The Practice of Crowdsourcing: Things to Know About Using Humans and Machines for Labeling

Omar Alonso

NTCIR-13, December 2017
Tokyo, Japan

Disclaimer

The views, opinions, positions, or strategies expressed in this talk are mine and do not necessarily reflect the official policy or position of Microsoft.

Outline

Introduction

Problems

Wetware programming

Quality control

Implementation considerations

Conclusion

Introduction

Human computation

Use humans as processors in a distributed system

Workers, raters, annotators, judges

Address problems that computers aren't good

Human Intelligence Task (HIT)

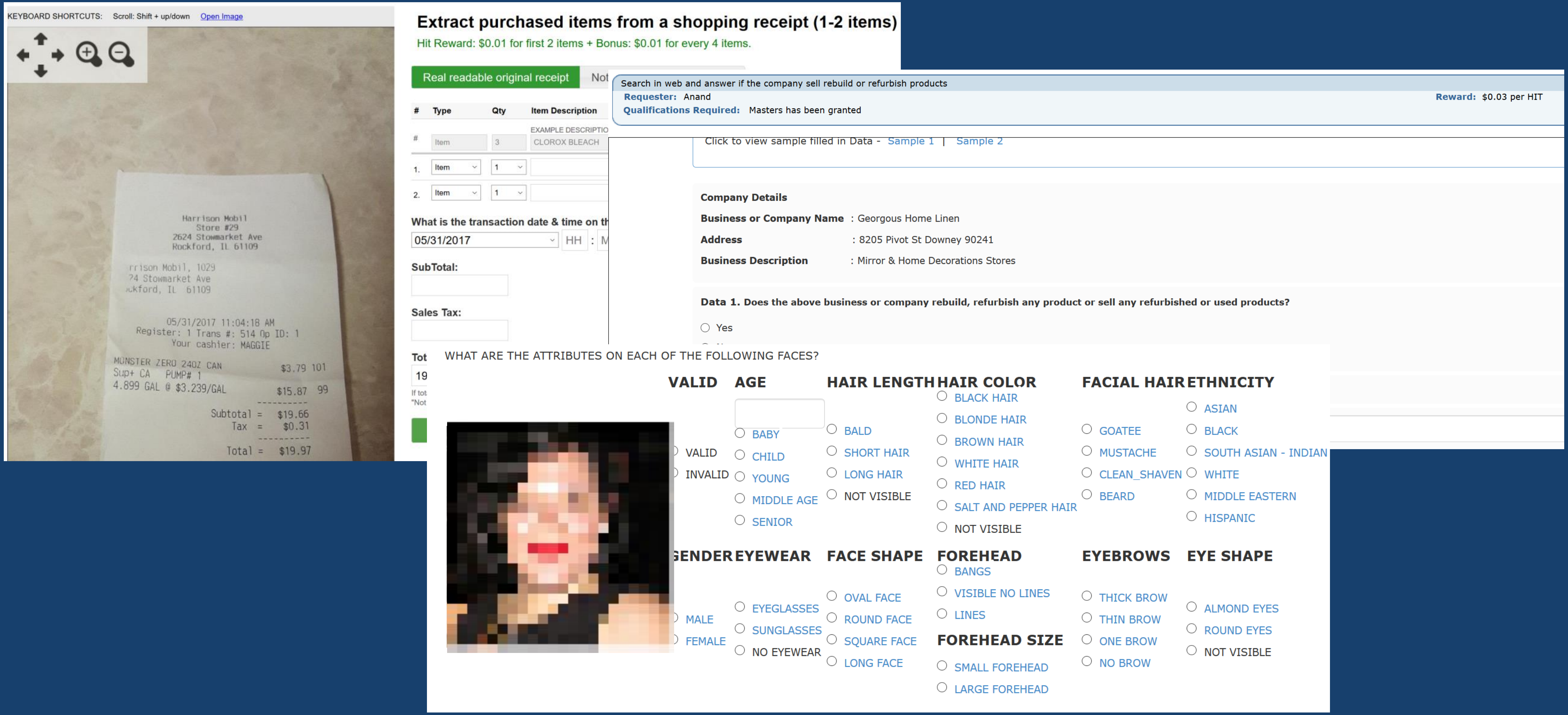
Available platforms

- Amazon Mechanical Turk
- CrowdFlower

