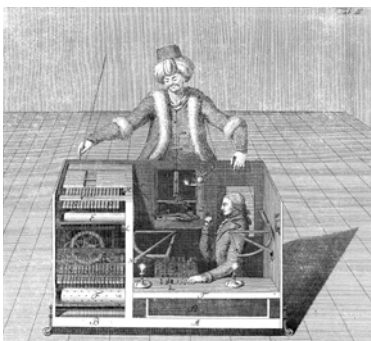


How do you turn millions of people into a CPU? **Lilly Irani** unravels the mysteries of “human-as-computation” in Amazon Mechanical Turk.

FROM **CROWD LABOR** TO **CLOUD LABOR**



WHEN WE THINK OF computing infrastructure, we think of server farms, personal computers, tangled cables, and operating systems. These are the machines that collect photos, videos, songs, and stories through ubiquitous technologies like Gmail and Facebook. As such data amasses in the wake of web 2.0, technologists have found limits to how far computers using artificial intelligence (AI) can organize and discriminate among such culturally significant data. It is trivial for a person to locate a puppy among a horde of cats and slightly more difficult to guess at family resemblances in a reunion photo, but training computers to perform such feats of cultural discrimination remains an open research problem. Twenty-five years ago Winograd and Flores (1986) declared such problems philosophically insurmountable for AI.

In recent years, however, technologists have found a new workaround to the limits of AI. The “human computation” movement in computer science has advocated for “leveraging the abilities of an unprecedented number of people via the web to perform complex computation” (Law & von Ahn, 2011: viii). The fruits of this research are familiar to anyone who has tried to log into a website only to be challenged with a distorted image of text. Website developers use those images, called CAPTCHAs, to discriminate real people trying to log into a site from password-guessing algorithms trying to break in; CAPTCHA stands for Completely Automated Public Turing test to Tell Computers and Humans Apart. CAPTCHAs succeed at blocking automated break-in attempts building on the observation that recognizing warped text is very hard for a computer but very easy for a literate human being. The more general desire to leverage these computers’ and humans’ differential capabilities are the foundation of the micro-task marketplace called Amazon Mechanical Turk (AMT).

The design of AMT emerged in the crucible of Amazon’s own computational demands for its business. Facing a website riddled with duplicate pages for single products, Amazon engineers had declared using artificial intelligence approaches “insurmountable” (Harinarayan, 2007). Instead, Amazon turned to human computers. To save the time and expense of hiring and managing large numbers of temporary work-

ers, Amazon engineers instead developed a website through which people could work simultaneously from their own computers to check each product for duplicates. Like home-based pieceworkers, these checkers were paid per product evaluated (Pontin, 2007). Workers completed tasks simple and unambiguous enough (ideally) to be completed without coworkers or direct managers; employers paid per unit produced and treated workers interchangeably. In all, the system amounted to a market for largely invisible cognitive pieceworkers. Keeping workers at a distance – here, by mediating them through anonymizing spreadsheets and Application Program Interfaces (API) – allowed Amazon to retain its existing divisions of labor and organizational practices. Where Amazon might have called AI through computer code, now they could call the labor pool similarly. Amazon’s CEO publicly announced AMT at MIT saying, “You’ve heard of software-as-a-service; well, this is human-as-a-service.” AMT preserved the social order of technologist-controlled computing, but enhanced this computing with human cognitive capabilities. Six years after Amazon made AMT available for public use, thousands of people do tasks on the service. Computer scientists at MIT and Berkeley have active projects developing databases, word processing tools, task design systems, and other complementary technologies building out the ecosystem of human computing.

So how do the thousands of people unacquainted with one another become a computing crowd? How does AMT extract “ground truth” data from situated cultural cognitions? How does AMT integrate potentially unruly masses into existing large-scale computing infrastructures? I will describe the various lightweight forms of probabilistic control that make AMT work and distinguish them from other highly-controlled computerized workplaces. This analysis builds on three years of experience with AMT through a combination of my role as a builder and maintainer of the AMT worker tool *Turkopticon*, informal interviews

with technologist employers, attendance at crowdsourcing conferences, and participation in worker web forums.

