Enhancing Reliability Using Peer Consistency Evaluation in Human Computation

Shih-Wen Huang and Wai-Tat Fu
University of Illinois at Urbana-Champaign
Department of Computer Science
201 N Goodwin Avenue
Urbana, IL 61801
{shuang51, wfu}@illinois.edu

ABSTRACT

Peer consistency evaluation is often used in games with a purpose (GWAP) to evaluate workers using outputs of other workers without using gold standard answers. Despite its popularity, the reliability of peer consistency evaluation has never been systematically tested to show how it can be used as a general evaluation method in human computation systems. We present experimental results that show that human computation systems using peer consistency evaluation can lead to outcomes that are even better than those that evaluate workers using gold standard answers. We also show that even without evaluation, simply telling the workers that their answers will be used as future evaluation standards can significantly enhance the workers' performance. Results have important implication for methods that improve the reliability of human computation systems.

Author Keywords

Crowdsourcing; Human Computation; User Behavior; Mechanical Turk; Evaluation

ACM Classification Keywords

H.5.3. Information Interfaces and Presentation (e.g. HCI): Group and Organization Interfaces

INTRODUCTION

Human computation [15] is a technique that utilizes human activities to replace traditional digital computers to perform certain computational tasks. The benefit of human computation is that some problems that are often difficult for machines can be accomplished relatively easily for humans (e.g., vision, natural language processing). The ESP game [29], for example, encourages players to generate image labels while enjoying the game. Human computation is also used for security [1], speech-to-text translation [17], and generation of labeled data for natural language processing [27]. These examples have merely begun to demonstrate the potential of human computation as a social computing technique that can be applied in a wide range of situations.

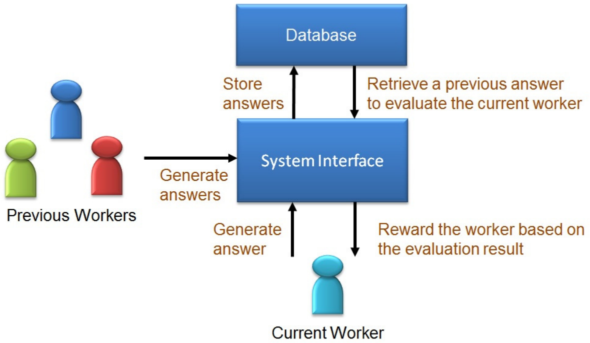

Database
Store answers
Retrieve a previous answer to evaluate the current worker
System Interface
Generate answers
Previous Workers
Generate answer
Reward the worker based on the evaluation result
Current Worker

Figure 1. A graphical representation of peer consistency evaluation. Every worker in the system is evaluated by the answers provided by the previous workers, which means it is a very scalable mechanism.

One of the biggest challenges for developing effective human computation systems is the reliability of the outcomes [11]. Unlike traditional computational systems, workers in a human computation system may fail to reliably generate correct outputs for many reasons. One notable reason is that workers may not have the incentives to put forth enough effort to finish the task assigned to them [24]. Another reason is that the instructions given to the workers may not be clear enough for them to follow [23]. These factors have made it difficult to build a reliable human computation system. The potential unreliability may in general decrease the utility of human computation systems. Systematic testing of methods that can potentially enhance the reliability of human computation systems is therefore critical for their success.

Peer consistency evaluation in output agreement games

Games with a purpose(GWAP) [30] harness the power of human computation by making their players work while enjoying the games. An output agreement game is one of the earliest and most popular forms of GWAP. A typical output agreement game has the following procedure: the game first randomly matches multiple players and provides them the same set of inputs, which can be images [29], words [26], or any

CSCW'13, February 23–27, 2013, San Antonio, Texas, USA.

data that the game designer wants to label. Then, the players start to generate outputs that are related to the inputs. The players will be rewarded if the outputs generated by different players reach a certain level of agreement. When the game ends, some versions of the agreed outputs will be collected as the labels that describe the data.

Output agreement games use the *consistency between the players to ensure the quality of the collected data*. Although previous research questioned whether peer consistency evaluation used in an output agreement game can persuade workers to generate high-quality outputs [25], recent research [9] has shown that the gaming environment not only could encourage the players to generate answers that matched other players' responses, but could also encourage players to generate high-quality answers.

These results allow us to hypothesize that human computation systems with peer consistency evaluation could outperform those without any evaluation because peer consistency evaluation provide evaluation standards that can approximate correct answers. This is our first hypothesis.

H1: *Peer consistency evaluation can serve as a valid evaluation method to motivate workers to generate outcomes with high-quality. Therefore, workers with peer consistency evaluation have better performance than those without evaluation.*

Peer consistency as a better evaluation mechanism

Existing human computation systems often use gold standard answers to evaluate their workers [6, 16, 22, 24]. A gold standard answer is an objectively determined correct answer. For example, gold standard answers for a part-of-speech (POS) tagging task can be the correct POS of each word in the sentences or articles of the tasks determined by human experts (e.g., linguists). Therefore, gold standard evaluation is considered one of the most objective mechanisms that can accurately measure the performance of workers.

