

Crowd Computing

Rob Miller

User Interface Design Group

MIT CSAIL

Joint work with Greg Little, Lydia Chilton, Max Goldman,
Jeff Bigham, Michael Bernstein,
David Karger, Mark Ackerman, Björn Hartmann, Joel Brandt,
Mason Tang, Elena Tatarchenko

Can You Read This?

For ~~university~~ several ~~months~~. Please ~~inform~~ ~~me~~ ~~with~~ ~~not~~ ~~me~~
at the ~~which~~ a ~~for~~ ~~garment~~ ~~make~~ Overall your ~~writing~~ ~~style~~
is a ~~bit~~ ~~too~~ ~~slow~~. You ~~do~~ ~~not~~ ~~have~~ ~~good~~ ~~ideas~~,
but ~~they~~ ~~get~~ ~~lost~~ ~~in~~ ~~the~~ ~~noise~~
L-

Crowd Computing

Coordinating a **crowd** (a large group of people on the web)
to do **micro-work** (small contributions)
that solves **problems** (that software or one user alone can't do)

volunteer



fun



social



paid



TurKit

human computation
algorithms

VizWiz

helping the blind see
with a crowd's eyes

Caesar

code-reviewing by a
crowd of students

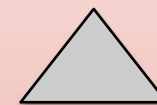
Soylent

a word processor
with a crowd inside

Adrenaline

crowds in
two seconds

User



Crowd

Software

Amazon Mechanical Turk

- highly-available, short, cheap, programmable...
a **prototyping platform** for crowd computing

amazonmechanicalturk Artificial Intelligence **90,071 HITs** available now Rob Miller |

Your Account HITs Qualifications

All HITs | **HITs Available To You** | HITs Assigned To You

Search for containing that pay at least \$ for which you are qualified

All HITs

1-10 of 1775 Results

Sort by: [Show all details](#) | [Hide all details](#) 1 [2](#) [3](#) [4](#) [5](#) > [Next](#) >> [Last](#)

Find the link and image that a forum thread is discussing

[View a HIT in this group](#)

Requester: [AEI](#) **HIT Expiration Date:** Jul 12, 2010 (59 minutes 35 seconds) **Reward:** \$0.05
Time Allotted: 8 minutes **HITs Available:** 1

Write a short answer to a question about writing

[View a HIT in this group](#)

Requester: [conjecture corporation](#) **HIT Expiration Date:** Jul 26, 2010 (1 week 6 days) **Reward:** \$0.04
Time Allotted: 60 minutes **HITs Available:** 20

Why We Need Crowd Algorithms

Please flip an actual coin, and type H if it shows heads, or T if it shows tails.

Requester: Rob Miller

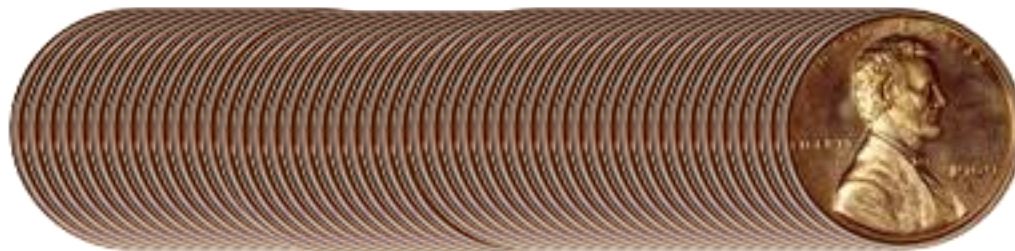
Reward: \$0.01 per HIT

HITs Available: 1

Duration: 60 minutes

Qualifications Required: None

Please flip an actual coin, and type H if it shows heads, or T if it shows tails.



70 heads

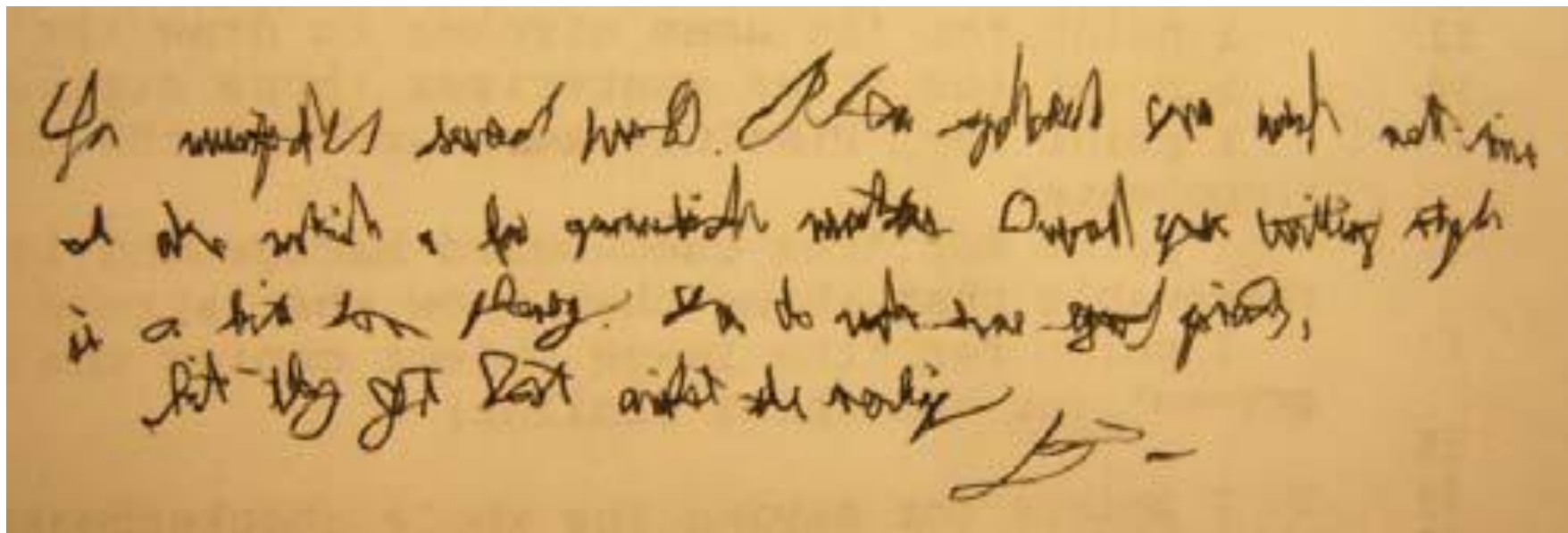


29 tails



1 "1"