CROWD CONTROL: ACCOMPLISHING ACCURACY, SPEED, AND SCALABILITY

Accuracy: Technology builders privilege accuracy in the world of AMT. There are two varieties by which requesters try make workers accurate. The first is a sort of statistical objectivity; given the same question, accuracy means exhibiting “the most plural judgement,” in the words of then Director of Amazon Web Services Peter Cohen (Sadun, 2006). This can mean simply assigning several workers the same task and using majority vote to decide on the “true” answer, called “the gold standard,” or “ground truth” in Computer Science research. More complex mechanisms might try to take into account biasing parameters of the workers such as experience or location. In the end, however, requesters count the most plural as the most accurate and reward workers accordingly. AMT’s version of statistical objectivity is a shift in artificial intelligence and natural language processing research, which has traditionally used experts to authoritatively establish “gold standard” data sets (Snow et al., 2008).

The second form of accuracy inheres in tasks that involve subjective or personal data, such as surveys or aesthetic judgments. Requesters need to figure out which workers are making good faith judgments and which ones are “malicious,” clicking randomly for money, or trying to corrupt the dataset. AMT maintains an “acceptance rate” for each worker to help requesters recruit workers with high rates of task acceptance. However, large scale requesters use a number of other methods to discriminate “good faith” workers from the “malicious.” Most methods boil down to asking obvious questions or providing tasks for which a gold standard is already known. Large-scale requesters maintain databases, organized by alphanumeric ID, recording workers’ past performance, geolocation, and other parameters

to create blacklists or whitelists of workers.

One requester I interviewed, for example, put up a digital version of the game *Mastermind* as a task and found that it was a slightly better predictor of his workers’ accuracy than Amazon’s reported accuracy rate. But logical acuity is not the only relevant performance. Requesters often restrict workers country location as a proxy for filtering workers without the presumed-to-be-stable cultural literacies their subjective tasks require.

Accuracy is achieved not by training, disciplining, or surveilling workers, but instead by what the founder of one crowdsourcing firm, Rick (a pseudonym), calls “pure approaches” to crowdsourcing. Pure approaches open the system to all workers and use filtering and redundancy to sort the good workers from the unintelligible or malicious, the ground truth from the inaccurate, and the usable data from the spam.

Speed: A hallmark of how we think about computers today is speed – speed in processing, speed in communication. The speed at which AMT accomplishes low-skill or unspecialized tasks is key to its appeal to requesters who might otherwise bring on temporary workers or interns to do the job. The speed at which someone’s tasks can be completed on AMT hinges on the size of the crowd present to take on the tasks. A thousand AMT workers working for a single day can process data far faster than hiring ten temps for a hundred days. While a day is still far slower than the near-instantaneous response times we’ve come to expect from silicon computers, AMT speed will do when usable computer algorithms don’t exist and are difficult to create.

Throwing a large number of brains at a problem means having large numbers of people instantly on call. The reach of the Internet into each of the world’s time zones means the sun never sets on Amazon’s technology platform. Amazon also possesses a uniquely liquid global currency – Amazon Gift Certificates. Though US and Indian workers can get paid in dollars and

rupees respectively, workers from another hundred countries can redeem their earnings in Amazon credits. The global reach of Amazon's currency and website means that whenever someone places a task or makes a call to the marketplace, workers are there to process the tasks.

Scalability: AMT offers developers scalability – developers can command as much or as little human computation as they want, incurring little to no maintenance costs. This scalability stems in part from Amazon's pricing and payment model. AMT charges users per task, giving employers complete discretion to reject work as unsatisfactory and deny the worker payment. Scalability functions in several ways to consumers of AMT labor. For a large corporation doing machine learning categorization, like eBay or Amazon, they need a workforce that is large enough to quickly categorize major inflows of images in parallel during phases when engineers are focusing on improving system performance. For a small startup, scalability offers the promise of low operating costs when small, without sacrificing the promise to investors that the fledgling company can handle the success of rapid growth. The computational quality of scalability, then, is not only technical but also rhetorical.

Scalability also derives from the legal architecture within which the AMT technology is embedded. Amazon's terms of service are designed to allow requesters to pay for data and nothing more. Workers need to understand American English and have access to a computer and an internet connection, but requesters do not pay to train and maintain employees and infrastructure. They pay only for the data workers produce to their liking, and they can refuse on a whim. Amazon's Terms of Participation define workers as contractors providing "work for hire" at prices independent of minimum wage laws. In intent, "work for hire" laws exempt professional contractors from US labor protections with the assumption that those contractors operate and invest in independent businesses

that provide them with opportunities for profit, judgment, foresight, and risk taking. Employers in the US, however, have long attempted with varying degrees of success to specify home workers, pieceworkers, and other low paid workers as contractors as tactic for reducing labor costs (Felstiner, 2011).