However, gold standard answers of a task can be very costly [22]. Thus, people often compromise by mixing a few tasks that have the gold standard answers with the majority of tasks without them, and evaluation is often only based on those that have the gold standard answers. This may decrease the effectiveness of the evaluation because repeated workers can find out and only work hard on the evaluation questions [22]. On the other hand, peer consistency evaluation directly utilizes outputs of previous workers to evaluate outcomes of future workers, such that the system can generate abundant questions to evaluate workers. Therefore, an important focus of the current article is to investigate the extent to which peer consistency evaluation can be as effective as gold standard evaluation to encourage high quality outcomes from workers, because *peer consistency evaluation is much more scalable and cost-effective.*

Compared with gold standard evaluation, a potential advantage of peer consistency evaluation is that it may also increase the perception that workers are connected to other workers in the system, as workers are informed that their answers will be evaluated by the answers of others and their answers will also be used to evaluate future workers' answers. *The implicit social interactions among workers* could serve as a form of social motivation for workers to work harder. For example, workers may be motivated to match the performance of others - a competitive motive; or to provide good answers to evaluate future workers - an altruistic motive [9]. Peer evaluation *also makes the rewards of the workers dependent on each others' answers*. This is because if one worker does not generate the correct answer, the other workers would be incorrectly penalized due to this erroneous evaluation standard. This can be another incentive for them to put more effort into the tasks because it is possible that workers may have the altruistic tendency to adjust their effort level to avoid unfair evaluation to future workers [2]. As a result, workers may realize that their performance not only affect their personal reward, but also the reward of their "colleagues".

Therefore, the altruism of workers can provide another incentive to motivate them in generating high-quality outcomes, which is our second hypothesis:

H2: *Peer consistency evaluation can further motivate workers utilizing the altruism of them. As a result, the workers with peer consistency evaluation can perform better than those with gold standard evaluation.*

On the other hand, this may also increase the perception of unfairness in the evaluation because it sometimes incorrectly penalizes the workers – i.e., if previous workers provided bad answers, the new workers might be penalized even if they generate good answers. The perception of unfair evaluation may decrease the incentives for the workers to provide good answers. This leads us to the third hypothesis:

H3: *The unfairness of peer consistency evaluation can discourage workers to put effort in generating good work, which harms the reliability of the system.*

The experiments

In the current study, we aim at examine the three hypotheses mentioned above by experiments that teased out different effects.

Our first experiment tested whether systems with peer consistency evaluation can generate reliable outcomes compared with systems with gold standard evaluation and with no evaluation (**H1**). Because it is possible that financial incentives may interact with the evaluation method, we also created three bonus levels in each evaluation condition to investigate their effects. Moreover, our second experiment separated the effects of evaluation and altruism of peer consistency evaluation to see if altruism itself was effective in enhancing worker performance (**H2**). Finally, we analyzed how the unfairness (incorrect penalty) affected the performance of workers in peer consistency evaluation (**H3**).

RELATED WORK

Human computation and crowdsourcing are two highly overlapped fields. However, the focus of the fields are slightly different – human computation tends to focus more on replacing digital computers with humans while crowdsouring focus more on replacing traditional human workers with online workers [24]. Therefore, the research in crowdsourcing often focuses more on managing the workflow to solve complex work [12, 13, 14]. On the other hand, research in human computation focuses more on generating reliable answers to questions that are difficult for digital computers to solve [24]. Given our goal is to understand tasks that are often difficult to handle by digital computers (e.g., natural language processing), we choose to use the term human computation throughout. We, however, believe that research in crowdsourcing is highly relevant to our research.

Many approaches have been proposed to enhance the reliability of human computation. Some researchers used statistical methods to eliminate errors and biases of the outcomes generated by the workers [3, 7, 11, 18, 27]. Other researchers proposed different task designs to induce workers to produce high-quality outputs. Huang *et al.* [8] proposed a prediction model that helped the requester achieve an optimized task design by varying the variables in the design (e.g. price, number of tasks). Dow *et al.* [4] showed that the instant self-assessment and external-assessment could enhance the quality of the works. Sun *et al.* [28] found that it is easier for workers to pick the better answer than generate the good answer. Therefore, they designed a tournament selection method that asks the workers to pick the better answers at each round. Mason and Watts [21] studied the relation between the financial incentives and the performance of the workers. They found that the amount of financial incentives did not affect the performance of the workers, but some compensation schemes did yield better outcomes. Lin *et al.* [19] designed AGENTHUNT, a system that can dynamically switch between alternative workflows to achieve better worker performance. Little *et al.* created Turkit [20], a framework that allowed the workers to iteratively improve the previous workers' work. Liem *et al.* [17] further incorporated a dual pathway structure to the system. The dual pathway structure evaluated the workers in one path using the works of the workers in the other path, and rewarded the workers based on the similarity between their works. They showed that the system with their proposed structure could achieve 96.6% accuracy on a speech-to-text transcription task. Although they did show that a system with peer consistency evaluation could have good performance, they focused on a particular application and did not analyze how and why peer consistency evaluation could enhance the reliability of the systems. Therefore, the main contribution of their study and ours differ.

