rely on non-gold standard corpora (Section 3). Section 4 introduces novel approaches for corpus creation that employ crowdsourcing and argues that these have several advantages over traditional and non-gold standard approaches. Finally, we describe an in-depth case study of how these techniques can be implemented to study deceptive online hotel reviews (Section 5).

2 Traditional Approaches

The deception literature involves a number of widely used traditional methods for gathering deceptive and truthful statements. We classify these according to whether they are *sanctioned*, in which the experimenter supplies instructions to individuals to lie or not lie, or *unsanctioned* approaches, in which the participant lies of his or her own accord.

2.1 Sanctioned Deception

The vast majority of studies examining deception employ some form of the sanctioned lie method. A common example is recruiting participants for a study on deception and randomly assigning them to a lie or truth condition. A classic example of this kind of procedure is the original study by Ekman and Friesen (1969), in which nurses were required to watch pleasant or highly disturbing movie clips. The nurses were instructed to indicate that they were watching a pleasing movie, which required the nurses watching the disturbing clips to lie about their current emotional state.

In another example, Newman et. al. (2003) ask

A Bayesian Framework for Modeling Human Evaluations

Himabindu Lakkaraju* Jure Leskovec* Jon Kleinberg† Sendhil Mullainathan‡

Abstract

Several situations that we come across in our daily lives involve some form of *evaluation*: a process where an *evaluator* chooses a correct label for a given *item*. Examples of such situations include a crowd-worker labeling an image or a student answering a multiple-choice question. Gaining insights into human evaluations is important for determining the quality of individual evaluators as well as identifying true labels of items. Here, we generalize the question of estimating the quality of individual evaluators, extending it to obtain diagnostic insights into how various evaluators label different kinds of items. We propose a series of increasingly powerful hierarchical Bayesian models which infer latent groups of evaluators and items with the goal of obtaining insights into the underlying evaluation process. We apply our framework to a wide range of real-world domains, and demonstrate that our approach can accurately predict evaluator decisions, diagnose types of mistakes evaluators tend to make, and infer true labels of items.

1 Introduction

Several seemingly unrelated tasks such as a crowd-worker on Amazon Mechanical Turk labeling an image, a librarian classifying a newly arrived title, a Yelp user rating a restaurant, or a student providing an answer to a multiple-choice test share an underlying theme. All of these and many more such situations are examples of human evaluation processes, in which an *evaluator* is shown an *item* and attempts to choose a *correct label* for it.

The result of each such evaluation depends on the attributes of both the evaluator and the item. For example, consider a crowd-worker (the evaluator) labeling images of birds (the items). The quality of the labels produced will likely depend on the characteristics of the crowd-worker, including her general level of expertise about birds and/or her experience with different geographical regions; the quality will also depend on the characteristics of the birds being labeled.

A long line of research has studied how to take multiple labels from non-expert evaluators and synthesize them into a single high-quality label [1]-[7], and how to estimate the performance of individual evaluators on various tasks [8]-[12]. However, relatively little attention has been focused on obtaining deeper insights into evaluations such as understanding the characteristics of mistakes being made and identifying the shared attributes of evaluators and items that are relevant to the quality of the resulting label. Discovering these patterns may in turn generate diagnostic insights such as which kinds of items are particularly hard to label or what types of mistakes certain kinds of evaluators are making.

In order to understand human evaluations, Dawid and Skene [10] proposed a model for estimating *confusion matrices* of individual evaluators. A *confusion matrix* models the labeling decisions of an evaluator. In a confusion matrix $\Theta^{(j)}$, entry (p, q) is the probability of an item with *true* label p being assigned label q by an evaluator j. Error-free evaluation corresponds to a diagonal confusion matrix, while off-diagonal entries record different types of errors. However, the problem is that often it is too expensive to obtain enough evaluations and enough ground-truth labels to estimate a separate confusion matrix for each evaluator. Further, it might not be possible to explain all the decisions of an evaluator with just one such confusion matrix. This is due to the fact that decisions also depend upon the characteristics of the items that an evaluator is judging.