L. von Ahn and L. Dabbish. "Designing games with a purpose". CACM, 2008

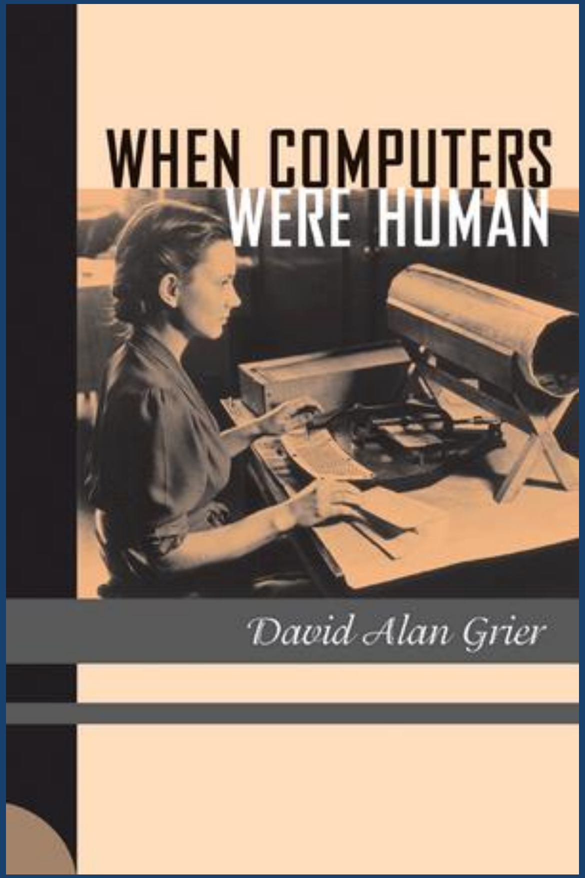
E. Law and L. von Ahn. *Human Computation*. Morgan & Claypool Publishers, 2011

A sample of HITs



In case you didn’t know

You are a computer



Some context

We assume supervised or semi-supervised learning

Large scale

Continuous

Crowdsourcing != Mechanical Turk

Why we need labels?

Information retrieval

Natural language processing

Machine learning

Active learning

Artificial intelligence

A sample of common tasks

Content moderation

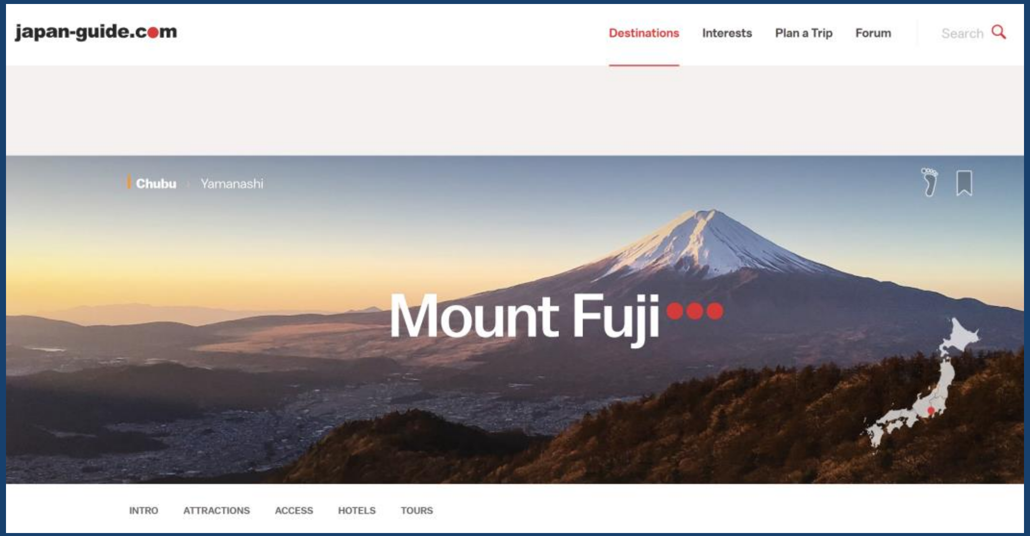
Information extraction

Search relevance

Entity resolution

A. Marcus and A. Parameswaran. *Crowdsourced Data Management.* Foundations and Trends in Databases, 2016

What is a label?