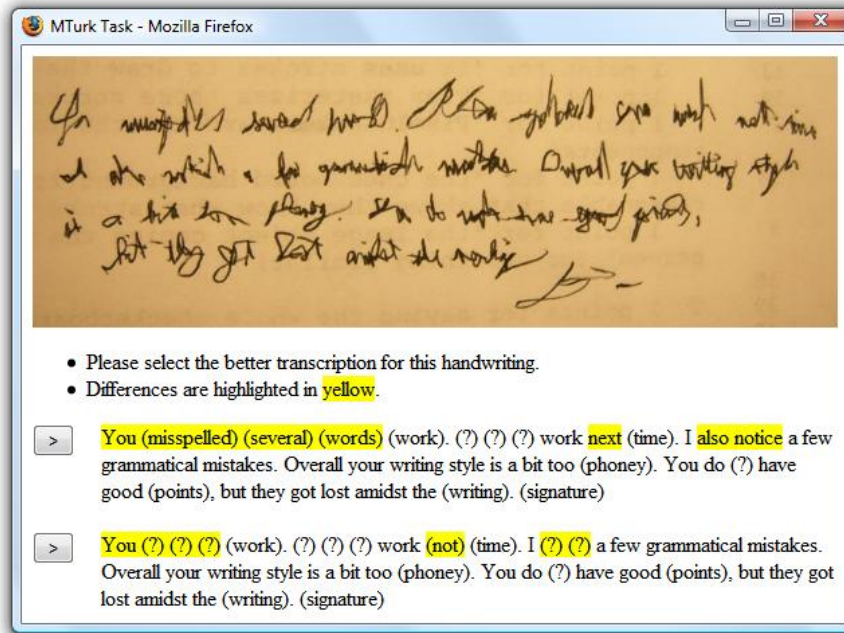
Tackling Hard Tasks with a Crowd



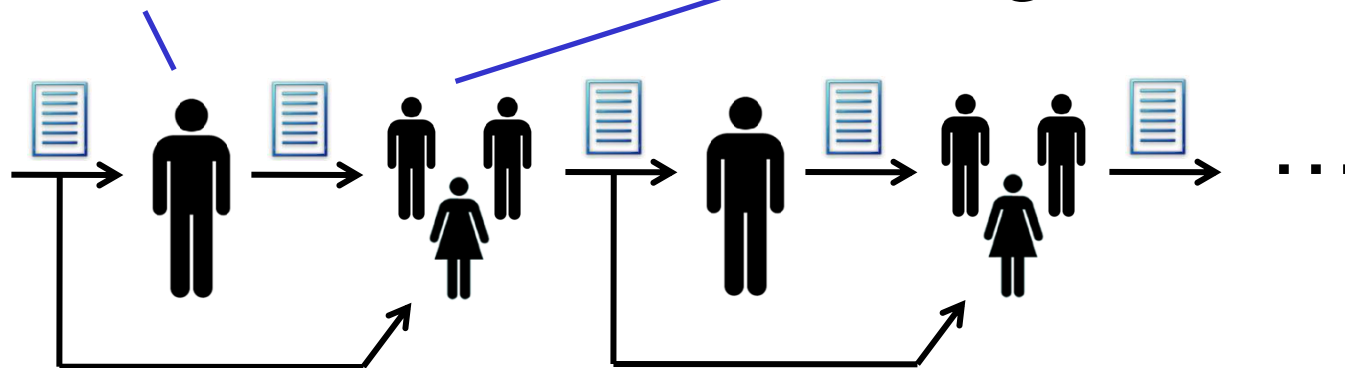
- **Problem:** one person can't do a good transcription
- **Key idea:** iterative improvement by many workers



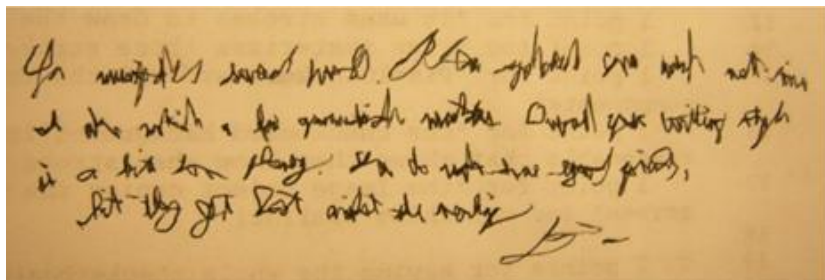
Greg Little et al. "Exploring iterative and parallel human computation processes." HCOMP 2010.



3 votes @ \$0.01



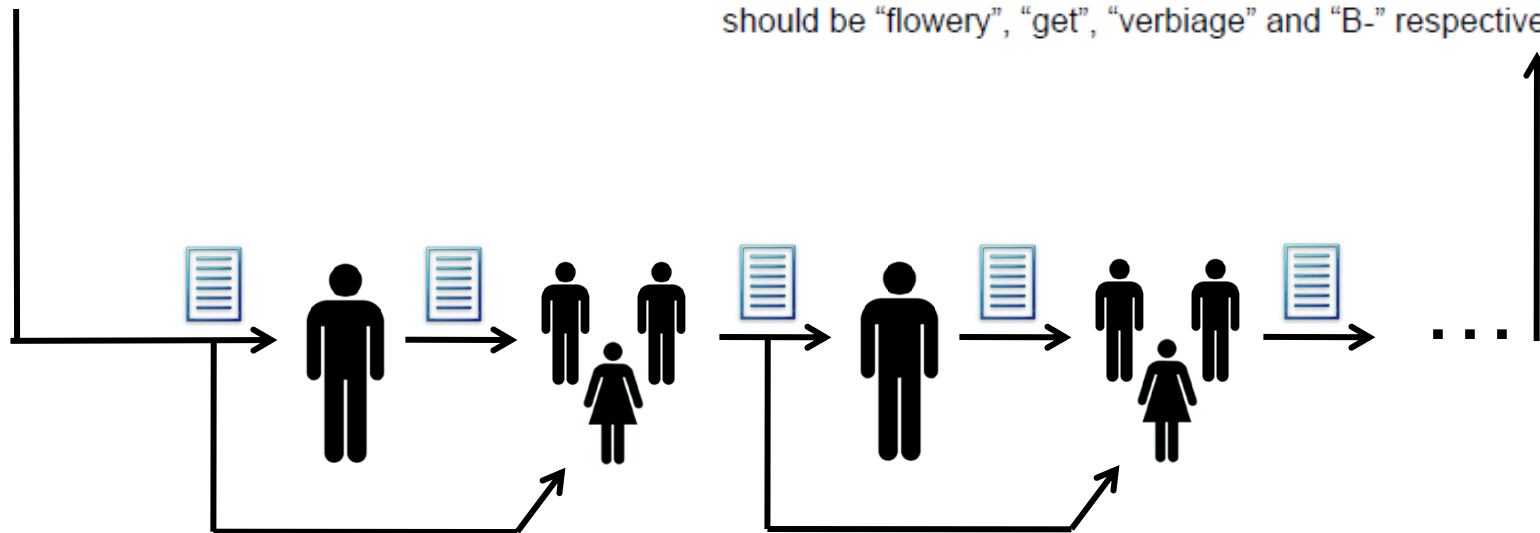
Improve-and-Vote Algorithm



After 9 iterations

"You (misspelled) (several) (words). Please spellcheck your work next time. I also notice a few grammatical mistakes. Overall your writing style is a bit too **flowery**. You do make some good (points), but they **got** lost amidst the **(writing)**. **(signature)**"

According to our ground truth, the highlighted words should be "flowery", "get", "verbiage" and "B-" respectively.



Another Example: Blurry Text

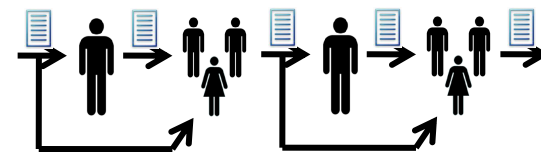
I had intended to hit the nail, but I'm not a very good aim it seems, and I ended up hitting my thumb. This is a common occurrence, I know, but it doesn't make me feel any less ridiculous having done it myself. My new strategy will involve lightly tapping the nail while holding it until it is embedded into the wood enough that the wood itself is holding it straight, and then I'll remove my hand and pound carefully away. We'll see how this goes.

After 8 iterations

I had intended to hit the nail, but I'm not a very good aim it seems and I ended up hitting my thumb. This is a common occurrence I know, but it doesn't make me feel any less ridiculous having done it myself. My new strategy will involve lightly tapping the nail while holding it until it is **embedded** into the wood enough that the wood itself is holding it straight and then I'll remove my hand and pound carefully away. We'll see how this goes.

Human Computation Algorithms

- Improve-and-Vote is a simple but effective crowd algorithm
- Considerations for crowd algorithm design:



Quality

- Improve-and-Vote is useful on a noisy crowd like MTurk (~30% of open-ended work is bad somehow, and ~3% of workers are spammers)
- Other crowds may have less noise

Time

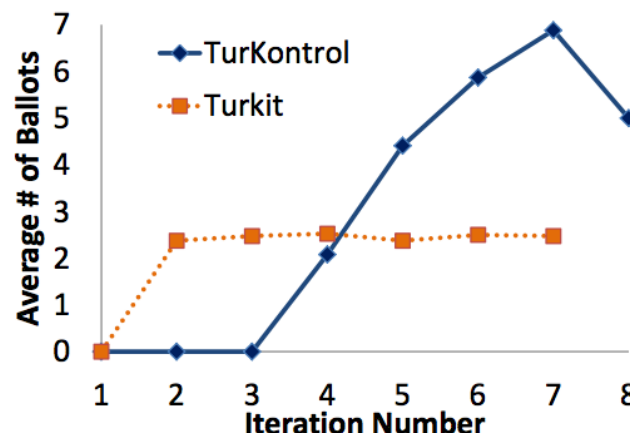
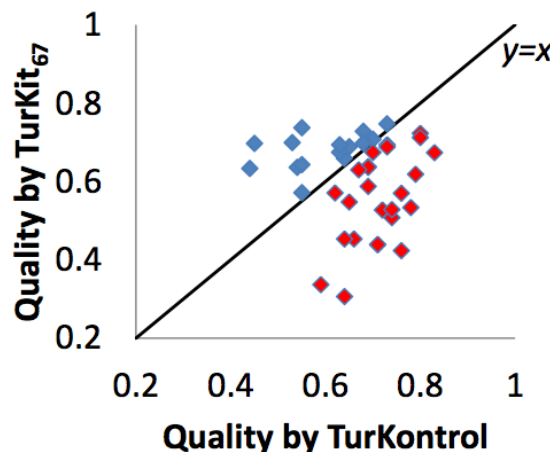
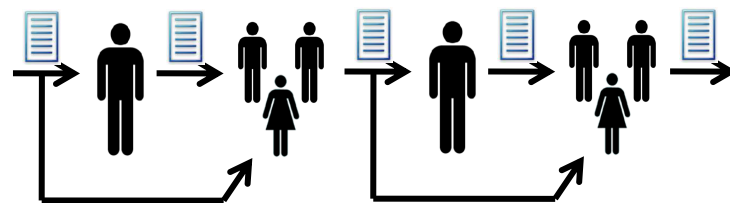
- Improve-and-vote runs slowly, because it's serialized

Incentives

- For MTurk, equivalent to cost in \$
- Other crowds need other incentives

TurKontrol: Optimizing Improve-and-Vote

- Maintain estimates of process state
 - artifact quality
 - worker ability
 - voter accuracy
- Make utility-maximizing decisions
 - when to get another voter
 - when to stop iterating



Peng Dai, Mausam, Daniel S. Weld. “Artificial Intelligence for Artificial Intelligence.” AAAI 2011.