Gold standard Evaluation

Gold standard evaluation is another mechanism that often adopted by human computation researchers [6, 16, 22] and industry (e.g., CrowdFlower[1]) to ensure the quality of human computation outcomes. Le *et al.* [16] studied how to spread the gold standard evaluation questions to enhance the reliability of collected outcomes. Harris [6] also showed that providing positive incentives for workers to match gold standard answers yield better works. Oleson *et al.* [22] noticed that gold standard data is difficult and costly to get, but the size of gold standard data is important to quality assurance. They solved this problem by using previous matched answers with high confidence as "programmatic gold" to expand the size of gold standard data. Though both programmatic gold and peer consistency evaluation use answers of previous workers to evaluate newer workers, the former mechanism aims to create more gold standard answers in a cost-effective way, while the latter directly tell the workers that their answers will be evaluated by an answer of a previous worker. Therefore, programmatic gold cannot benefit from the possible social interactions between the workers of peer consistency evaluation.

To the best of our knowledge, our study is the first one that makes a systematic comparison between peer consistency evaluation and traditional gold standard evaluation and proves that peer consistency evaluation is a valid mechanism to enhance the reliability of human computation.

EXPERIMENTAL DESIGN

The goal of our experiment is to see whether peer consistency evaluation can improve the reliability of human computation systems. We recruited 270 subjects from Amazon Mechanical Turk (AMT)[2], a platform that recruits online workers to solve human intelligence tasks (HITs) [10]. We asked the workers to perform tasks on systems with no evaluation (N), gold standard evaluation (GS), and peer consistency evaluation (PC). We also varied the level of bonus the workers could receive when passing the evaluation to study if there were interactions between the financial incentives and evaluation methods.

Noun counting task

The workers in our experiment were asked to count the words with a particular part-of-speech (noun) in a list of 30 words. Each word in the word list had equal probabilities to be a noun or not a noun. We created 1,000 tasks, and the answers to the tasks ranged from 7 to 23. Since we recruited 270 workers in our experiment and each worker completed five tasks, 1,350 tasks were randomly picked from the task pool. There were two advantages of this task that made it a very good test-bed for us to examine the ability of the evaluation mechanisms to enhance reliability. First, it had a ground truth (objectively correct answer) to serve as the gold standard, so the quality of an answer could be measured by the difference between the answer and the gold standard. Moreover, the workers needed to put in a certain level of effort to solve the task[3]. This task is also representative of the tasks used in previous human computation systems (e.g., [27, 29]). An example interface of the task is shown in Figure 2.

Evaluation mechanisms in the systems

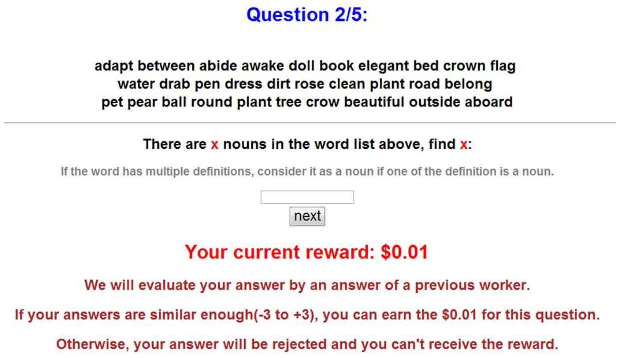
In our experiment, we compared the systems with three evaluation mechanisms:

- **No evaluation (N)**: The system with no evaluation was used to imitate a basic human computation system. The workers who worked in the system received a bonus regardless of the quality of the answers that they generated.

[1] https://crowdflower.com/

[2] https://www.mturk.com/mturk/welcome

[3] The difference between an average answer and the gold standard was 3.78, which supports our argument.



Question 2/5:

adapt between abide awake doll book elegant bed crown flag
water drab pen dress dirt rose clean plant road belong
pet pear ball round plant tree crow beautiful outside aboard

There are x nouns in the word list above, find x:

If the word has multiple definitions, consider it as a noun if one of the definition is a noun.

next

Your current reward: $0.01

We will evaluate your answer by an answer of a previous worker.

If your answers are similar enough(-3 to +3), you can earn the $0.01 for this question.

Otherwise, your answer will be rejected and you can't receive the reward.

Figure 2. An example interface of the noun counting task

- **Gold standard evaluation (GS)**: In the system with gold standard evaluation, the answers generated by the workers were evaluated by the correct answers. The workers earned a bonus if their answer was accurate enough. We allowed the workers to make some minor mistakes by accepting the answers with difference to the gold standards less than or equal to three[4].

- **Peer consistency evaluation (PC)**: Peer consistency evaluation was similar to gold standard evaluation, but the standards used to evaluate the workers were different. When a worker was working with a system utilizing peer consistency evaluation, the system first randomly picked a previous worker and used the answer generated by that previous worker as the standard of evaluation. Then, the worker who was performing the task could earn a bonus if the difference between the answer of the current worker and the evaluation standard was less than or equal to three (the same difference as in gold standard evaluation).

Bonus levels

The workers who worked in the experiment could earn a low bonus ($0.01), a medium bonus ($0.05), or a high bonus ($0.10) if they passed the evaluation of the task. We chose these three bonus levels based on an empirical study [10] that shows that more than 90% of the HITs on AMT have a price tag less than $0.10, and 70% of them have a price tag less than $0.05. These are also the three financial incentive levels used in [21]. Of course, the reasonable price of a task should be determined by many factors (e.g., difficulty of task, interestingness of work). However, the purpose of this manipulation is to understand the extent to which the effects of different evaluation methods might interact with those induced by different levels of financial incentives.