Present work: Obtaining diagnostic insights into human evaluations. In this work, we provide a framework for obtaining insights into human evaluations by casting it as a problem of inferring confusion matrices which explain the decisions made by evaluators.

In order to address the aforementioned drawbacks of existing solutions, we propose a novel hierarchical Bayesian framework. The crux of this framework involves inferring two sets of clusters - groups of evaluators and items respectively - which can guide the process of estimating the confusion matrices. The intuition behind the clustering process is to group together all the evaluators who share similar attributes and evaluation styles. Similarly, all the items grouped into the same cluster would share similar attributes and are likely to

*Stanford University, {himalv,jure}@cs.stanford.edu
†Cornell University, kleinber@cs.cornell.edu
‡Harvard University, mullain@fas.harvard.edu

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
On its ten-year anniversary in November 2015, Amazon Mechanical Turk (AMT) continues to thrive as an effective online labor market, but one that raises concerns about worker welfare. A layer of technology separates Amazon's crowd workers (Turkers) from the Requesters for whom they complete work. This separation makes it possible for Requesters to coordinate large crowd workforces, but it also means that each transaction with a worker is mostly anonymous [38, 34, 30], abstract [1, 3], and legally 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 of AMT's hands-off policy. These aspects of rejection lead to feelings of unfairness, to mistrust in Requesters and AMT, and to perceptions of AMT work as risky.

This, in turn, leads workers to adopt strategies to minimize risk: avoiding new and known bad Requesters, sharing information about their experiences with other Turkers, and choosing tasks with clear, concrete descriptions and evaluation criteria. These risk-averse strategies, though rational given the current structure of the market, affect both the kinds of problems AMT can solve and the quality of living and learning Turkers can gain. This in turn harms the long-term prospects for individual workers, Requesters, and the market as a whole to grow and innovate toward the "Future of Crowd Work" envisioned by Kittur et al. [29].

Our contribution is twofold. First, we present an empirical analysis of how AMT's design and policies affect Turkers'

CHI'16, May 07-12, 2016, San Jose, CA, USA
 ISBN 978-1-4503-3362-7/16/05...$15.00
DOI: http://dx.doi.org/10.1145/2858036.2858539

[1] https://www.mturk.com/mturk/conditionsofuse

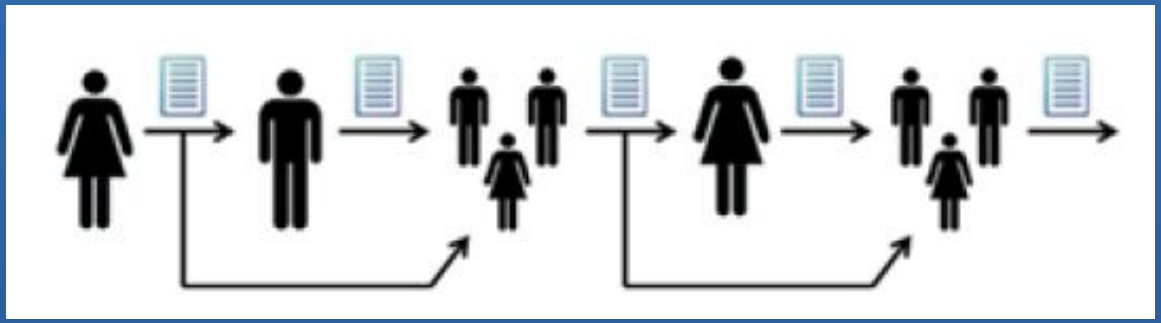
Human Computation

Thinking computationally about organized human labor

- Algorithms
- Abstractions
- Performance measures
  (correctness, accuracy, efficiency, cost, …)
- System building tools
- ….