Query = mount fuji

Task: Given the query, is the page relevant?

Answers: very, somewhat, not

Labels: 1, 0.5, 0

Careful with that ~~axe~~ data, Eugene

In the era of big data and machine learning

labels -> features -> predictive model -> optimization

Labeling perceived as boring

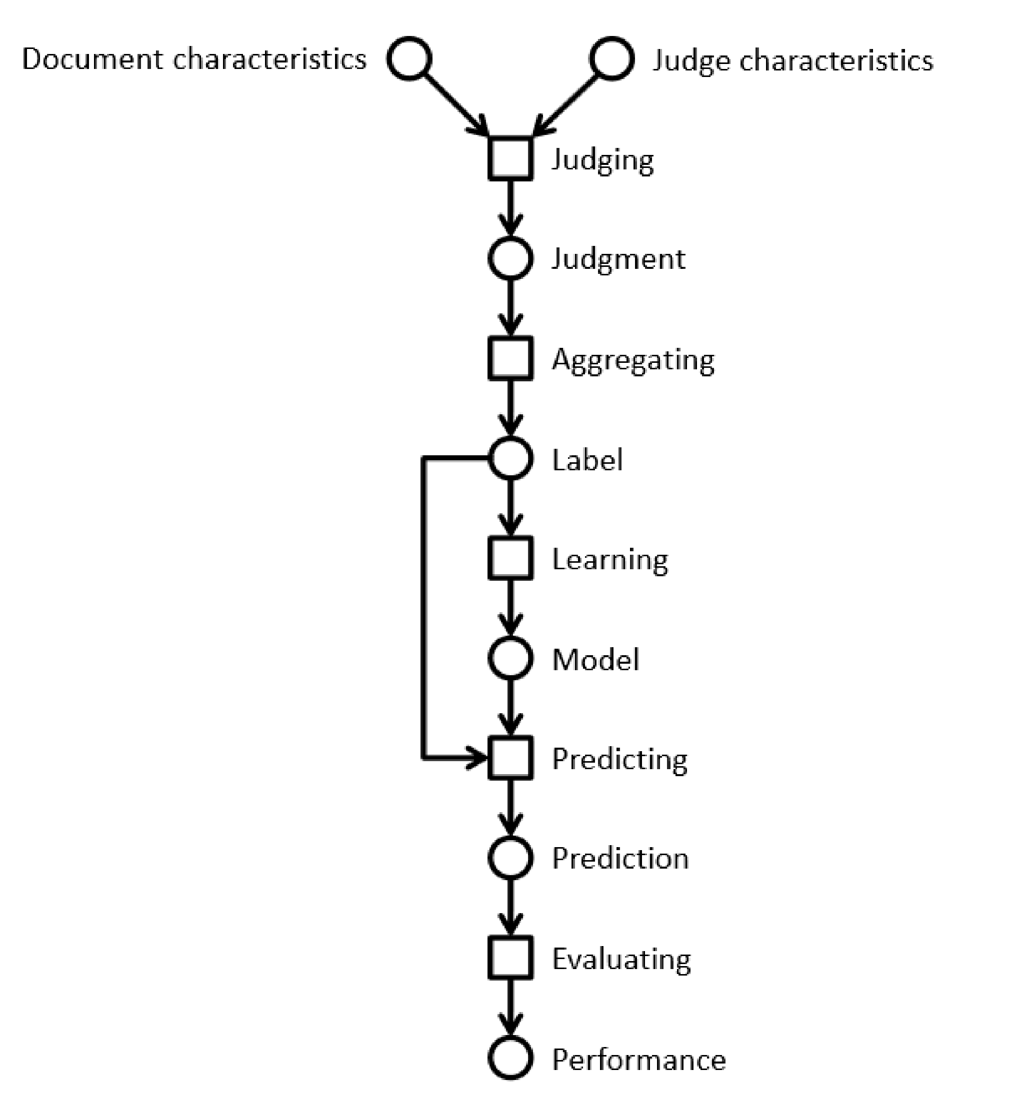
Tendency to rush labeling

Quality is key

Garbage in, garbage out

Lifecycle of a label

Information retrieval example



Using a crowd to label a data set

Using ML to process the complete data set

Three types of labeling tasks

Objective

Objective question has a correct answer

Partially objective

Judgment question has a best answer

Subjective

Subjective question has consistent answer

HC & crowdsourcing in the field

The state of the field

Human-labeled data is more important than ever

Requirements

- Throughput -> ASAP; I need the labels for yesterday
- Cost -> cheap; if possible free
- Quality -> top

Performed as a one-off by 3rd party (crowd or editors)

Non trivial amount of work to get good results

Very limited functionality in current platforms

Problems

Monolithic HITs

The structure of a HIT mirrors the structure of the task the developer is working on

Similar to Conway's law in software engineering

Task complexity

Lengthy instructions

RTFM doesn't work

We don't think of HC/crowdsourcing as programming

How to improve

Use established programming practices

Careful, we are dealing with humans and not machines

Wetware programming

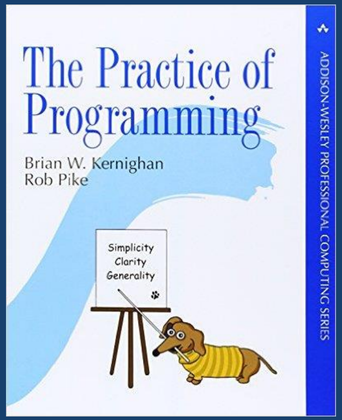
“Machines have no common sense; they do exactly as they are told, no more and no less” - D. Knuth
"Errare humanum est" - Seneca