Subject recruitment on Amazon Mechanical Turk

We published our HITs on AMT from 5/13/2012 to 5/22/2012. The title and the description of these HITs were both ”Count the nouns in a word list”. We did not require our workers to pass any qualification tests, but we did require them to have more than 100 approved HITs and a HIT approval rate higher than 95%. This is a quality assurance mechanism usually used by requesters in AMT, and it prevented the results of our experiment from being affected by too many spammers. The price tags for the HITs were $0.05, which were the rewards the workers could earn regardless of their performance. On the HIT page, we provided a link to route the workers to our experiment website and a text area for the workers to enter a unique eight-digit completion code. When the workers completed their jobs, our website showed them the total bonus they could earn for their performance and the completion code that they could enter in the original HIT page on AMT. We then paid the workers both of their rewards (performance independent) and bonuses (performance codependent) based on the completion codes they entered.

Experimental interface

After the workers visited the experiment website, they were randomly assigned to one of the 9 conditions[5]. On the introduction page, the system told the workers that they had to answer 5 questions about counting the nouns in a word list. It also showed how much bonus they could earn for each question. After starting the task, the workers could see the interface for performing the noun counting task (Figure 2). In the upper part of the interface, the system showed a word list and an input box for them to provide their answers. The system specifically told the workers to count a word as a noun if one of its definition is a noun because it is possible for a word to have multiple definitions and only some of them are nouns. This made the gold standard answer of the question clearer to the workers. The system also told the workers how it would evaluate them in the lower part of the interface. The exact words used in this part by different systems are summarized below:

- **No evaluation (N)**: You can earn bonus[6] for providing the answer.

- **Gold standard evaluation (GS)**: We will evaluate your answer by the correct answer. If your answer is similar enough (-3 to +3) to the correct answer, you can earn bonus for this question. Otherwise, your answer will be rejected and you can’t receive the reward.

- **Peer Consistency Evaluation (PC)**: We will evaluate your answer by an answer of a previous worker. If your answers are similar enough (-3 to +3), you can earn the bonus for this question. Otherwise, your answer will be rejected and you can’t receive the reward.

When the workers finished each question, the system informed the workers whether they passed the evaluation or not. If they passed the evaluation, the system added the bonus of the question to the total bonus earned by the workers and show it in the middle part of the interface. After finishing 5 questions, the workers would be redirected to a completion page which showed them the amount of bonus they could earn and the completion code.

[4] We tried only accepting the answers that were exactly correct, but this greatly hurt the performance of the workers as providing the exact number of nouns in a long word list can sometimes be very difficult.

[5] 3 evaluation mechanisms X 3 bonus levels.

[6] It showed the amount of bonus: $0.01, $0.05, or $0.10

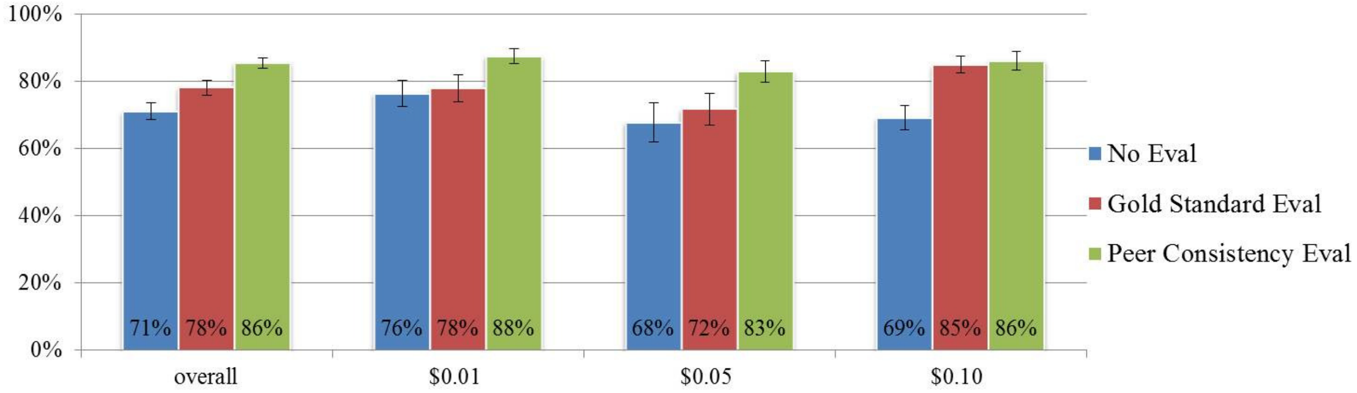


Figure 3. The mean of the average label accuracies of the workers under different evaluation mechanisms and bonus levels (with standard error). In general, the workers under peer consistency evaluation outperformed the ones under gold standard evaluation and without evaluation.