Killer	whales	are	beautiful	animals	.	I	remember	seeing
these	huge	,	**smooth**	,	black	and	white	creatures	jumping
high	into	the	air	at	Sea	World	,	as	a	kid	.

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.



Annie Kai Ian Cheang View profile ★★★★★ (1 user) More options Apr 10 1995, 3:00 am

Hi you all:

I don't know if this is the proper place to post this, but I got the message below from one of the other newsgroup I'm in, and I just want to post it here to see if anyone could help.

---------- forwarded message begins----------
Date: Mon, 10 Apr 1995 04:44:56 PST
Reply-To: Medical student discussion list <MEDSTU-L%UNMVMA.BIT...@cmsa.Berkeley.EDU>
Sender: Medical student discussion list <MEDSTU-L%UNMVMA.BIT...@cmsa.Berkeley.EDU>
From: Cai Quanqing <ca...@MCCUX0.MECH.PKU.EDU.CN>
X-To: medst...@unmvma.bitnet
To: Multiple recipients of list MEDSTU-L <MEDSTU-L%UNMVMA.BIT...@cmsa.Berkeley.EDU>

Hi,

This is Peking University in China, a place those dreams of freedom and democracy. However, a young, 21-year old student has become very sick and is dying. The illness is very rare. Though they have tried, doctors at the best hospitals in Beijing cannot cure her; may do not even know what illness it is. So now we are asking the world -- can somebody help us?

Here is a description of the illness:

The young woman -- her name is Zhu Ling -- is a student in the chemistry department. On DEC. 5, 1994, Zhu Ling felt sick to her stomach. Three days later, her hair began to fall out and within two days she was completely bald. She entered the hospital, but doctors could not discover the season for her illness. However, after she was in the hospital for a month, she began to fell better and her hair grew back. Zhu Ling went back to school in February, but in March her legs began to ache severely, and she felt dizzy. She entered XieHe Hospital - Chinese most famous hospital. In early March and on March 15, her symptoms worsened. She Began to facial paralysis, central muscle of eye's paralysis, self-controlled respiration disappeared. So she was put on a respirator.

The doctors did many tests for many diseases(include anti-H2V, spinal cord puncture, NMR, immune system, chemical drug intoxication ANA,ENA,DSONA,ZG and Lyme), but all were negative, except for Lyme disease(ZGM(+)).

The doctors now think that it might be acute disseminated encephalomyelitis(ADEM) or lupus erythematosus(LE), but the data from the tests do not support this conclusion.

The doctors are now treating Zhu Ling with broad-spectrum antibiotic of cephalosporin, anti-virus drug, hormone, immun-oadjuvent, gamma globulin intravenous injection and have given her plasma exchange(PE) of 10,000 CCs. But Zhu Ling has not responded -- she reamers in a vegetative state, sustained by life support.

If anyone has heard of patients with similar symptoms -- or have any ideas as to what this illness could be, please contact us. We are Zhu Ling's friends and we are disparate to help her.

This is the first time that Chinese try to find help from Internet, please send back E-mail to us. We will send more crystal description of her illness to you.

Thank you very much
Peking University
April 10th, 1995

==========

Please foreword this message to your freinds if you think they can help us ,Thanks advanced!

email:ca...@mccux0.mech.pku.edu.cn

End of messages

« Back to Discussions

« Newer topic Older topic »

International Electronic Link Solves Medical Puzzle

DIPLOMATS FOSTERING the slow thaw in US-China relations might look to the medical community for inspiration.

Via the worldwide computer Internet and other means of communication, physicians and other medical scientists from coast to coast in the United States and at least 17 other countries have helped their mainland China colleagues treat a university student with a challenging array of signs and symptoms.

The patient, Zhu Lingling, 21, a junior studying physical chemistry in Beijing, reportedly experienced abdominal pain and alopecia in December 1994 but returned to college in February after she responded to Chinese traditional therapy and nutritional support.