21. Quinn, A., and Bederson, E. A Taxonomy of Distributed Human Computation.
22. Ross, J., Silberman, S., Zaldivar, J., et al. Who Are the Crowdworkers? Understanding Demographics in Amazon's Mechanical Turk. *CHI '10*, ACM Press (2010).
23. Smith, J., and Hancock, J. Degree of Automation to Support Improved Human-Computer Interaction. *IEEE Transactions on Systems, Man, and Cybernetics* 37, 1 (2007), 10-20.

24. Simon, J., Morris, D., and Pasu, S. MySeme: automatic
acoustic segment generation for vocal modeling. *Proc. CHI*
'08, New York: ACM Press (2008).

25. Snow, D., O'Connor, B., Wilensky, K., and Sprockel, A. Y. Leap
and fast: building good evaluation support around
transcription. In *Proceedings of the 2008 Conference on*

26. Sorokin, A. and Forsyth, D. In-fidelity data annotation with Amazon Mechanical Turk. *CVPR*, (2008).
27. von Ahn, S. and Dabbish, L. Labeling images with a computer game. *CHI '08*, ACM Press, (2008).

Shortn

This paper presents Soylent, a word processing interface that uses crowd workers to help with proofreading, document shortening, editing and commenting tasks. Soylent is an example of a new kind of interactive user interface in which the end user has direct access to a crowd of workers for assistance with tasks that require human attention and common sense. Implementing these kinds of interfaces requires new programming patterns for interface software, since crowds behave differently than computer systems. We have introduced one important pattern, Find-Fix-Verify, which splits complex editing tasks into a series of identification, generation, and verification stages that use independent agreement and voting to produce reliable results. We evaluated Soylent with a range of editing tasks, finding and correcting 82% of grammar errors when combined with automatic checking, shortening text to approximately 85% of original length per iteration, and executing a variety of human macros successfully.

Future work falls in three categories. First are new crowd-driven features for word processing, such as readability analysis, smart find-and-replace (so that renaming "Michael" to "Michelle" also changes "he" to "she"), and figure or citation number checking. Second are new techniques for optimizing crowd-programmed algorithms to reduce wait time and cost. Finally, we believe that our research points the way toward integrating on-demand crowd work into other authoring interfaces, particularly in creative domains like image editing and programming.

24. Scott, R., O'Connor, B., Isard, D., and Ng, A.Y. Cheap and fast—but is it good? evaluating non-expert annotations for natural language tasks. *ACL '08*, (2008).

25. Sorokin, A. and Forgy, D. Utility data association with Amazon Mechanical Turk. *CVPR '08*, (2008).

26. van der Aalst, L. and Dabbish, L. Labeling images with a computer game. *CHI '04*, ACM Press (2004).

4. Clarke, J. and Lapata, M. Models for sentence compression: a comparison across domains, training requirements and evaluation measures. *ACL '06*, Association for Computational Linguistics (2006).

5. Cohen, T. and Lapata, M. Sentence compression beyond word deletion. *COLING '08*, (2008).

6. Cypher, A. *What's What?* MIT Press, Cambridge, MA, 1993.

Amazon Mechanical Turk. www.mturk.com, 2006. ACM Press.

22. Sala, M., Partridge, K., Jacobson, L., and Beggs, J. An Exploration into Activity-Informed Physical Advertising Using PEST. *Periscope '07*, Springer Berlin Heidelberg (2007).

23. Simon, I., Morris, D., and Bess, S. Nylong: automatic accompaniment generation for vocal melodies. *Proc. CHI '08*, ACM Press (2008).

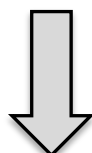
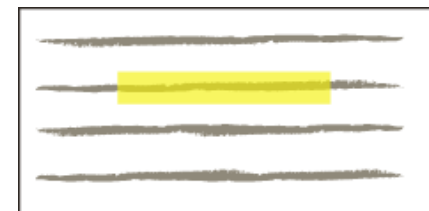
Michael Bernstein et al. "Soylent: a Word Processor with a Crowd Inside." UIST 2010. Best student paper.



Find-Fix-Verify Pattern

Find

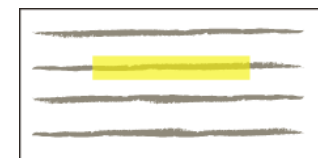
“Identify at least one area that can be shortened without changing the meaning of the paragraph.”



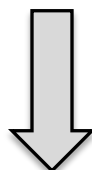
Keep patches found by at least two people

Fix

“Edit the highlighted section to shorten its length without changing the meaning of the paragraph.”



Soylent, a prototype...



Randomize order of suggestions

Verify

“Choose at least one rewrite that has significant style errors in it.
Choose at least one rewrite that significantly changes the meaning of the sentence.”

- ☐ Soylent ~~is,~~ a prototype...
- ☐ Soylent ~~is a~~ prototype...
- ☒ Soylent is a ~~prototype~~ test...

Shortn Performance

- Length
15% shorter on average (10-22% overall)
- Cost
\$1.41 per paragraph
\$0.55 to Find an average of two patches
\$0.48 to Fix each patch
\$0.38 to Verify the results
- Time
Wait: median 18.5 min ($Q_1 = 8.3$ min, $Q_3 = 41.6$ min)
Work: median 2.0 min ($Q_1 = 60$ sec, $Q_3 = 3.6$ min)

Eating Our Own Dog Food

Find-Fix-Verify in Soylent

Both Shortn and Crowproof use the Find-Fix-Verify pattern. We will use Shortn as an illustrative example. To provide the user with near-continuous control of paragraph length, Shortn should produce many alternative rewrites without changing the meaning of the original text or introduce⁷ grammatical errors.

⁷ Word's grammar checker, eight authors and six reviewers did not catch the error in this sentence. Crowproof later did, and correctly suggested that “introduce” should be “introducing”.

VizWiz: Helping the Blind See

What's in this can?

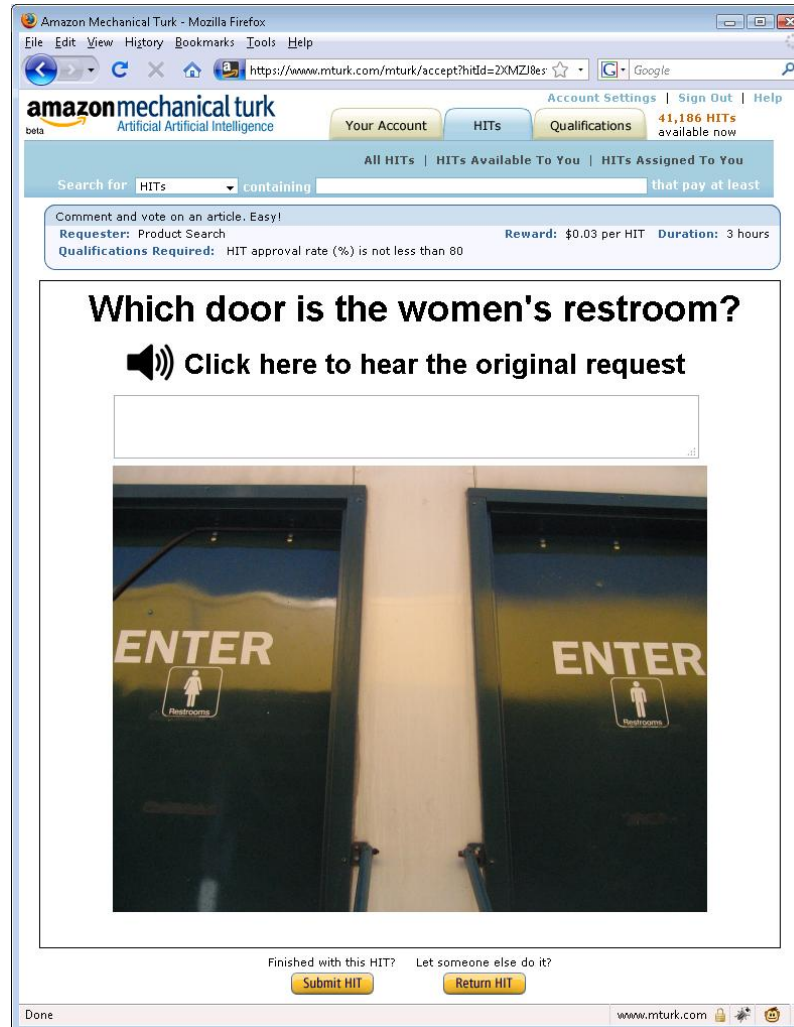


Which door is the women's restroom?