	N vs. GS	N vs. PC	GS vs. PC
overall	71%(24.7%) vs. 78%(21.3%)* (t(174) = 2.04, p < 0.05)	71%(24.7%) vs. 86%(14.6%)* (t(144) = 4.75, p < 0.001)	78%(21.3%) vs. 86%(14.6%)* (t(158) = 2.62, p < 0.005)
0.01	76%(21.5%) vs. 78%(21.7%) (t(58) = 0.26, p = 0.79)	76%(21.5%) vs. 88%(11.8%)* (t(45) = 2.47, p < 0.01)	78%(21.7%) vs. 88%(11.8%)* (t(45) = 2.13, p < 0.05)
0.05	68%(31.6%) vs. 72%(25.5%) (t(56) = 0.53, p = 0.60)	68%(31.6%) vs. 83%(16.8%)* (t(44) = 2.32, p < 0.05)	72%(25.5%) vs. 83%(16.8%)* (t(50) = 2.01, p < 0.05)
0.10	69%(19.6%) vs. 85%(13.5%)* (t(52) = 3.64, p < 0.001)	69%(19.6%) vs. 86%(15.1%)* (t(55) = 3.77, p < 0.001)	85%(13.5%) vs. 86%(15.1%) (t(57) = 0.33, p = 0.74)

Table 1. The comparison between the mean (standard deviation) of the average label accuracy of different evaluation mechanism at various bonus levels and their t-test statistics. The significant results (p-value < 0.05) are marked with *.

RESULTS

270 subjects (30 subjects per treatment) were recruited from AMT, each of them completed 5 noun counting task and generated 5 labels, so 1,350 labels (150 labels per treatment) were collected for our result analysis.

To evaluate the results of our experiment, we first computed the label accuracy of each collected label. The label accuracy shows how close the label is to the gold standard answer. Therefore, the formula for label accuracy is:

$$Label\ Accuracy = 1 - \frac{|N_{GS} - N_{worker}|}{Max(30 - N_{GS}, N_{GS})}$$

Where N_{GS} is the correct number of nouns in the list, N_{worker} is the number of nouns in the list reported by the worker. Therefore, the numerator of the second term in the formula is the distance between the label generated by the worker and the gold standard answer, and the denominator of it is the distance between the worst possible answer and the gold standard answer. (If $N_{GS} > 15$, the worst possible answer is 0. Otherwise, it is 30.) Therefore, the label accuracy is a number between 1 and 0, where 1 represents that the label is exactly the same as the gold standard, and 0 means that the answer is the worst possible answer. Then the average label accuracy of each worker was computed by averaging the label accuracy of the five labels generated by the worker. We used the average label accuracy of the workers to measure their quality of works.

We conducted a 3X3 Repeated Measure ANOVA to see if there were significant effects of the evaluation mechanisms, bonus levels, and their interactions. The results showed that the evaluation mechanisms did significantly affect the average label accuracy of the workers ($F(2, 261) = 11.08,\ p < 0.001$). Moreover, the effect of bonus levels was marginally significant ($F(2, 261) = 2.72,\ p = 0.067$). However, the effect of the interactions of the two factors were not significant ($F(4, 261) = 1.15,\ p = 0.34$). Because the analysis shows that the effects of bonus levels is only marginal to significant and the interaction between evaluation method and bonus levels was not significant, in the current result analysis, we focus on the effects of the evaluation mechanisms.

The effects of the evaluation mechanisms

The mean of the average label accuracies for the workers with different evaluation mechanisms and bonus levels are summarized in Figure 3 and the comparisons between different conditions are summarized in Table 1. The results show that the average label accuracy of the workers in the systems with peer consistency evaluation was significantly higher than the systems with no evaluation. In addition, there was also a significant difference between the performance of the workers in the systems with gold standard evaluation and the systems with no evaluation. Interestingly, *the workers in the systems with peer consistency evaluation also outperformed the workers in the systems with gold standard evaluation*.

When we further analyzed the results of the systems with different bonus levels, we observed that with low or medium bonuses ($0.01 or $0.05), the workers under peer consistency evaluation performed significantly better than the workers under no evaluation and the workers under gold standard evaluation. In addition, there was no significant difference between the workers assessed via no evaluation and gold standard evaluation.

However, when we increased the bonus to the highest level ($0.10), there was nearly no difference between the workers under peer consistency evaluation and the ones under gold standard evaluation, and the difference between the workers under gold standard evaluation and the ones under no evaluation became significant.

These differences, of course, need to be interpreted cautiously, as the overall interaction effect between bonus and evaluation method was not significant. Nevertheless, we can see that the difference between the ability to enhance reliability between the two evaluation methods is larger when the workers received low or medium bonuses. While the main effects of evaluation methods and bonus levels are strong, future studies can further test their possible interaction effect by introducing more levels of bonus in different evaluation methods.

Peer consistency as valid evaluation mechanism

Previous research [9] on peer consistency evaluation in output agreement games provided a game theoretic analysis for why it could generate higher quality works in a gaming environment. The analysis showed that because both players in the game would like to earn points, the player chooses the answer that is most likely to be the same as the answer generated by the other player. Thus, the best strategies for both players are putting more effort to select the correct answer for the task (or the answers that are close to it). The reason is that since the players cannot communicate with each other during the game, high-accuracy answers can serve as a protocol for the two players. By choosing the answers that are close to the gold standard answer, it increases the probability for them to generate similar answers and earn points.