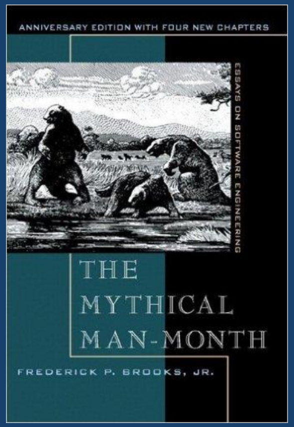
Generic approach

Well-known techniques for writing programs

Humans executing a task on a machine

A programming view for humans and machines





Humans executing code

Instruction set is somewhat unknown

Latency

Cost/incentives

Errors

Task difficulty

Human factors

Asking questions

Part art, part science

Instructions are key

Workers may not be experts so don't assume the same understanding in terms of terminology

Show examples

Hire a writer

- Engineer writes the specification
- Writer communicates

HIT design

Self-contained, short, and simple

Document presentation & design

Engage with the worker

Need to grab attention

Localization

Reliability

What to look for

Agreement, reliability, validity

Inter-rater agreement

Agreement between judges

Agreement between judges and the gold set

Some statistics

Cohen’s kappa, Fleiss’ kappa, Krippendorff’s alpha

kappa or alpha values > 0.8 is unrealistic

Patterns of disagreements

Program structure

Design HITs that humans can do well

Data pipelines and workflows

Taxonomy creation

Cascade

3 HIT primitives and global structure inference

Near-dupes evaluation

1 HIT for identifying a news article (Mechanical Turk) and 1 HIT for near-dupes detection (UHRS)

Different quality strategies and parallelization

L. Chilton, G. Little, D. Edge, D. Weld, J. Landay. "Cascade: Crowdsourcing Taxonomy Creation". CHI 2013

O. Alonso, D. Fetterly, M. Manasse. "Duplicate News Story Detection Revisited". AIRS 2013.

Testing and debugging

The problem

Testing

Attempt to break a program

Debugging

You know the program is broken

How do we test & debug a HIT?

