Opportunities for Crowdsourcing Research on Amazon Mechanical Turk

Jenny J. Chen, Natala J. Menezes, and Adam D. Bradley
Amazon Mechanical Turk
410 Terry Ave North
Seattle, WA 98109
{jennych, natala, abradley}@amazon.com

ABSTRACT
Many crowdsourcing studies have been conducted that utilize Amazon Mechanical Turk, a crowdsourcing marketplace platform. The Amazon Mechanical Turk team proposes that comprehensive studies in the areas of HIT design, workflow and reviewing methodologies, and compensation strategies will benefit the crowdsourcing field by establishing a standard library of repeatable patterns and protocols.

Author Keywords
Amazon Mechanical Turk, crowdsourcing, repeatability, HIT design, compensation, Worker Qualifications, result accuracy, work velocity.

ACM Classification Keywords
H.5.2 [User Interfaces], H5.3 Group and Organization Interfaces: Computer supported cooperative work; H5.3 Group and Organization Interfaces: Web-based interaction.

General Terms
Design, Experimentation, Human Factors

INTRODUCTION
Amazon launched its Mechanical Turk crowdsourcing marketplace platform in late 2005, and as the platform has grown and advanced, so has the corresponding research on crowdsourcing and the marketplace itself. Mechanical Turk has been utilized in studies ranging from human linguistic annotation to image classification and has been a topic of interest for the HCI, information retrieval, computer science, economics, and data mining research communities. In most studies, the platform has been primarily featured as the subject of a crowdsourcing study or used to generate data for other studies. While these research findings are of great interest to the crowdsourcing community, we believe there is an opportunity to analyze the dynamics of the marketplace and the uniqueness of crowdsourcing in the areas of Human Intelligence Task (HIT) design, multi-step task workflow and reviewing methodologies, and compensation strategies. These investigations should be designed, conducted, and published in a manner such that the research experiments can be repeated, potentially yielding standard design patterns and methods to achieve high quality, consistent results for a variety of human computation tasks.

HIT DESIGN
Mechanical Turk is engineered to be a flexible marketplace platform; therefore, a large number of variables can affect submitted HIT results. Requesters using the platform have control over three aspects of their HIT:

- HIT Settings
- Worker Qualifications
- HIT Layout Design

HIT Settings
HIT settings reflect the properties that Requesters can set when creating HITs and assignments on the Mechanical Turk platform. The properties include the time allotted for a Worker to complete an assignment, how long the HIT will be available in the marketplace, the reward amount for successfully completing an assignment, the number of assignments available for a given HIT in the marketplace, and the maximum time the Worker must wait for the results to be approved. Requesters adjust these properties as they react to changes in the marketplace and learn which combinations of settings produce their desired Worker behavior. When publishing research findings, HIT settings should be included to allow for replication of experiments and to establish verified protocols for various types of tasks.

Worker Qualification
Requesters can restrict the availability of their HITs by allowing only Workers with specified Qualifications to work on their HITs. Qualifications range from simple, such as geo-location of the Worker, to complex, such as tests that a Worker must pass in order to be granted the Qualification. Tests can also be used to train Workers and to acclimate them to the expected HITs. The growth and maintenance of qualified pools of Workers is an area that could also benefit from further investigation. Similar to having established cell lines in biological studies, an established group of Workers could be re-used by several different studies and remove the need to independently create a group of trusted Workers for each study; although, the durability of group

member participation may pose an interesting challenge to such a practice.

HIT Layout Design

HIT layout design has significant influence on a Worker's ability to complete HITs quickly, effectively, and accurately. Due to the large amount of variability in the design of a HIT layout, research in this area would be useful in developing design patterns that identify best practices for achieving price, accuracy, and speed goals. We believe this research falls primarily into three categories: HIT ergonomics, HIT instructions, and defensive design.

HIT Ergonomics

It is important to design the visual appearance of the HIT so that Workers can quickly and easily understand what is expected of them. When the HIT is structured in a manner that allows Workers to efficiently complete the task, they are able to submit results at higher velocities. In order to have more control over the visual presentation, many Requesters opt to create externally hosted HIT displays using the Mechanical Turk Requester API so that they can host the HIT display on their own web servers. Research in this area should investigate whether there are certain usability standards that all HITs should adhere to and if these standards and design patterns vary by task type. For example, a photo moderation task and an image categorization task both require image labeling, but the former may result in a visual layout that allows Workers to quickly moderate several images, while the latter may be best presented through a layout that allows images to be grouped into various category buckets.