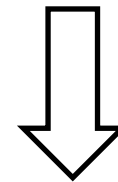


Jeffrey Bigham et al. "VizWiz: Nearly Real-time Answers to Visual Questions." UIST 2010. Best paper award.

Helping the Blind See



Mechanical
Turk



1. the left one
2. LEFT
3. on the left

Deployment

- 10,000+ downloads, dozens of questions/day

Do you see picnic tables across the parking lot?



(13s) no
(46s) no

What temperature is my oven set to?



(69s) it looks like 425 degrees but the image is difficult to see.
(84s) 400
(122s) 450

Can you please tell me what this can is?



(183s) chickpeas.
(514s) beans
(552s) Goya Beans

Adrenaline: A Crowd-Powered Camera



Mechanical
Turk



Ten second video

Time to final picture:

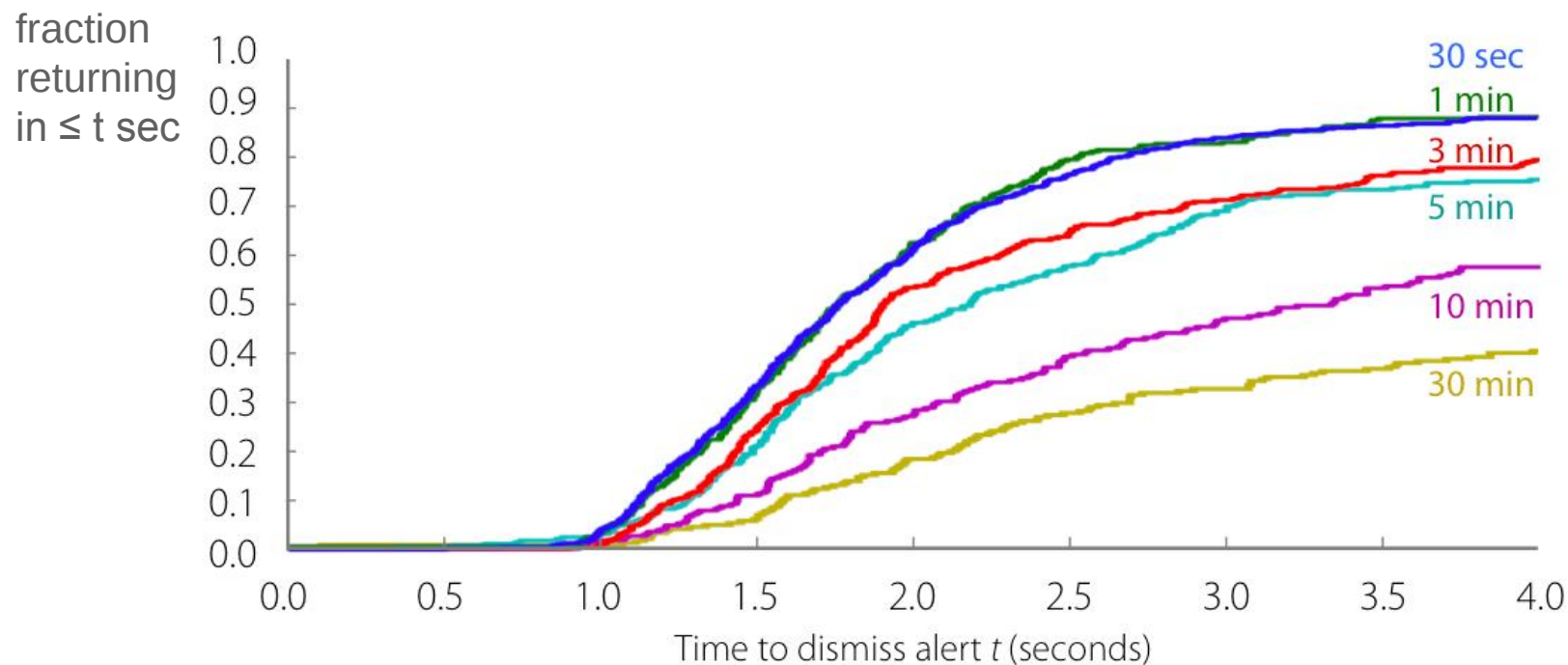


Michael Bernstein, Joel Brandt, et al. “Crowds in Two Seconds: Enabling Realtime Crowd-Powered Interfaces.” **UIST 2011.**



Step 1: Get the Crowd Fast

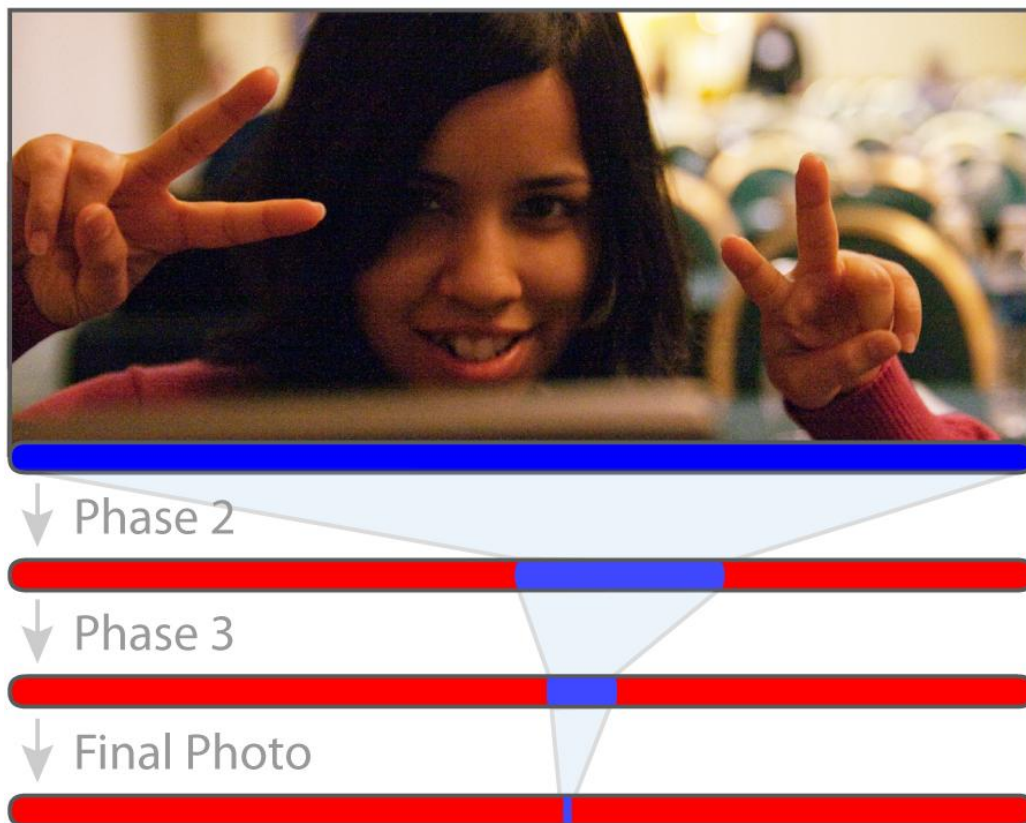
- Retainer model
 - Recruit crowd in advance, and pay them to wait a few minutes