A month later, she was hospitalized again with a variety of central nervous system complaints and became comatose within 5 days.

After tests ruled out a number of tentative diagnoses and she did not respond significantly to treatment, students at Beijing University who had learned of the complicated case sent an electronic mail request for diagnostic and therapeutic assistance. This was relayed by several groups on the Internet.

Apparently the first to respond from the United States with the correct diagnosis was Stephen O. Cunnion, MD, PhD, MPH.

A US Navy captain who is an infectious disease epidemiologist in the Department of Preventive Medicine and Biostatistics, Uniformed Services University, Bethesda, Md, Cunnion diagnosed the problem as thallium poisoning.

In the next 4 weeks more than 80 other individuals and groups in medicine—out of some 2000 that eventually responded—supported Cunnion's diagnosis. At least one physician in Beijing had suggested the thallium poisoning diagnosis, but it had not been put to the test there.

However, following the Internet response, the Chinese conducted tests that confirmed the thallium poisoning diagnosis. Physicians and other medical scientists in California, particularly the University of California, Los Angeles (UCLA), coordinated much of the succeeding effort with colleagues elsewhere.

Cunnion says that Chinese physicians now report the young woman has regained consciousness. The prognosis is encouraging, say many of the experts involved, but recovery—perhaps with only limited neurological sequelae—may be a long process.

The source of the poisoning has not been determined. One report indicates that Chinese authorities are looking into the possibility of criminal intent.

In the meantime, physicians and others involved in the evolving field of telemedicine suggest that this experience offers some insight into its future potential.—by Phil Gunby

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.

CrowdLang - Programming Human Computation Systems

Interweaving Human and Machine Intelligence in a Complex Translation Task

Patrick Minder
University of Zurich
Dynamic and Distributed
Information Systems Group
minder@ifi.uzh.ch

Abraham Bernstein
University of Zurich
Dynamic and Distributed
Information Systems Group
bernstein@ifi.uzh.ch

ABSTRACT

Today, human computation systems are mostly used for batch processing large amount of data in a variety of tasks (e.g., image labeling or optical character recognition) and, often, the applications are the result of extensive lengthy trial-and-error refinements.

A plethora of tasks, however, cannot be captured in this paradigm and as we move to more sophisticated problem solving, we will need to rethink the way in which we coordinate networked humans and computers involved in a task. What we lack is an approach to engineer solutions based on past successful patterns.

In this paper we present the programming language and framework CrowdLang for engineering complex computation systems incorporating large numbers of networked humans and machines agents incorporating a library of known successful interaction patterns. CrowdLang allows to design complex problem solving tasks that combine large numbers of human and machine actors whilst incorporating known successful patterns.

We evaluate CrowdLang by programming a text translation task using a variety of different known human-computation patterns. The evaluation shows that CrowdLang is able to simply explore a large design space of possible problem solving programs with the simple variation of the used abstractions.

In an experiment involving 1918 different human actors we, furthermore, show that a mixed human-machine translation significantly outperforms a pure machine translation in terms of adequacy and fluency whilst translating more than 30 pages per hour and that the mixed translation approximates the human-translated gold-standard to 75% using the automatic evaluation metric METEOR. Last but not least, our evaluation illustrates that a new human-computation pattern, which we call staged-contest with pruning, outperforms all other refinements in the translation task.

Author Keywords

CrowdLang, Programming Language, Human Computation, Collective Intelligence, Crowdsourcing, CrowdLang, Translation Software, Pattern Recombination

A short research note summarizing this technical paper with the title "How to Translate a Book Within an Hour - Towards General Purpose Programmable Human Computers with CrowdLang" was published at *ACM WebSci Conference 2012*.