Similar analysis also applies to our current study; in our human computation systems with peer consistency evaluation, workers did not know which previous worker would be chosen in the evaluation. As a result, there was no way for them to communicate with each other or agree on some particular answers in advance. Therefore, putting more effort to provide answers with high accuracy would be generally a desirable strategy for workers, which could be a factor that enhanced the overall reliability of the system. This analysis explains why the system with peer consistency evaluation had a higher reliability compared to the systems without evaluation in our experiment, which supports **H1** that *peer consistency evaluation can serve as a valid evaluation mechanism to motivate workers to generate high-quality work.*

Reciprocal altruism facilitating worker performance

A notable difference between peer consistency evaluation and gold standard evaluation is that because the system with peer consistency evaluation used the answers of previous workers as evaluation standards, the performance of the workers not only affected their own bonus, but also the bonus of the future workers of the system. If workers did not put enough effort into selecting a high-quality answer, not only would they lose the chance to earn the bonus, future workers evaluated by this erroneous answer might also not get the bonus. Therefore, workers might have an altruistic motive to put more effort into the task in order to maximize the bonus-earning potential of their "colleagues". (**H2**)

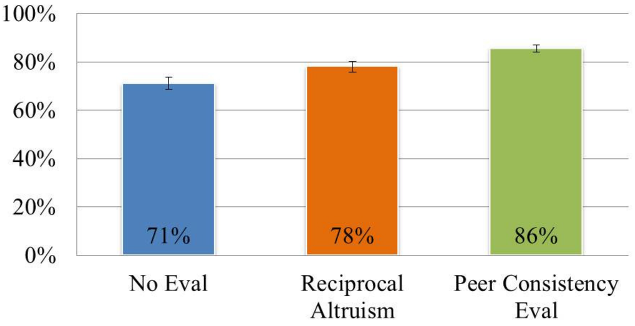


Figure 4. The mean of the average label accuracy of the system with reciprocal altruism compares to the system with no evaluation and peer consistency evaluation.

To test if this hypothesis is true, we created another human computation system with a condition we called *reciprocal altruism (RA)* and recruited 90 workers[7] to conduct the same noun counting task on this system. The workers of the system in the RA condition were not evaluated, but they were told that their answers would be used as evaluation standards to evaluate future workers. In other words, the condition was similar to the peer consistency evaluation condition, *except* that workers were told that their answers would not be evaluated. The exact words we put on the interface were as follows:

- **Reciprocal altruism (RA)**: You can earn bonus for providing the answer. We will use your answer as an evaluation standard for future workers. They can only earn bonus if their answers are similar to yours.

The mean of the average label accuracy of the system with reciprocal altruism is shown in Figure 4. The results showed that even without any evaluation, the workers with reciprocal altruism ($\mu = 78\%,\ SD = 20.8\%$) performed significantly better than those without reciprocal altruism ($\mu = 71\%,\ SD = 24.8\%$) ($t(173) = 2.02,\ p < 0.05$). This provides support to **H2** that the *altruistic motive did make the workers provide more reliable answers*. The other thing we found was that workers under peer consistency evaluation still performed significantly better than those in the RA condition ($t(160) = 2.80,\ p < 0.001$). One reasonable explanation is that both reciprocal altruism (that their answers were used to evaluate future workers) and evaluation feedback (that their answers were evaluated using previous workers answers) are factors that provide additive effects that made workers perform better. Peer consistency evaluation allows its system to benefit from the additional motivating effects of both factors.

Unfairness as insignificant factor in worker performance

[7] 30 workers each bonus level

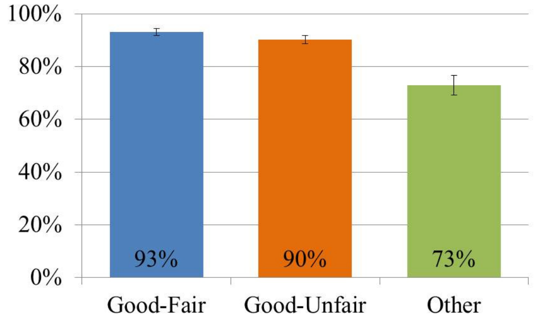


Figure 5. The comparison between the performance of the good workers were treated fairly, unfairly, and other workers under peer consistency evaluation.

It is reasonable to assume that the workers under gold standard evaluation had higher incentives to work harder because peer consistency evaluation sometimes penalized workers even if they put more effort to generate correct answers. (**H3**) Our results, however, clearly showed the opposite effect: workers in peer consistency evaluation conditions outperformed those in gold standard evaluation conditions.

To understand this seemingly counter-intuitive result, we analyzed whether the unfairness of peer consistency evaluation hurt the performance of workers. We divided the 5 questions answered by each worker into two groups, and used the first half (first two questions) to identify those that were *good workers* in peer evaluation condition. Specifically, workers were considered good workers if they generated answers that were similar to the gold standard answers (distance less than 3) in the first 2 questions. These *good workers* should have earned the bonuses if the system evaluated their answers using gold standard answers. 62 out of 90 workers in the systems with peer consistency evaluation were considered good workers by this definition. These good workers were further divided into two groups: the first group was treated fairly, in the sense that they actually earned the bonus because the answers used to evaluate them were also close to the gold standard answers. In contrast, the second group of them were treated unfairly, in the sense that they failed to pass peer consistency evaluation for at least one of the first two questions because they were evaluated by answers that were different from the gold standard answers. 33 good workers were treated fairly, and the other 29 good workers were treated unfairly by the system with peer consistency evaluation in the first two questions. By comparing how these two groups of good workers performed in the next three questions, we could test the extent to which the fairness of evaluation could impact performance.