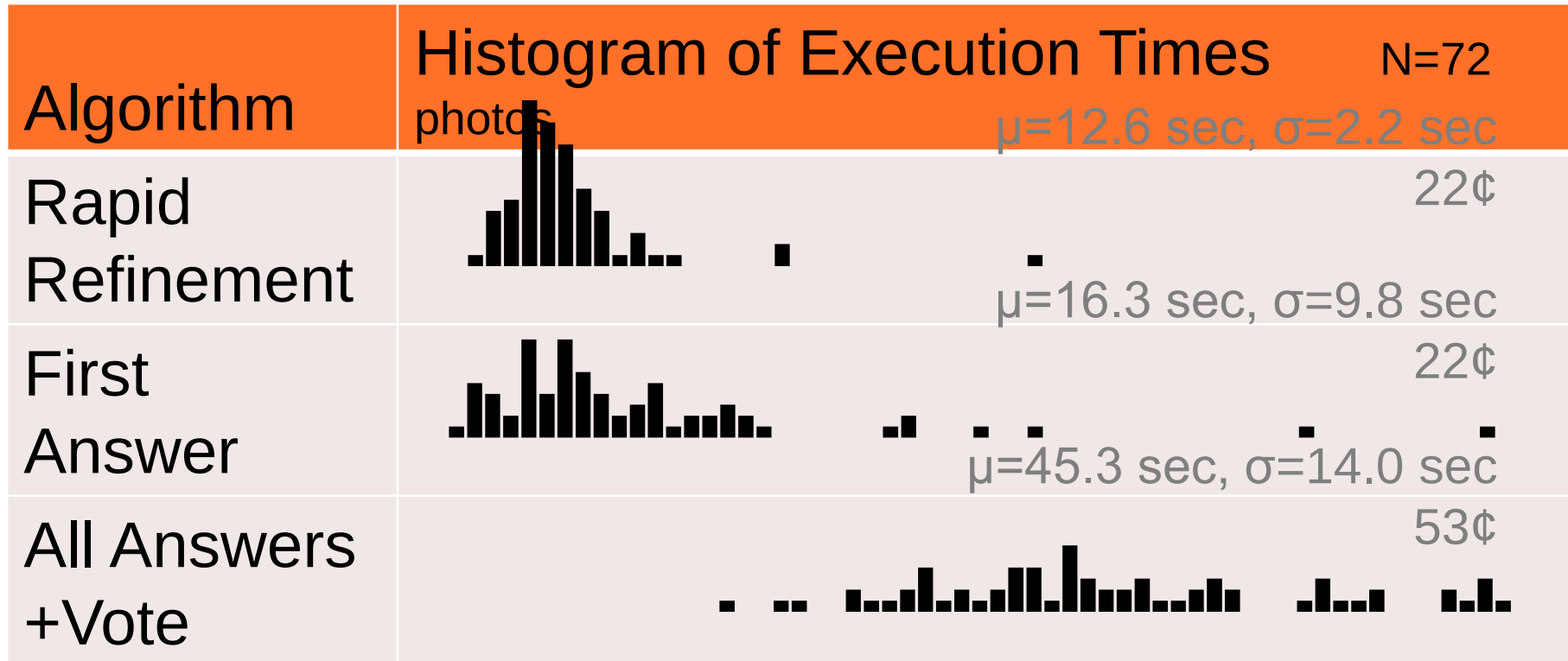
- A worker on retainer costs \$0.30 per hour

Step 2: Get the Crowd to Work Fast

- Rapid refinement pattern
 - Recognize potential agreement early, and use it to reduce the search space



Results



Caesar: Crowd-Driven Code Review

- Students in MIT software engineering classes write lots of code
 - roughly 10kloc in problem sets and projects
- Automatic grading is necessary but not sufficient

```
// compute n! requires n >= 0
int factorial(int n) {
    if (n == 0) return 1;
    else return n * factorial(n-1);
}
```

correct and understandable

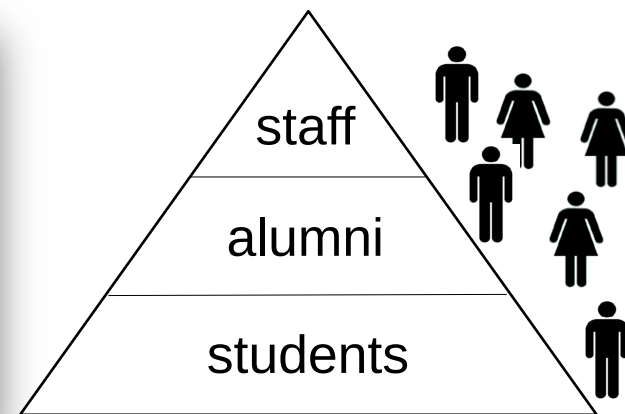
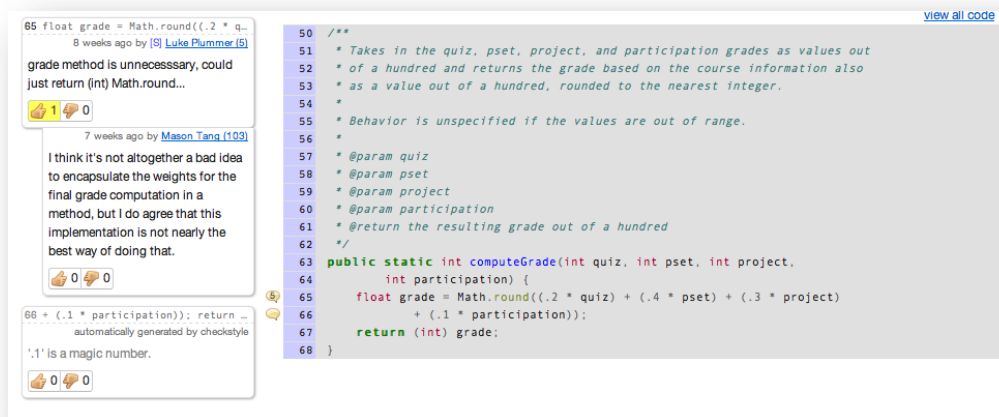
```
int factorial(int n) {
    int i, result=1;
    if (n == 0) result = 1;
    else {
        for (i = 1; i < n; ++i) result *= i;
        result = result*n;
        return result;
    }
    return 1;
}
```

correct but confusing

- we need human readers, and we want line-by-line feedback

Caesar: Crowd-Driven Code Review

- Chop up student programs into **chunks**
- Assign the chunks for review by a **mixed crowd**