ACM Classification Keywords

H.1.2 User/Machine Systems : H.4.1 Workflow management

General Terms

Algorithms, Human Factors, Languages

1. INTRODUCTION

Much of the prosperity gained by the industrialization of the economy in the 18th century arose from the increased productivity by dividing work into smaller tasks performed by more specialized workers. Wikipedia, Google and other stunning success stories show that with the rapid growth of the World Wide Web and the advancements of communication technology, this concept of Division of Labour can also be applied on knowledge work [24, 23]. These new modes of collaboration— whether they are called collective intelligence, human computation, crowdsourcing or social computing[1] —are now able to routinely solve problems that would have been unthinkably difficult only a few years ago by interweaving the creativity and cognitive capabilities of networked humans and the efficiency and scalability of networked humans in processing large amount of data [5]. The advent of crowdsourcing markets such as Amazons Mechanical Turk (MTurk)[2], Clickworker[3], or CrowdFlower[4] even fosters this development and Bernstein et al. suggest that, as the scale and scope of these human-computer networks increase, we can view them as constituting a kind of a global brain [5].

Even though there exist hundreds of human computation systems that harness the potential of this "global brain", our understanding of how to *"program"* these systems is still poor because human computers are different from traditional computers due to the huge motivational, error and cognitive diversity within and between humans [5]. As a consequence, today, human computation is only used for massive parallel information processing for tasks such as image labeling or tagging. These tasks share in common that they are massively

[1]There is an ongoing debate in the research field about the clear distinction between these concepts [28, 17, 23]. In the context of this paper and in analogy to Law et al. [17], we simply consider *human computation* as computation that is carried out by humans and likewise the term *human computation systems* (*HCS*) describes "paradigms for utilizing human processing power to solve problems that computers cannot yet solve" [29].

[2]http://www.mturk.com

[3]http://www.clickworker.com/

[4]http://www.crowdflower.com/

Human-powered Sorts and Joins

Adam Marcus Eugene Wu David Karger Samuel Madden Robert Miller
MIT CSAIL
{marcua,sirrice,karger,madden,rcm}@csail.mit.edu

ABSTRACT

Crowdsourcing markets like Amazon's Mechanical Turk (MTurk) make it possible to task people with small jobs, such as labeling images or looking up phone numbers, via a programmatic interface. MTurk tasks for processing datasets with humans are currently designed with significant reimplementation of common workflows and ad-hoc selection of parameters such as price to pay per task. We describe how we have integrated crowds into a declarative workflow engine called Qurk to reduce the burden on workflow designers. In this paper, we focus on how to use humans to compare items for sorting and joining data, two of the most common operations in DBMSs. We describe our basic query interface and the user interface of the tasks we post to MTurk. We also propose a number of optimizations, including task batching, replacing pairwise comparisons with numerical ratings, and pre-filtering tables before joining them, which dramatically reduce the overall cost of running sorts and joins on the crowd. In an experiment joining two sets of images, we reduce the overall cost from $67 in a naive implementation to about $3, without substantially affecting accuracy or latency. In an end-to-end experiment, we reduced cost by a factor of 14.5.

1. INTRODUCTION

Crowd-sourced marketplaces such as Amazon's Mechanical Turk make it possible to recruit large numbers of people to complete small tasks that are difficult for computers to do, such as transcribing an audio snippet or finding a person's phone number on the Internet. Employers submit jobs (Human Intelligence Tasks, or HITs in MTurk parlance) as HTML forms requesting some information or input from workers. Workers (called Turkers on MTurk) perform the tasks, input their answers, and receive a small payment (specified by the employer) in return (typically 1–5 cents).

These marketplaces are increasingly widely used. Crowdflower, a startup company that builds tools to help companies use MTurk and other crowdsourcing platforms now claims to more than 1 million tasks per day to more than 1 million workers and has raised $17M+ in venture capital. CastingWords, a transcription service, uses MTurk to automate audio transcription tasks. Novel academic projects include a word processor with crowdsourced editors [1] and a mobile phone application that enables crowd workers to identify items in images taken by blind users [2].