HIT Instructions

In addition to the actual task itself, Requesters provide Workers with a set of instructions that detail how to complete the work. These instructions explain requirements for good answers and which kinds of answers may be rejected. Well-designed instructions impact the Worker experience and submitted work quality, as evidenced in studies like [1] which found that simplifying and adding clear images to instructions improved completion rates and submission quality.

In the left-hand side of Figure 1, Requester QuestionSwami has designed a content generation HIT that integrates detailed instructions along with real-time validations of Worker input. This Requester has reported that the quality of their submissions has increased significantly as a result of their improved HIT design and detailed instructions. In contrast, the content generation HIT on the right-hand side displays a long and confusing set of instructions with no built-in validations. The Requester for that HIT claims to get very mixed-quality results from Mechanical Turk. Research in this area can establish a set of crowdsourcing task design patterns that Requesters could use to format HIT layouts and result in instructions that consistently achieve a positive Worker experience and high quality work submissions.

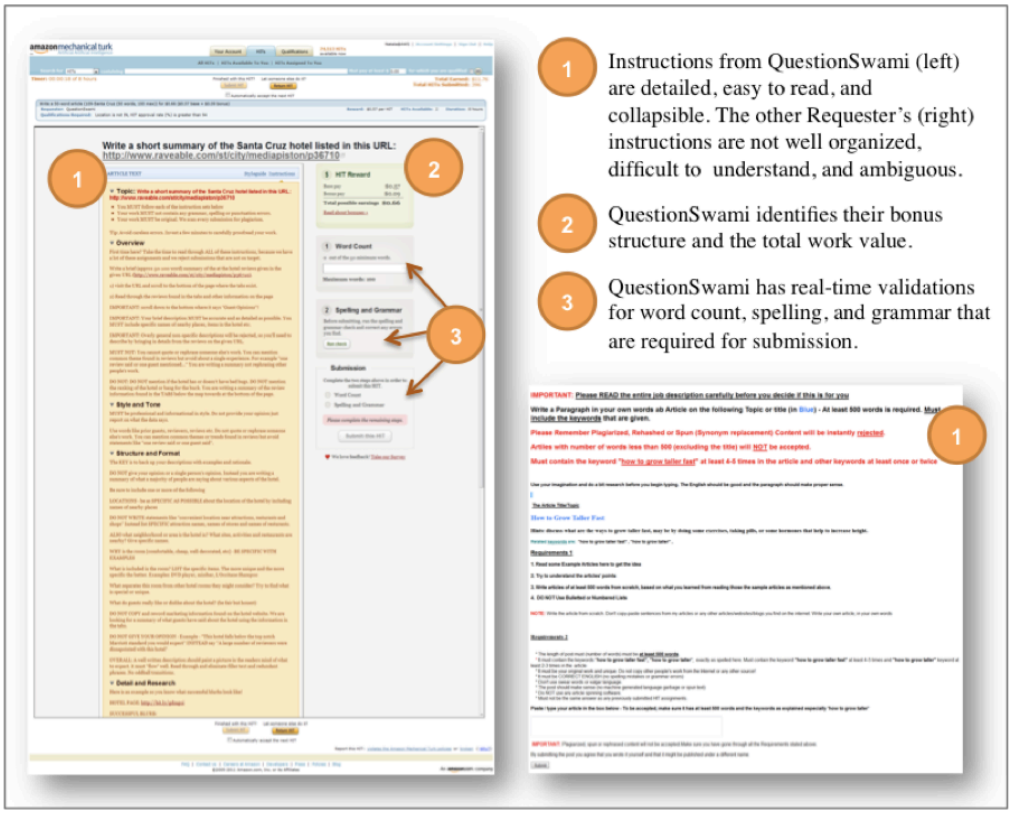

Figure 1. Comparison of two content generation HITs

Defensive Design

As marketplaces typically involve the exchange of goods, they tend to also attract unsavory participants who choose to exploit the system for deceptive purposes. HITs should be defensively designed so that it is difficult for Workers to submit low-quality HIT results and create noise in the result data. Kittur et al. asked Workers to assess the quality of Wikipedia articles [2]. In order to ensure that Workers actually read the article and did not blindly submit quality scores, the HIT also required Workers to answer specific questions about the article, such as the number of references, images, and sections, as well as provide keywords to summarize the contents of the article. The answers to the questions were known by the Requester's HIT reviewing system, which would reject the Worker's submission if the questions were not answered correctly. These mechanisms (variously referred to as "gold standards", "canaries", "invisible CAPTCHAs", or "validation methodologies") are widely used, and if executed well, provide Requesters with validation of the Worker's performance. This strategy allows Requesters to develop confidence in the submitted work quality, but may need to be combined with other techniques. Research in this area should explore the boundaries, costs, and benefits of this and other patterns for defensive human computation, particularly when the Requester only has limited a priori knowledge about the content of the majority of their HITs.

WORKFLOW AND REVIEWING STRATEGIES

In many cases, before a Requester begins to focus on HIT design, they first need to break their project into smaller components that are often handled by multiple HITs and managed through a review workflow. This is often referred to as task or project decomposition and if done correctly, can greatly improve the quality of submitted work.

Bernstein et al. has established a HIT workflow pattern for proofreading and editing text: Find-Fix-Verify [3]. When they asked Workers to edit text with free-access, they found that 30% of the submitted HIT results were unusable because some Workers put in minimal effort to complete the HIT and did not submit work of usable value. However, when the task was broken down into three separate HIT workflow stages, they found that they could make use of the HIT results from Workers who put in minimal effort. Find-HITs asked Workers to highlight areas of text that could be shortened. Fix-HITs asked Workers to edit and shorten the highlighted area without changing the meaning of the text. Finally, Verify-HITs asked Workers to flag edits that changed the meaning of the text.