	Machine computation	Human computation
Design	Throw away	Reluctant to throw away
Testing	Systematic	Ad-hoc
Debugging	Programmer's fault	Worker's fault

A background story

Twitter classifier

Detect if a tweet is interesting or not?

Standard ML approach

Get labels

Feature engineering

Modeling with a tool (e.g., Weka, etc.)

Production classifier

Moderate kappa values

What’s going on?

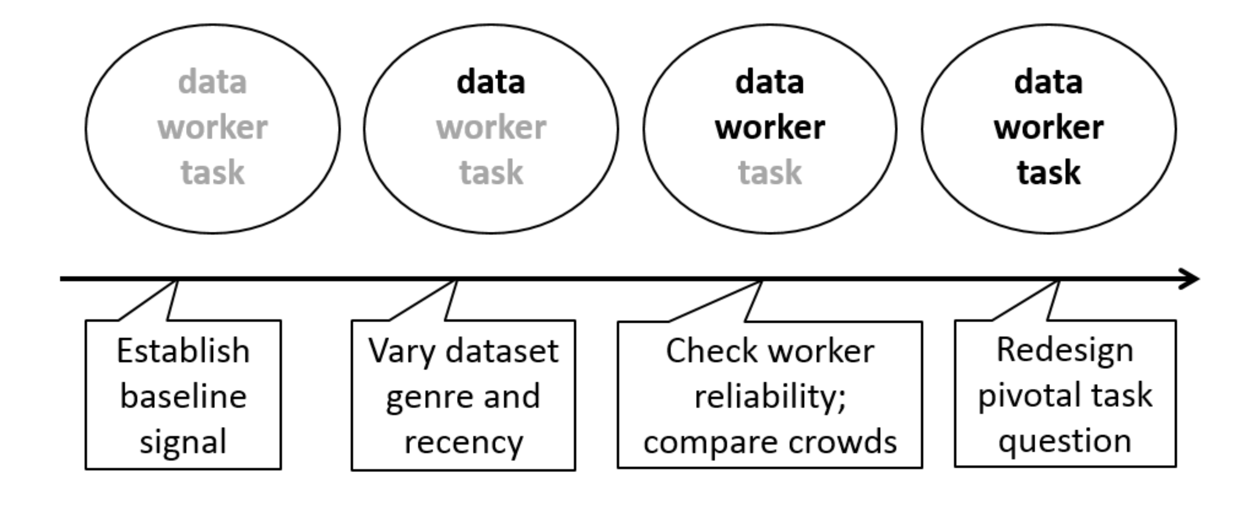
Debugging framework

Human computation tasks are difficult to debug

Multiple contingent factors

Framework

Data-worker-task
Rapid iteration
Small data sets
Emphasis on testing before scaling



O. Alonso, C. Marshall, M. Najork. “Debugging a Crowdsourced Task with Low Inter-rater Agreement”. JCDL 2015

HIT as baseline

Paul Allen offers up $8M for artificial intelligence researchers to uncover 'world-changing breakthroughs': geekwire.com/2014/paul-alle...

Q1. Do you think the tweet is interesting to a broad audience?

- ○ Yes
- ○ No

	B1 (older, random)	B2 (recent, random)
% interesting	16.7%	14.3%
Krippendorff's α	0.013	0.052

Worker reliability and expertise

Borrowed idea from reCAPTCHA: use of control term

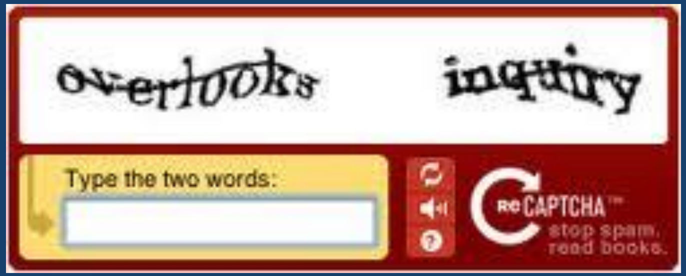
Human Intelligence Data Driven Enquires (HIDDEN)

2 more questions as control

1 algorithmic

1 semantic

Adapt your labeling task



HIT with HIDDENs

Tweet de-branded

Paul Allen offers up $8M for artificial intelligence researchers to uncover 'world-changing breakthroughs': geekwire.com/2014/paul-alle…

HIDDENs

Q1. How many hashtagged words (words that begin with a "#") are in this tweet?