There are several reasons that systems like MTurk are of interest to database researchers. First, MTurk developers often implement tasks that involve familiar database operations such as filtering, sorting, and joining datasets. For example, it is common for MTurk workflows to filter datasets to find images or audio on a specific subject, or rank such data based on workers' subjective opinion. Programmers currently waste considerable effort re-implementing these operations because reusable implementations do not exist. Furthermore, existing database implementations of these operators cannot be reused, because they are not designed to execute and optimize over crowd workers.

A second opportunity for database researchers is in query optimization. Human workers periodically introduce mistakes, require compensation or incentives, and take longer than traditional silicon-based operators. Currently, workflow designers perform ad-hoc parameter tuning when deciding how many assignments of each HIT to post in order to increase answer confidence, how much to pay per task, and how to combine several human-powered operators (e.g., multiple filters) together into one HIT. These parameters are amenable to cost-based optimization, and introduce an exciting new landscape for query optimization and execution research.

To address these opportunities, we have built *Qurk* [11], a declarative query processing system designed to run queries over a crowd of workers, with crowd-based filter, join, and sort operators that optimize for some of the parameters described above. Qurk's executor can choose the best implementation or user interface for different operators depending on the type of question or properties of the data. The executor combines human computation and traditional relational processing (e.g., filtering images by date before presenting them to the crowd). Qurk's declarative interface enables platform independence with respect to the crowd providing work. Finally, Qurk automatically translates queries into HITs and collects the answers in tabular form as they are completed by workers.

Several other groups, including Berkeley [5] and Stanford [13] have also proposed crowd-oriented database systems motivated by the advantages of a declarative approach. These initial proposals, including our own [11], presented basic system architectures and data models, and described some of the challenges of building such a crowd-sourced database. The proposals, however, did not explore the variety of possible implementations of relational operators as tasks on a crowd such as MTurk.