Little et al. has initiated investigations into iterative and parallel workflows for human computation tasks [4]. Initial findings indicate that iterative workflows improve the average results for refining tasks, such as writing and collective-brainstorming, while parallel workflows yield the best results for creative tasks, such as transcribing blurry text and individual-brainstorming. Given the numerous possibilities for human computation tasks, the space of workflow patterns has great potential for broader exploration.

Requesters also impact the quality of their results through the reviewing strategy that they employ:

- *Single assignment* – This is the most basic process where a Requester asks a single Worker to complete a task and uses the answer that the Worker submits. Because of Worker behavior variability, this technique is best only when the HIT is restricted to a trusted group of Workers.
- *Forced agreement* – The Requester distributes two assignments for each HIT. If both answers match then the results are considered usable. A well-known forced agreement example is Luis von Ahn's ESP game, which displays the same image to two different users [5]. Users submit potential labels for the image until both users have submitted the same word. The game accepts the submission only when both users have submitted the same word or elect to pass to a new image.
- *Plurality* – The practice of using Worker agreement on answers is commonly accomplished by distributing a task to two Workers, and if the submitted answers match, then the results are used. If the answers do not match, a third tiebreaker assignment is created. Other techniques create multiple assignments and use the majority answer or apply statistical techniques to assess the reliability of each worker and the probable correct answer.
- *Expert review* - Requesters can employ a trusted Worker group to review the results of other Workers. The review HIT usually takes less time to complete than the original HIT, and a smaller set of trusted Workers can validate the results from a larger set of Workers.
- *Known answer question* – Known answer questions are questions where the Requester knows the answer and can be inserted as separate HITs or as questions within individual HITs. If a Worker fails a known answer question, Requesters can choose to invalidate all of that Worker's results, only the recent submissions, or just that particular submission.

These reviewing strategies can be used in combination. For example, if a plurality of two assignments does not result in a match, instead of creating a third tiebreaker assignment, the HIT can be repackaged for a trusted expert Worker to review. Or in a photo moderation task, inserting a *known answer question* can be used to determine if a Worker gave an honest effort to complete the HIT and combining that with *plurality* to gather submissions from multiple Workers to determine the correct answer for that HIT. A given reviewing strategy may not be effective for all categories of HITs. It is unlikely that *plurality* or a *known answer question* would be good solutions for a HIT that asks for a restaurant review, as content generation HITs typically lack a definitive answer. This list of reviewing strategies is by no means exhaustive. Requesters can leverage work history

and Qualifications, as well as other techniques, such as statistical methods and machine learning algorithms, to assess a Worker's HIT submission. Further studies should be conducted on the effectiveness of various reviewing strategies and which methods are best suited for different types of tasks.