Q1 (alpha = 0.888)

- 0 (no hashtags)
- 1
- 2
- 3 or more

Q2. Does the tweet name a specific person?

Q2 (alpha = 0.708)

- Yes
- No

The main question

Q3. Do you think the tweet is interesting to a broad audience?

Q3 (alpha = 0.160)

- Yes
- No

HIT re-design

Tweet de-branded

Paul Allen offers up $8M for artificial intelligence researchers to uncover 'world-changing breakthroughs': geekwire.com/2014/paul-alle...

HIDDENs

Q1. How many hashtagged words (words that begin with a "#") are in this tweet?

- ○ 0 (no hashtags)
- ○ 1
- ○ 2
- ○ 3 or more

Q1 (alpha = 0.910)

Q2. Does the tweet name a specific person?

- ○ Yes
- ○ No

Q2 (alpha = 0.758)

Breakdown by categories to get better signal

Q3. Please check all the boxes that apply to this tweet

- ☐ Worthless
- ☐ Trivial
- ☐ Funny
- ☐ Makes me curious
- ☐ Contains useful information
- ☐ Important news

Q3 Worthless (alpha = 0.384)
Q3 Trivial (alpha = 0.097)
Q3 Funny (alpha = 0.134)
Q3 Makes me curious (alpha = 0.056)
Q3 Contains useful info (alpha = 0.079)
Q3 Important news (alpha = 0.314)

Algorithms for quality control

Algorithms used in practice

Majority vote

Programmatic gold

EM

Get another label

Vox populi

V. Sheng, F. Provost, P. Ipeirotis. "Get Another Label? Improving Data Quality Using Multiple, Noisy Labelers". KDD 2008.

D. Oleson et al. "Programmatic gold: Targeted and scalable quality assurance in crowdsourcing". In Human Computation Workshop, 2011.

O. Dekel, O. Shamir. "Vox populi: Collecting high-quality labels from a crowd". COLT 2009.

Crowd-workers reviewing work

Soylent

Find-fix-verify

Interactive crowdsourcing

FamilySearch

Arbitration

Peer review

D. Hansen et al. “Quality control mechanisms for crowdsourcing: peer review, arbitration, & expertise at familysearch indexing”, CSCW 2013

M. Bernstein et al. “Soylent: A Word Processor with a Crowd Inside”, UIST 2010

Behavioral features

Focus on the way workers work instead of what they produce

Task fingerprinting

High correlation with work quality

Wernicke

- Information extraction
- Weighted majority voting
- Behavioral features outperform performance-based methods

J. Rzeszotarski and A. Kittur. “Instrumenting the Crowd: Using Implicit Behavioral Measures to Predict Task Performance”. UIST 2011.

S. Han, P. Dai, P. Paritosh, D. Huynh. “Crowdsourcing Human Annotation on Web Page Structure: Infrastructure Design and Behavior-Based Quality Control”. ACM TIST 2016

Practical considerations

What to use?

Depends on complexity and infrastructure access

Voting and honey pots

Cheap and easy to implement

EM-based approaches

Assumes historical performance

Worker verification

More HIT development

Implementation

"Hence, plan to throw one away; you will, anyhow" - F. Brooks

So far …

This is all good but looks like a ton of work

The original goal: good labels

Data quality and experimental designs are *preconditions* to make sure we get the right stuff

Labels will be used for rankers, ML models, evaluations, etc.