The mean of the average label accuracies on the next 3 questions (the second half of the 5 questions) of these two groups and other workers that did not do well in the first 2 questions are shown in Figure 5. Although the good workers who were treated fairly ($\mu = 93\%,\ SD = 1.2\%$) performed slightly better than those who got penalized incorrectly ($\mu = 90\%,\ SD = 1.5\%$), the difference of the performance between these two groups was not significant ($t(55) = 1.45,\ p = 0.15$). Moreover, the workers who were treated unfairly still performed significantly better than the other workers that did not perform well in the first two questions ($\mu = 73\%,\ SD = 20.5\%$) ($t(39) = 4.22,\ p < 0.001$). This is in accordance with the previous research that there is a very low correlation between the performance of the workers and the quality of evaluation standard [9]. Therefore, at least in our samples of workers, *we did not find that unfairness of evaluation in peer consistency evaluation could discourage good performance*. This disproves **H3**.

Implicit social interaction as possible factor in enhancing human computation reliability

Research has shown that social interaction can attract workers to voluntarily conduct tasks for human computation systems [9]. In our experiment, when the workers could earn high levels of bonus, both peer consistency evaluation and gold standard evaluation could enhance the reliability of the systems. However, when the workers only could earn low or medium bonuses, peer consistency evaluation seemed more effective. It is possible that peer consistency evaluation produced a feeling among the workers that they were not doing the task alone because they knew that their answers were to be evaluated by other workers in the systems, and that their answers also would be used to evaluate other workers. This implicit social interaction created a social motivation to encourage them to put extra effort into solving the task. This could explain why the systems with peer consistency evaluation outperformed the systems with gold standard evaluation, especially when workers only receive a minimum level of bonus. In future studies, it will be useful to conduct a series of carefully controlled experiments that focus on how the implicit social interactions created by peer consistency evaluation affect workers' performance.

Possible effects of financial incentive levels

Although the ANOVA results show that the effect of bonus level was only marginally significant, we found weak evidence of a possible trend in our results that might provide useful insights for future studies. Specifically, we found that the means of the average label accuracies of the workers dropped from 81% to 74% when we increased the bonus from $0.01 to $0.05, and this number climbed back to 80% when we further raised the bonus to $0.10. This is in accordance with the findings from previous research [5] that financial incentives not only bring the utility of monetary reward to the workers, they also reduce the intrinsic values (e.g., enjoyment, altruism) for the workers to perform the tasks voluntarily. We believe that this potentially interesting interaction between financial incentives and social motivation will be an important topic for human computation research.

CONCLUSIONS AND FUTURE WORK

In this paper, we show that peer consistency evaluation is effective in enhancing the reliability of human computation systems. Even though the system does not evaluate its workers by gold standard answers, the mechanism is effective in motivating workers to put in their effort to generate answers that are close to the gold standard answers. In fact, we found that although peer consistency evaluation sometimes incorrectly penalizes the workers, the potential unfairness in evaluation does not seem to have a large effect that significantly harms the performance of good workers. Moreover, we show that reciprocal altruism (i.e., preventing future workers from being penalized incorrectly) may even make peer consistency evaluation a better mechanism than traditional gold standard evaluation. Since gold standard answers are costly to generate, this result provides researchers and practitioners in human computation (including crowdsourcing) a much more cost-effective option to ensure system reliability.

In addition, we found that the implicit social interaction of peer consistency evaluation might be another factor that helps to improve reliability. In our experiment, the workers under peer consistency evaluation performed well even when the bonus they could earn was low. This showed that implicit social interaction might provide some form of a social motivation to workers, even when the financial incentive was low. This points to the need to conduct more systematic studies on how social interactions of the workers in a human computation system affect different forms (social, finanicial, or others) of incentives and performance. One possible way is, for example, to test if the workers under peer consistency evaluation are willing to do more tasks than those without it. This could show whether the implicit social interaction really motivates the workers to do the tasks voluntarily. Another possibility is to strengthen the social interactions (e.g., reducing anonymity, message boards) to test how it affects the reliability of the system.

We believe that more studies are needed to test the limitation of peer consistency evaluation by applying it to different kinds of tasks. It is possible, for example, that certain tasks may have a wider range of variability of performance, or certain systematic biases may be observed in workers' responses. In these situations, peer consistency evaluation may either lead to worse performance because of the general perception that answers from a previous worker are more likely to be wrong than correct; or that certain biases may be magnified as workers start to perceive that previous workers may likely generate wrong answers (i.e., future workers adapt to bad answers rather than being encouraged to provide good ones). While these are some of the many possible directions that are worthy of pursuing, we believe that results from the current study provides a small but significant step towards better understanding of the complexity involved in a human computation system that are useful for the CSCW community.