On AMT, workers hand over completed work to employers, along with attendant intellectual property rights, regardless of whether the employer approves the work or chooses to pay. Employers can reject the work at their discretion; Amazon neither provides nor advocates for dispute resolution short of mandatory arbitration. Even tax reporting is essentially optional as long as requesters hire each individual worker for less than \$600 a year. Amazon's legal architecture leaves requesters free to focus on eliciting and extracting data accurately, quickly, and in a scalable manner. However, there are challenges to managing variously sized crowds of workers within a relatively fixed size organization. Large-scale requesters facing the challenges of managing these crowds are developing techniques I'm calling automatic management.

AUTOMATIC MANAGEMENT AND POLITICS WITH LARGE NUMBERS

For a small start up, managing a workforce of 60,000 people may seem an insurmountable challenge. Yet this is the challenge faced by large-scale requesters. Requesters building on AMT have developed and are constantly refining techniques to manage this workforce in a computer-automated fashion. For AMT to be scalable, the effort that goes into using AMT – setting up tasks, choosing workers, communicating with workers, and deciding who gets paid and who doesn't – must be manageable for someone who might commission 10,000 workers in the span of a few hours. Dahn Tamir, a large-scale requester, explains:

You cannot spend time exchanging email. The time you spent looking at the email costs more than what you paid them. This has to function

on autopilot as an algorithmic system...and integrated with your business processes.

One practice of automated management is "setting up incentives" so that workers self-select into tasks they are good at and learn to avoid tasks they are bad at. "You have to set up incentives right so everyone is aligned and they do what we want them to do. You do it like that, not by yelling at them," another crowdsourcer explained to me. In practice, "setting up incentives" means denying or reducing payment to those who provide work outputs that do not meet requesters needs. The choice of whether or not to pay is based on assessments of accuracy determined algorithmically and is registered through system calls or a spreadsheet upload to the AMT system.

Large-scale requesters also rely on automated filtering criteria, whether based on Amazon's limited worker information (e.g. task approval percentage) or detailed data they gather by interacting with workers. Workers who are below the requesters preferred threshold are simply never shown the task. Those requesters who have more intricate means of sorting "good" workers from the bad may blacklist the bad or whitelist the good. In either case, workers are sorted solely through their performance in the system. At the scale of workforce and the speed of micro-tasks that characterize AMT, there is little time for discipline and little opportunity to mold workers. Sharon Chiarella, VP of AMT, explained that minimal interaction and monitoring allows for efficient human resources management by reducing the decisions employers have to make while simultaneously ensuring that workers are not discriminated against on the basis of race or gender (Chiarella, 2009). This minimalism differs sharply from the surveillance and control of the panoptic, "informatized" workplace more typically described (Head, 2003; Zuboff, 1988). Instead, requesters sort desirable workers through faint signals of mouse clicks, text typed, and other digital traces read closely as potential indicators.

Within this large scale, fast moving, and highly mediated workforce, dispute resolution between workers and employers becomes intractable. Workers can contact the requester through a web form on Mechanical Turk if they are dissatisfied with a rejection; but requesters most commonly do not respond personally and Amazon requires no dispute resolution. Requesters have full discretion in choosing to pay workers or even blocking the worker permanently; by AMT's design logic, dispute resolution does not scale. In AMT's transactional logic of data elicitation and automated management, even dispute messages become informational rather than agonistic. Rick admitted that while dispute resolution on AMT scale is impossible, keeping dispute messages in the system gives his company a valuable signal about their algorithm's performance in managing workers and tasks. Disputes, then, become a signal to optimize automated management systems; in AMT as designed, worker struggle consists simply of exit.