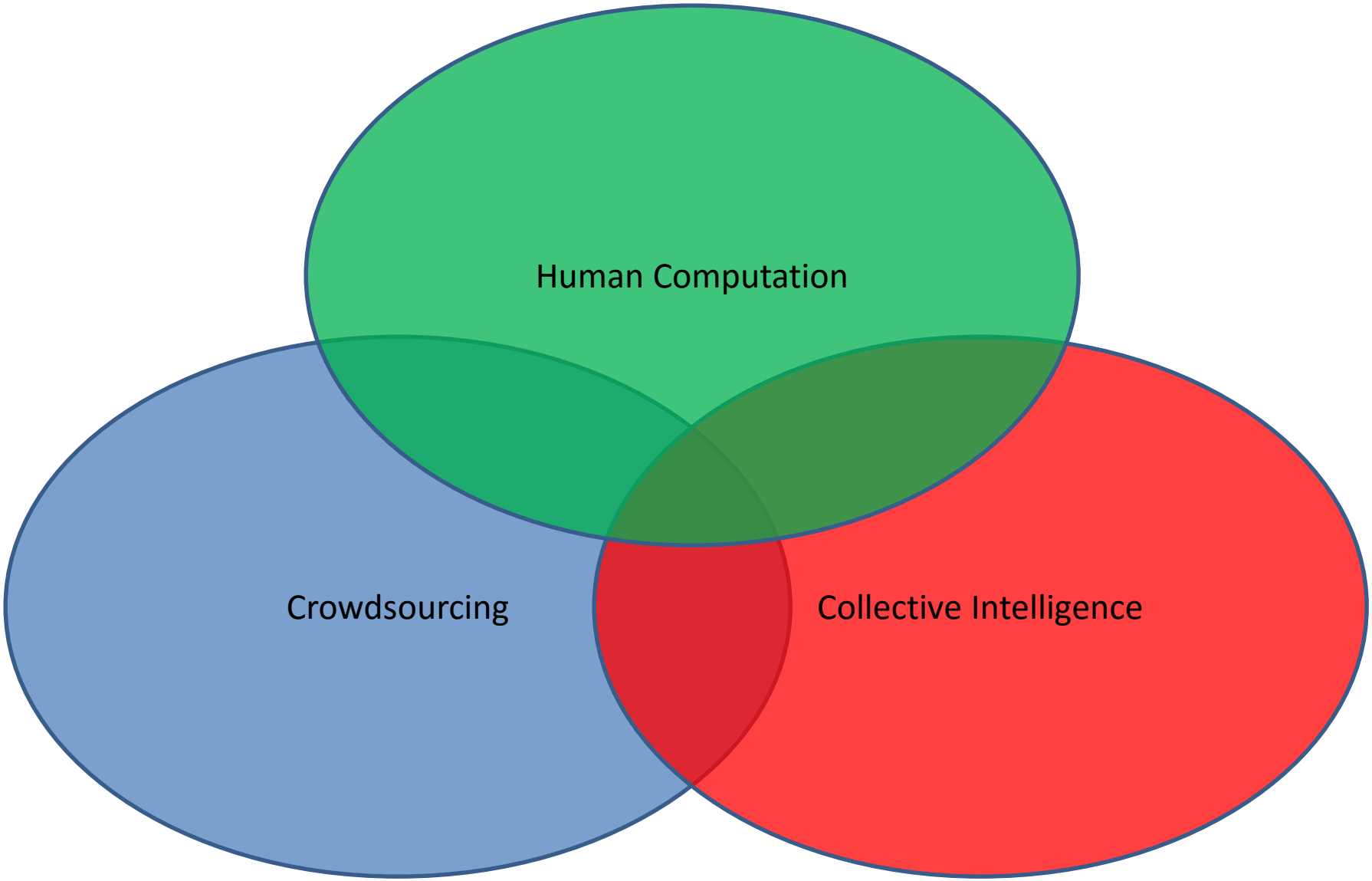
In this paper, we focus on the implementation of two of the most important database operators, joins and sorts, in Qurk. We believe we are the first to systematically study the implementation of these operators in a crowdsourced database. The human-powered versions of these operators are important because they appear everywhere. For example, information integration and deduplication can be stated as a join between two datasets, one with canonical identifiers for entities, and the other with alternate identifiers. Human-powered sorts are widespread as well. Each time a user provides a rating, product review, or votes on a user-generated content website, they are contributing to a human-powered ORDER BY.

Sorts and joins are challenging to implement because there are a variety of ways they can be implemented as HITs. For example,

Proceedings of the VLDB Endowment, Vol. 5, No. 1

What Work in HComp Look Like

- Populating the space of ideas
- Developing understanding
  - Experimental
  - Behavioral science
  - Mathematical
- Establishing engineering principles and tools


Human Computation
Crowdsourcing
Collective Intelligence

Course Style

- Research-centered introduction to the field
  - Learn about the key work and ideas through recent papers
  - Learn by doing
  - Final project output: Conference-like research paper
    - (Human subjects)

Grading

- Projects and Assignments: 66%
  - Smaller ones relevant to week's papers
  - Final Project (Report and Presentation)
- Class presentations and participation: 33%
  - Weekly paper write-ups
  - Leading in-class discussions of papers
- Completing course evaluation: 1%

Assignments for Next Lecture (Tue, Aug 30)

1. Email haym.hirsh@cornell.edu:
   - Name and email
   - Program of study, year
   - Interests and background
2. Read:
   - Alexander J. Quinn and Benjamin B. Bederson. "Human computation: a survey and taxonomy of a growing field." *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 2011.

Assignments for Next Lecture (Tue, Aug 30)

3. Initiate setting up a *worker* account at Amazon Mechanical Turk:
   - Go to mturk.com, click on “Sign in as a Worker” at upper right, enter your email, click on “No, I am a new customer” and proceed from there
4. Complete IRB Training:
   - https://www.oria.cornell.edu/documents/IRB%20Training%20(for%20new%20human%20participant%20researchers).pdf