- Results from 2 semesters in MIT 6.005 software engineering

13 problem sets, 2200 submissions

21,500 comments

5% alums

16.2% upvoted

8% staff

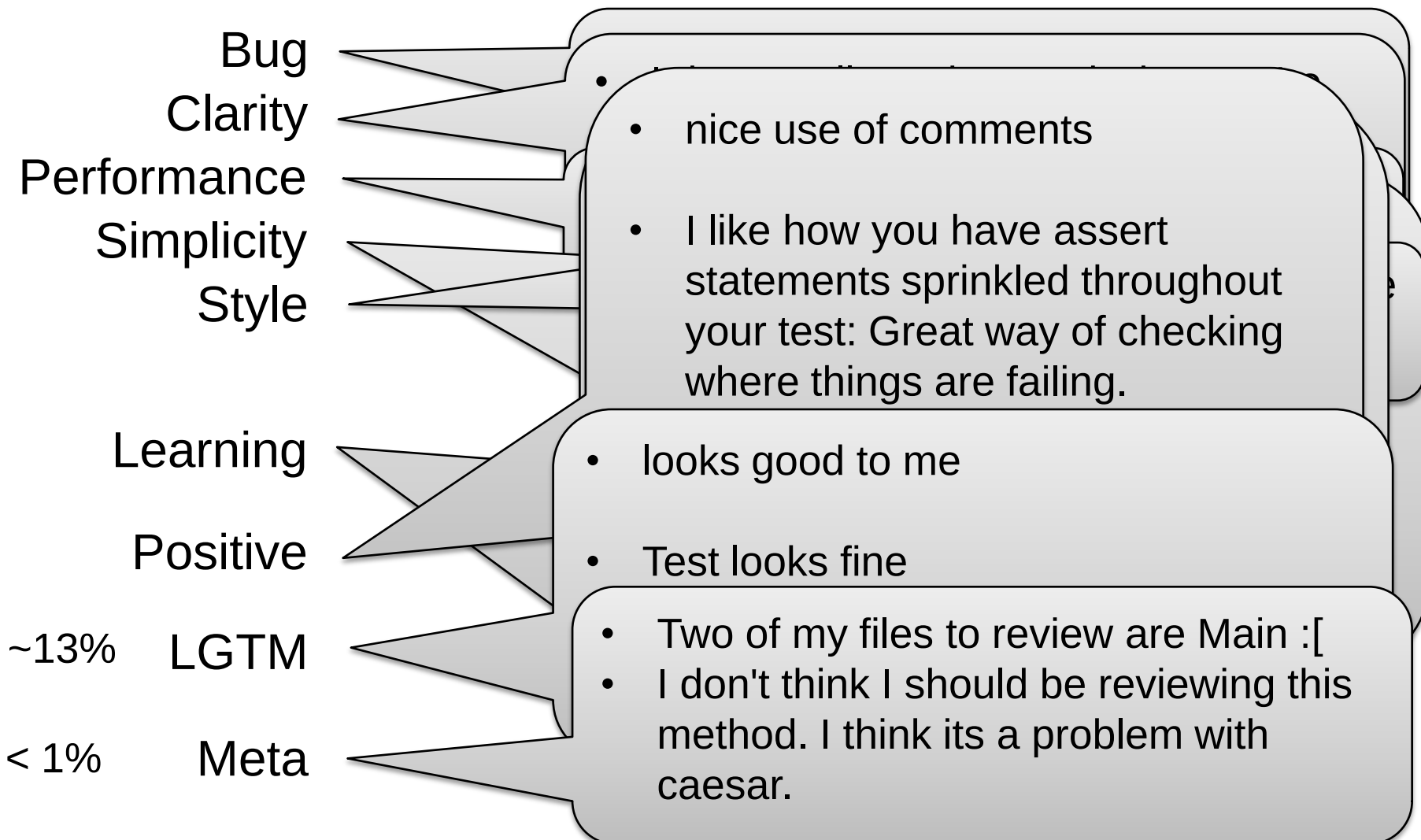
0.7% downvoted

87% students

9.6 comments per submission

comments come back in < 3 days

Kinds of Comments



Alums Have Different Context

- **alum:** avoid abbreviating variable names... 'hi' would be especially confusing to someone who isn't familiar with english
- **student:** Idk about this one though. I've seen lo and hi at various places. I'd say this is fine. Though, the integer N should be lower case, just to conform with the java naming convention.
- **alum:** **When you're in industry and working on code with people who aren't necessarily familiar with English...** it's a problem to use phonetic abbreviations. There is really no good reason to abbreviate variable names especially since IDEs autocomplete.
- **student:** For that, yes, I agree. Sadly, some IDEs don't autocomplete very well. *cough*
- **code author:** These were the variables that were **given to us**

What's Next

TurKit

human computation
algorithms

VizWiz

helping the blind see
with a crowd's eyes

Caesar

code-reviewing by a
crowd of students

Soylent

a word processor
with a crowd inside

Adrenaline

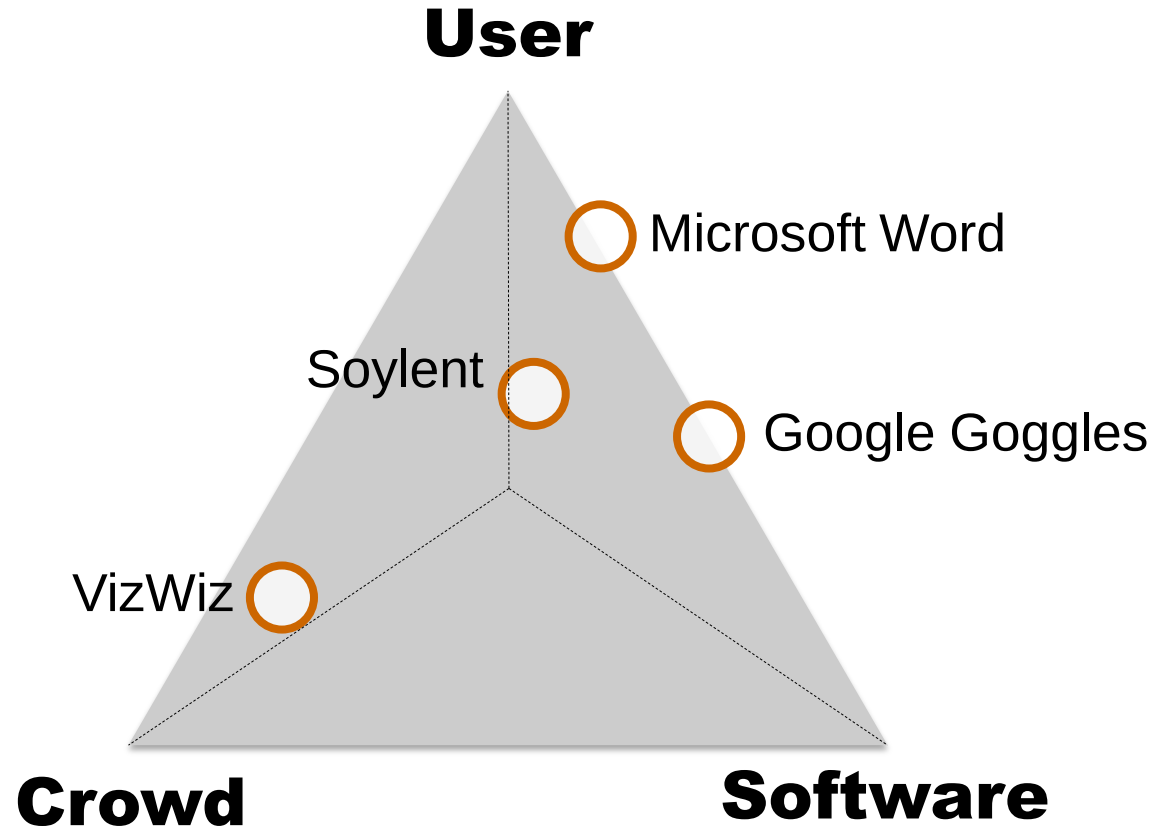
crowds in
two seconds

User

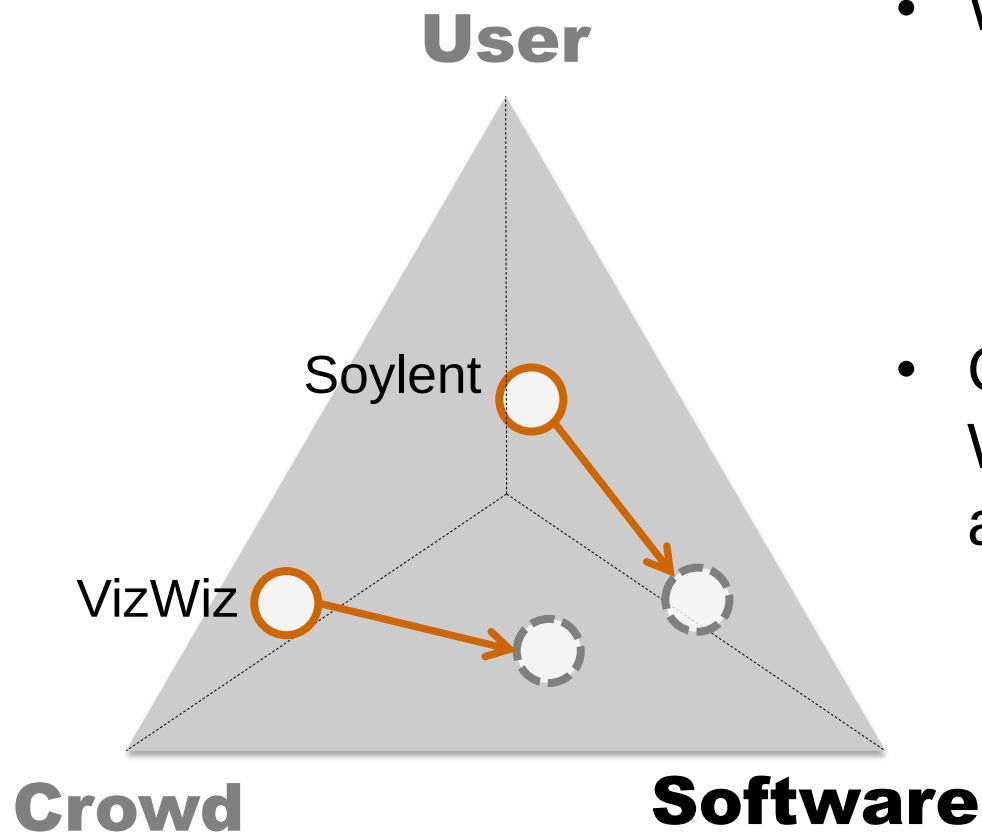
Crowd

Software

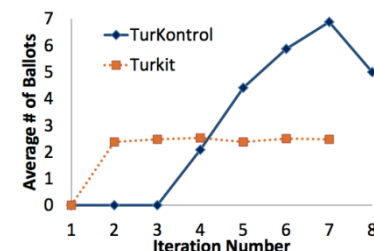
Crowd-Powered Design Space



Deployable Wizard of Oz



- Wizard of Oz
 - tried-and-true prototyping technique when we don't know how to write the software part of a system
- Crowd computing enables Wizard of Oz systems that are **useful** and **deployable**
 - collect data from real use
 - use AI for performance/cost improvement