COMPENSATION STRATEGIES

Compensation is a significant factor within the relationship between Requesters and Workers. Even if Requesters communicate with Workers through other means, such as email or forum postings, compensation is still the most influential factor in work velocity and Worker satisfaction. For many Workers, the monetary reward is their primary motivation and their approval rating is secondary. A few compensation strategies include:

- *Baseline Rewards*: The reward compensation is the advertised reward price.
- *Bonus Rewards*: In addition to the baseline reward, Workers can accrue additional compensation through a bonus. Bonuses can be distributed in various ways. Quality-driven bonuses depend upon a quality assessment of submitted work. Randomly allocated bonuses are usually achieved through a lottery system. Guaranteed bonuses are given when Workers attain an achievement, such as submitting 100 assignments or reaching the top of a leader-board.
- *Negative Rewards*: If the Worker's submission is bad, then the Worker is a paid a lower than advertised price. This is technically not possible on the Mechanical Turk platform, but Requesters can partially simulate this by advertising a low price plus a bonus and paying bonuses for all submissions except for ones that are poor in quality. Another approach is to reject the Worker's low-quality submission, but grant a smaller bonus.

In addition to the monetary reward for submitting an assignment, the manner and timeliness in which assignments are approved affects work velocity. Some Requesters choose to approve all assignments as a method of increasing a batch's completion velocity; however, this strategy often encourages the submission of invalid work, lowering the quality of results. Many Requesters choose to reject assignments that are not valid or useful, which means that a Worker who did the task is not paid. This can improve accuracy by discouraging Workers that submit low-quality work from participating. Understanding the effect that different compensation strategies have upon work quality and velocity is an area that has some study to date but could benefit from deeper investigation.

CONCLUSION

We posit that extensive research in HIT design, task workflow and review, and compensation strategies is needed in the crowdsourcing field. HIT design patterns can increase work quality for many crowdsourcing platforms, including Amazon Mechanical Turk. Discovering effective combinations of task workflow and review strategies and compensation methodologies will improve the marketplace experiences for both Requesters and Workers. Published research findings should include extensive detail regarding human computation experiments. This disclosure will establish repeatable protocols and libraries of HIT design patterns and allow experiments to be independently verified by multiple researchers and leveraged by the crowdsourcing community.

ACKNOWLEDGMENTS

We are very grateful to the members of the Amazon Mechanical Turk team who provided feedback and ideas, particularly James Willeford and Rob Showalter. We would also like to thank the members of the crowdsourcing community and the thousands of Workers and Requesters who have helped us build a dynamic marketplace.

AUTHORS

Jenny J. Chen is an Engineer for Amazon Mechanical Turk. She received a B.S. in Computer Science from Stanford University.

Natala J. Menezes is a Senior Product Manager for Amazon Mechanical Turk. She received a B.A. in Philosophy from the University of Minnesota, Twin Cities.

Adam D. Bradley is a Senior Engineer for Amazon Mechanical Turk. He received a Ph.D. in Computer Science from Boston University and a B.S. in Computer Science from the University of North Carolina at Asheville.

REFERENCES

1. Khanna, S., Ratan, A., Davis, J., and Thies, W. Evaluating and Improving the Usability of Mechanical Turk for Low-Income Workers in India. In *Proc. DEV 2010*, ACM Press (2010).
2. Kittur, A., Chi, E.H., and Suh, B. Crowdsourcing User Studies With Mechanical Turk. In *Proc. CHI 2008*, ACM Press (2008), 453-456.
3. Bernstein, M.S., Little, G., Miller, R.C., Hartmann, B., Ackerman, M.S., Karger, D.R., Crowell, D., and Panovich, K. Soylent: A Word Processor with a Crowd Inside. In *Proc. UIST 2010*, ACM Press (2010), 313-322.
4. Little, G., Chilton, L.B., Goldman, M., and Miller, R.C., Exploring Iterative and Parallel Human Computation Processes. In *Proc. HCOMP 2010*, ACM Press (2010), 68-76.
5. von Ahn, L. and Dabbish, L. Labeling Images with a Computer Game. In *Proc. CHI 2004*, ACM Press (2004), 319-326