Automatic management techniques in AMT are, in a sense, an automation of human resources departments. Recruitment, interviewing, and selection are replaced by an infrastructure that defines the terms of worker entry, entry, exit, and the production of cognitive commodities. These techniques build on much older forms of Taylorism and scientific management; they circumscribe the scope of workers contributions to the overall product as a way of centralizing process planning and consolidating authority in a managerial class (Head, 2003; Noble, 1977). AMT liberates technologists from disciplining workers face-to-face or negotiating over the best way to pursue a goal; it is continuous with the fragmentation of worker collectivities and the centralization of power. In AMT, the Taylorist manager becomes the computer systems programmer. But instead of the total systems view of the informed workplace (Zuboff 1988) advocated by Scientific Management, AMT is made efficient and pleasurable precisely by what employers do not have to know or think about.

AMT and the workforce that powers it become a pleasurable platform for computational innovation. Wendy Chun calls this "causal pleasure"—the sense of power and control a skilled user feels working on and through an operating system on a computer – a "microworld" (Chun, 2005; Edwards, 1996). "Human-as-a-service" places an assemblage of humans and computers under a technologist's interactive control to inspire the technologist's sense of creativity and exploration. As Tamir puts it: "You can try things...When I was wrong, it really didn't matter. I spent a few bucks. The loss was minimal." Accessing workers through APIs, according to founders of one AMT competitor, is key to enabling software engineers to innovate. AMT mediates access to crowds of workers, global competition keeps price-per-task low, and technologists manage those workers lightly, statistically, and expediently. The result is a stable, reliable and enabling infrastructure. In this system, specific workers' agencies – their wrong answers, their complaints, their unwillingness to take a low price, or their choice to leave the labor pool – are largely irrelevant in the operation of a system that structures work to treat people as fungible cognition. Without holding any particular person in "standing reserve" (Heidegger, 1977), AMT's standing reserve of human cognition is achieved. □

LILLY IRANI is a PhD Candidate in the Department of Informatics at UC Irvine.

ACKNOWLEDGEMENTS

Comments from Fred Turner, Chris Kelty, Caitlin Zaloom, Nick Seaver, and Kavita Philip, as well as comments from Paul Dourish's research group have shaped the work presented in this paper. All errors are my own.

REFERENCES

- Chiarella, S. 2009. Personal communication at Mechanical Turk Meetup, June 10. San Francisco, CA.
- Chun, W. 2005. "On Software, or the Persistence of Visual Knowledge." *Grey Room*, 18, 26-51.
- Edwards, P. 1996. *The Closed World: Computers and the Politics of Discourse in Cold War America*. Cambridge: MIT Press.
- Felstiner, A. 2011. "Working the Crowd: Employment and Labor Law in the Crowdsourcing Industry." *Berkeley Journal of Employment & Labor Law*, 32(1), 143-204.
- Harinarayan et al. 2007. U.S. Patent No. 7,197,459. Washington, D.C.: U.S.
- Head, S. 2003. *The New Ruthless Economy: Work and Power in the Digital Age*. Cambridge: Cambridge University Press.
- Heidegger, M. 1977. *The Question Concerning Technology and Other Essays*. New York: Harper & Row, 3-53.
- Howe, J. 2009. *Crowdsourcing: Why the Power of the Crowd is Driving the Future of Business*. New York: Random House.
- Gantz, J. 2008. "An Updated Forecast of Worldwide Information Growth Through 2011". IDC. Retrieved 2009-04-20.
- Law, E. & von Ahn, L. 2011. *Human Computation*. Morgan & Claypool Publishers.
- Noble, D. 1977. *America by Design: Science, Technology, and the Rise of Corporate Capitalism*. Oxford: Oxford University Press.
- Pontin, J. 2007. "Artificial Intelligence with Help from Humans." *New York Times*. Mar. 25, 2007. Accessed at <http://www.nytimes.com/2007/03/25/business/yourmoney/25Stream.html>
- Sadun, Erica. 2006. "Earning the Big Bucks (Not!) with Amazon's Mechanical Turk - O'Reilly Mac DevCenter Blog." *MacDevCenter.com* Apr 17. Web. 11 Dec 2009.
- Snow, R. et al. 2008. "Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks." *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 254-263.
- Winner, L. 1986. *The Whale and the Reactor: A Search for Limits in the Age of High Technology*. Chicago: University of Chicago Press.
- Winograd, T. & Flores, F. 1986. *Understanding Computers & Cognition: A New Foundation for Design*. Norwood, NJ: Addison Wesley.
- Zuboff, S. 1988. *In the Age of the Smart Machine: The Future of Work and Power*. Basic Books.