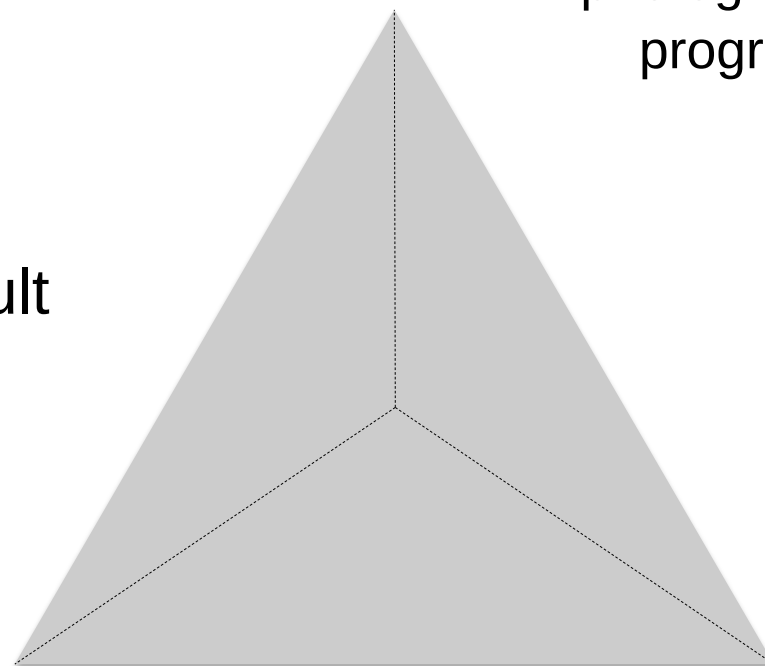
The User Plays a Key Role

- The end-user knows the goal and the context of the work
- The end-user provides coherence to a creative product
- The end-user has final responsibility for the result

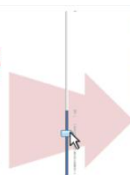
User writer
blind person
photographer
programmer

Crowd

Software



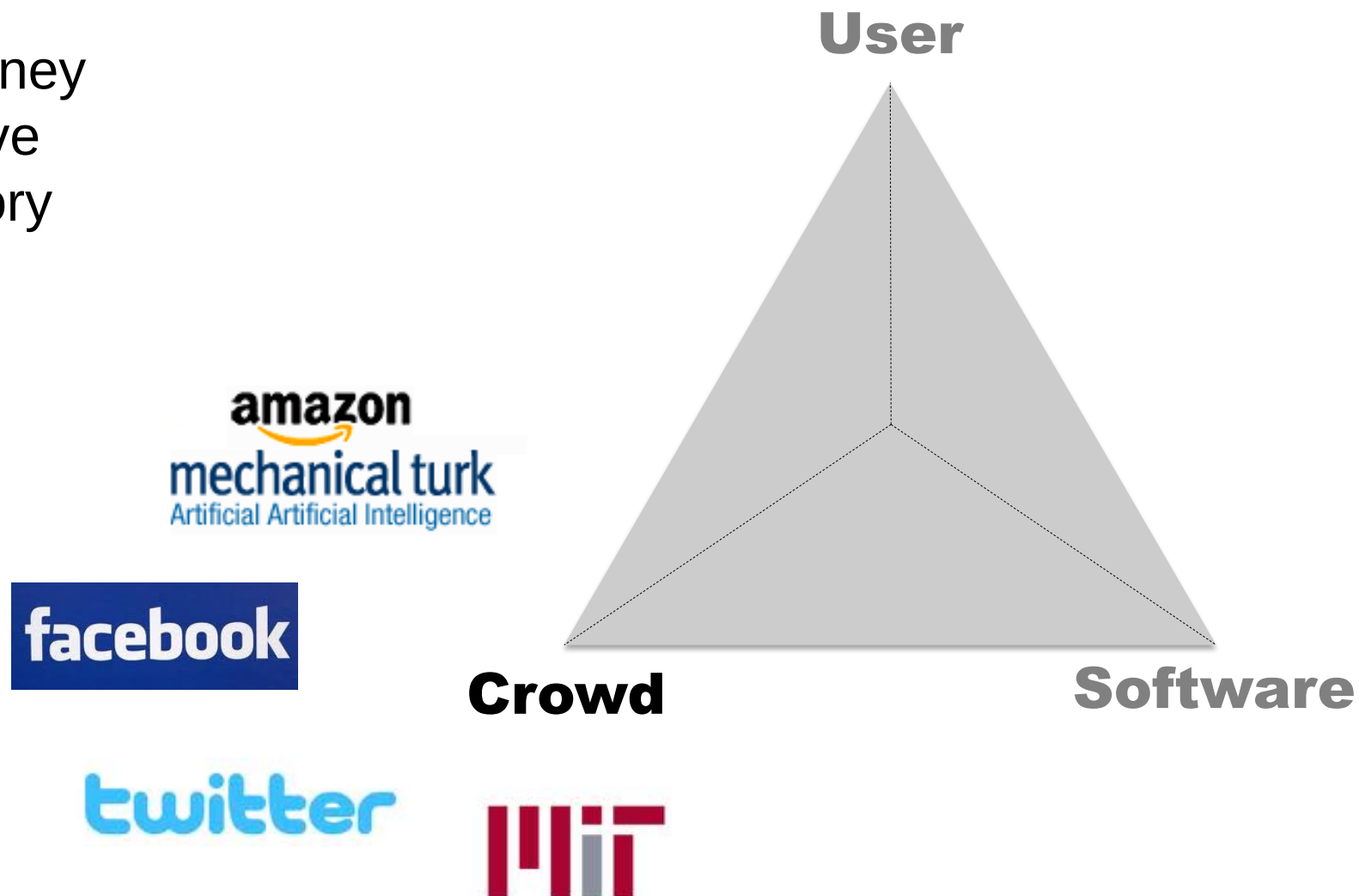
Automatic clustering generally helps separate different kinds of records that need to be edited differently, but it isn't perfect. Sometimes it creates more clusters than needed, because the differences in structure aren't important to the user's particular editing task. For example, if the user only needs to edit near the end of each line, then differences at the start of the line are largely irrelevant, and it isn't necessary to split based on those differences. Conversely, sometimes the clustering isn't fine enough, leaving heterogeneous clusters that must be edited one line at a time. One solution to this problem would be to let the user rearrange the clustering manually, perhaps using drag-and-drop to merge and split clusters. Clustering and selection generalization would also be improved by recognizing common text structure like URLs, filenames, email addresses, dates, times, etc.



Automatic clustering generally helps separate different kinds of records that need to be edited differently, but it isn't perfect. Sometimes it creates more clusters than needed, because the differences in structure aren't relevant to a specific task. Conversely, sometimes the clustering isn't fine enough, leaving heterogeneous clusters that must be edited one line at a time. One solution to this problem would be to let the user rearrange the clustering manually using drag-and-drop edits. Clustering and selection generalization would also be improved by recognizing common text structure like URLs, filenames, email addresses, dates, times, etc.