Don't cut corners

Development

Coding

Patterns

Modularization

Testing and debugging

Maintenance

Monitoring

Implementation details

Phase	Recommendation
Coding	One language for extracting data from clusters and compute metrics. Avoid moving data from different tools; encoding, data formats, etc.
Design	Use patterns as much as possible. Examples: iterative refinement, find-fix-verify, do-verify, partition-map-reduce, price-divide-solve. Get ready to throw away HITs and results.
Modularization	Design HITs that humans can do well. Think in terms of pipelines and workflows
Testing and debugging	Don't patch a bad HIT: rewrite it. Identify problems with data, workers, and task design.
Maintenance	Version all templates and metadata including payment structure.
Monitoring	Dashboard and alerts.
Documentation	Document the essence of the HIT and its mechanics/integration points.

Machines and humans in sync

Delicate balance but lots of potential

When to use a machine or human for computation

Labels for the machine != labels for humans

Best algorithms for the machine may not be the best choices for humans

Takeaways

Repeatable label quality at scale works but requires a solid programming principles

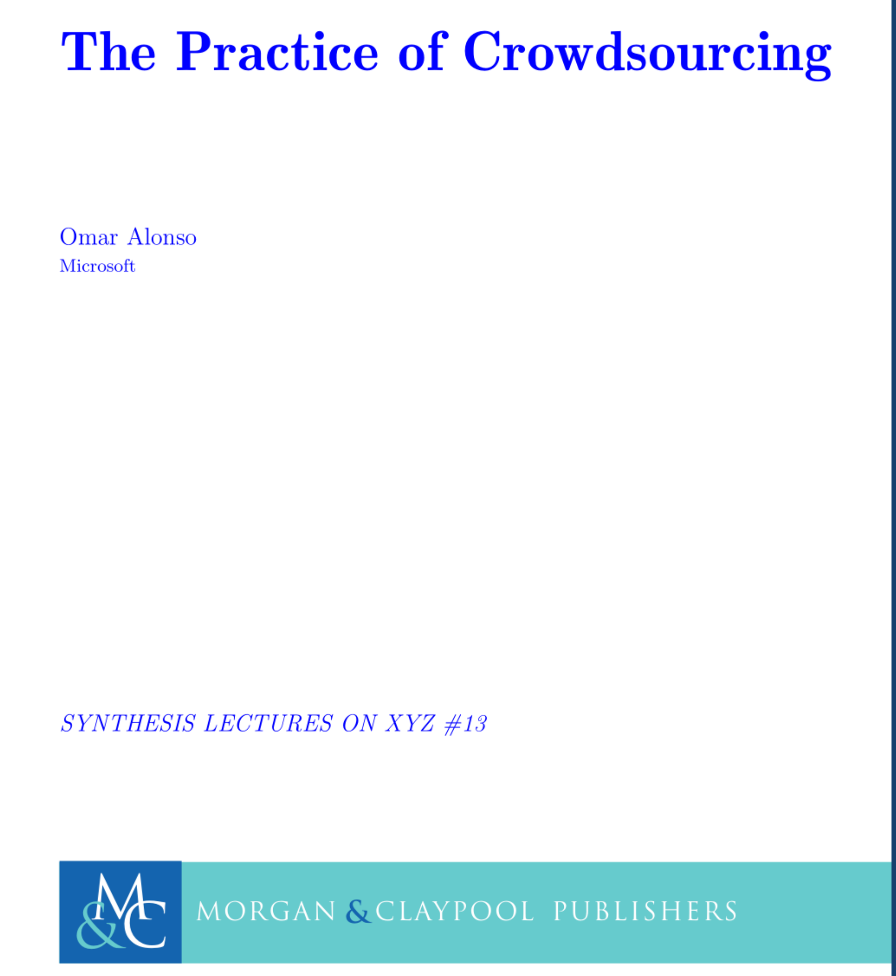
Three aspects that need attention: workers, work and task

Lots of different skills and expertise required

Programing machines is hard, programming applications that involves computations by machines + humans is *harder*

Thank you!

Book under development



The Practice of Crowdsourcing

Omar Alonso
Microsoft

SYNTHESIS LECTURES ON XYZ #13

MORGAN & CLAYPOOL PUBLISHERS

Email: omalonso@microsoft.com
Twitter: @elunca