REFERENCES

1. Ahn, L. V., Blum, M., Hopper, N. J., and Langford, J. Captcha: using hard ai problems for security. In *Proceedings of the 22nd international conference on Theory and applications of cryptographic techniques*, EUROCRYPT'03, Springer-Verlag (Berlin, Heidelberg, 2003), 294–311.
2. Bandiera, O., Barankay, I., and Rasul, I. Social preferences and the response to incentives: Evidence from personnel data. *The Quarterly Journal of Economics 120*, 3 (2005), 917–962.
3. Dawid, A. P., and Skene, A. M. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics) 28*, 1 (1979), pp. 20–28.
4. Dow, S., Kulkarni, A., Klemmer, S., and Hartmann, B. Shepherding the crowd yields better work. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, CSCW '12, ACM (New York, NY, USA, 2012), 1013–1022.
5. Gneezy, U., and Rustichini, A. Pay enough or don't pay at all. *The Quarterly Journal of Economics 115*, 3 (2000), 791–810.
6. Harris, C. G. You're hired! an examination of crowdsourcing incentive models in human resourse tasks. In *Proceedings of WSDM 2011 Workshop on Crowdsourcing for Search and Data Mining* (2011).
7. Hirth, M., Hossfeld, T., and Tran-Gia, P. Cost-optimal validation mechanisms and cheat-detection for crowdsourcing platforms. In *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on* (30 2011-july 2 2011), 316 –321.
8. Huang, E., Zhang, H., Parkes, D. C., Gajos, K. Z., and Chen, Y. Toward automatic task design: a progress report. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '10, ACM (New York, NY, USA, 2010), 77–85.
9. Huang, S.-W., and Fu, W.-T. Systematic analysis of output agreement games: Effects of gaming environment, social interaction, and feedback. In *Proceedings of HCOMP12: The 4th Workshop on Human Computation* (2012).
10. Ipeirotis, P. G. Analyzing the amazon mechanical turk marketplace. *XRDS 17*, 2 (Dec. 2010), 16–21.
11. Ipeirotis, P. G., Provost, F., and Wang, J. Quality management on amazon mechanical turk. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '10, ACM (New York, NY, USA, 2010), 64–67.
12. Kittur, A., Khamkar, S., André, P., and Kraut, R. Crowdweaver: visually managing complex crowd work. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, CSCW '12, ACM (New York, NY, USA, 2012), 1033–1036.
13. Kittur, A., Smus, B., Khamkar, S., and Kraut, R. E. Crowdforge: crowdsourcing complex work. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, UIST '11, ACM (New York, NY, USA, 2011), 43–52.
14. Kulkarni, A., Can, M., and Hartmann, B. Collaboratively crowdsourcing workflows with turkomatic. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, CSCW '12, ACM (New York, NY, USA, 2012), 1003–1012.

15. Law, E., and Von Ahn, L. *Human Computation*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2011.

16. Le, J., Edmonds, A., Hester, V., and Biewald, L. Ensuring quality in crowdsourced search relevance evaluation: The effects of training qustion distribution. In *Proceedings of the SIGIR 2010 Workshop on Crowdsourcing for Search Evaluation* (2010).

17. Liem, B., Zhang, H., and Chen, Y. An Iterative Dual Pathway Structure for Speech-to-Text Transcription. In *Proceedings of the AAAI Workshop on Human Computation (HCOMP)* (2011).

18. Lin, C., Mausam, M., and Weld, D. Crowdsourcing control: Moving beyond multiple choice. In *Proceedings of HCOMP12: The 4th Workshop on Human Computation* (2012).

19. Lin, C. H., Mausam, and Weld, D. S. Dynamically switching between synergistic workflows for crowdsourcing. In *AAAI* (2012).

20. Little, G., Chilton, L. B., Goldman, M., and Miller, R. C. Turkit: tools for iterative tasks on mechanical turk. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '09, ACM (New York, NY, USA, 2009), 29–30.

21. Mason, W., and Watts, D. J. Financial incentives and the "performance of crowds". *SIGKDD Explor. Newsl. 11*, 2 (May 2010), 100–108.

22. Oleson, D., Sorokin, A., Laughlin, G., Hester, V., Le, J., and Biewald, L. Programmatic gold: Targeted and scalable quality assurance in crowdsourcing. In *Proceedings of HCOMP11: The 3rd Workshop on Human Computation* (2011).

23. Paritosh, P. Human computation must be reproducible. In *Proceedings of CrowdSearch: Crowdsourcing Web search 2012* (2012).

24. Quinn, A. J., and Bederson, B. B. Human computation: a survey and taxonomy of a growing field. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, CHI '11, ACM (New York, NY, USA, 2011), 1403–1412.

25. Robertson, S., Vojnovic, M., and Weber, I. Rethinking the esp game. In *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, CHI EA '09, ACM (New York, NY, USA, 2009), 3937–3942.

26. Seemakurty, N., Chu, J., von Ahn, L., and Tomasic, A. Word sense disambiguation via human computation. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '10, ACM (New York, NY, USA, 2010), 60–63.

27. Snow, R., O'Connor, B., Jurafsky, D., and Ng, A. Y. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, Association for Computational Linguistics (Stroudsburg, PA, USA, 2008), 254–263.

28. Sun, Y.-A., Roy, S., and Little, G. D. Beyond independent agreement: A tournament selection approach for quality assurance of human computation tasks. In *Proceedings of HCOMP11: The 3rd Workshop on Human Computation* (2011).

29. von Ahn, L., and Dabbish, L. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '04, ACM (New York, NY, USA, 2004), 319–326.

30. von Ahn, L., and Dabbish, L. Designing games with a purpose. *Commun. ACM 51* (Aug. 2008), 58–67.