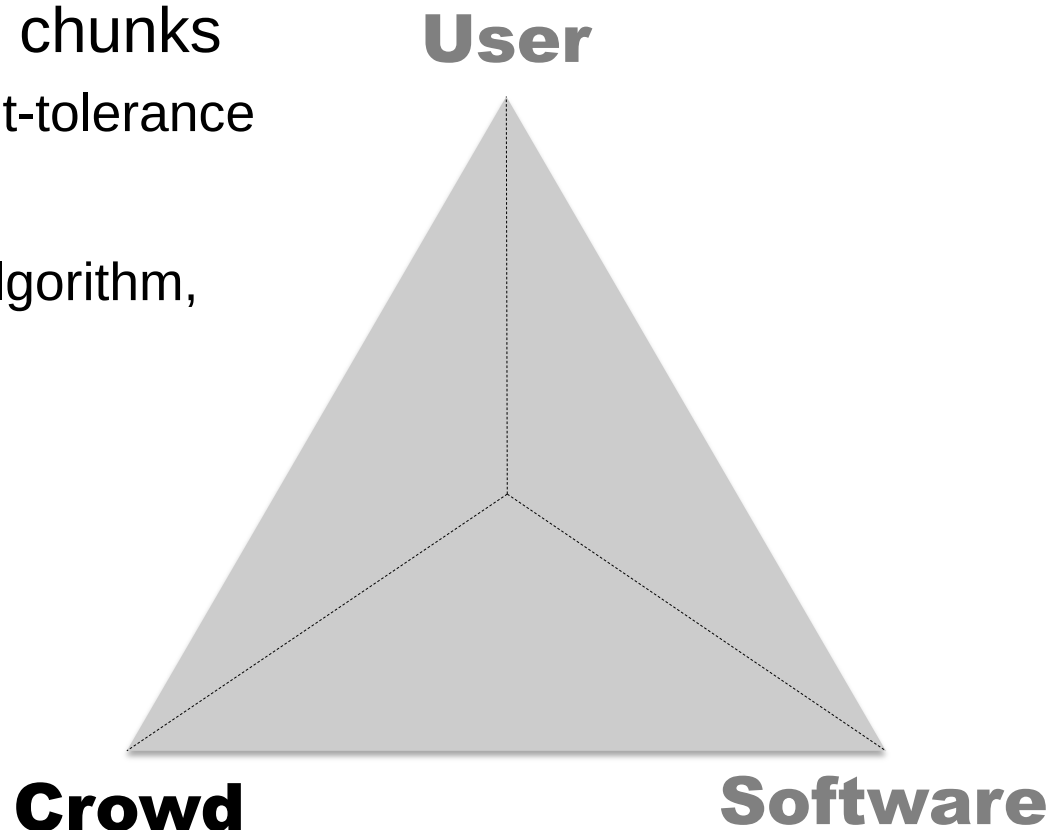
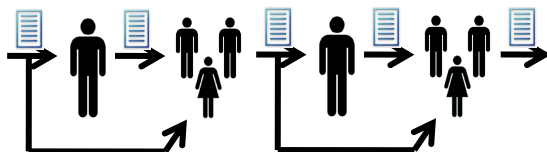
Many Crowds, Many Incentives

- Money
- Love
- Glory



Hints for Crowd System Design

- Divide work into small chunks
 - for parallelism and fault-tolerance
- Expect noise
 - either design a good algorithm, or refine the crowd

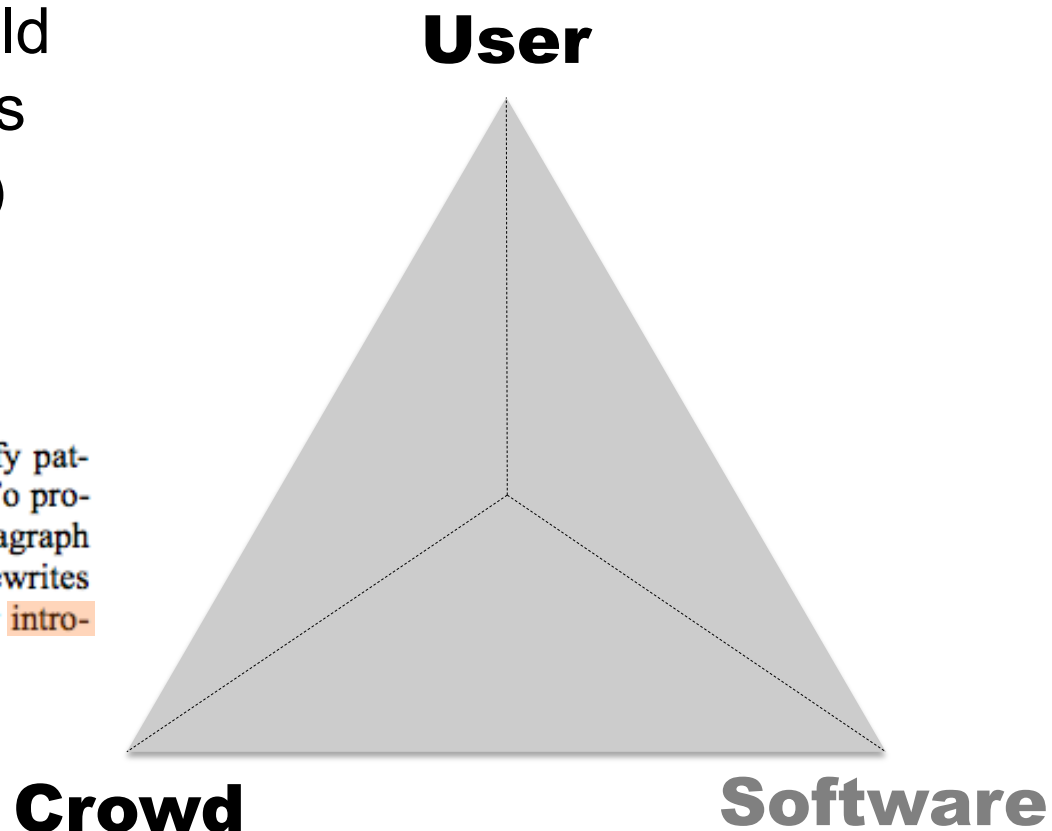


Crowds Are Human!

- Crowd's abilities should complement the user's
 - diversity (“many eyes”)
 - different competence

Find-Fix-Verify in Soylent

Both Shortn and Crowproof use the Find-Fix-Verify pattern. We will use Shortn as an illustrative example. To provide the user with near-continuous control of paragraph length, Shortn should **produce** many alternative rewrites without **changing** the meaning of the original text or **introduce**⁷ grammatical errors.



Ongoing Work

Driver



Searchers

Smartphone

Programmer

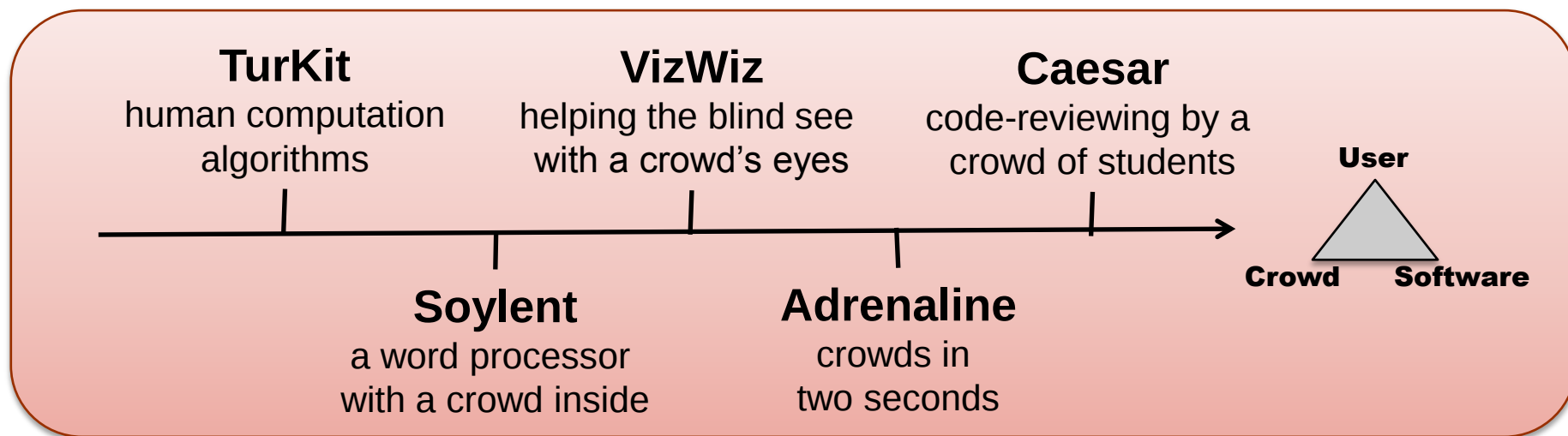
```
2 public static void main(String[] args) {  
3     System.out.println("Contributors:");  
4     System.out.println("Alic  
5     System.out.println("Bo  
6 }
```

Helpers

Eclipse

- Open questions in crowd computing
 - Scalability
 - Designing incentives for different kinds of crowds
 - Mobile crowds
 - Expert crowds
 - Task routing
 - Transitioning from crowd to AI

Conclusion



Thanks to support from NSF, Quanta Computer, Xerox
... and a crowd of more than 20,000 turkers