Proceedings of the Fifth Conference on
Human Computation and Crowdsourcing
(HCOMP 2017)

Deja Vu: Characterizing Worker Reliability Using Task Consistency

**Alex C. Williams,[1] Joslin Goh,[2] Charlie G. Willis,[3] Aaron M. Ellison,[4]
James H. Brusuelas,[5] Charles C. Davis,[3] Edith Law[1]**
[1]David R. Cheriton School of Computer Science, University of Waterloo
[2]Department of Statistics and Actuarial Science, University of Waterloo
[3]Department of Organismic and Evolutionary Biology, [4]Harvard Forest, Harvard University
[4]Faculty of Classics, University of Oxford
{alex.williams,joslin.goh,edith.law}@uwaterloo.ca, {charleswillis,aellison,cdavis}@fas.harvard.edu,
james.brusuelas@classics.ox.ac.uk

Abstract

Consistency is a practical metric that evaluates an instrument's reliability based on its ability to yield the same output when repeatedly given a particular input. Despite its broad usage, little is understood about the feasibility of using consistency as a measure of worker reliability in crowdwork. In this paper, we explore the viability of measuring a worker's reliability by their ability to conform to themselves. We introduce and describe *Deja Vu*, a mechanism for dynamically generating task queues with consistency probes to measure the consistency of workers who repeat the same task twice. We present a study that utilizes *Deja Vu* to examine how generic characteristics of the duplicate task — such as placement, difficulty, and transformation — affect a workers task consistency in the context of two unique object detection tasks. Our findings provide insight into the design and use of consistency-based reliability metrics.

Introduction

Quality control is a common and important challenge for crowdsourced datasets. Due to their natural susceptibility to workers performing a task incorrectly by accident or with intent, a key objective for crowdsourcing systems is identifying reliable workers. From simple majority-vote approaches to sophisticated machine-learning based models, a broad range of techniques have been developed to manage the quality of crowdsourcing data, yet the topic has remained at the forefront of concerns for both practitioners and researchers of crowdsourcing alike (Ipeirotis, Provost, and Wang 2010; Jung and Lease 2011). This raises a philosophical question—what defines a "good" worker when no objective measure of quality exists, and how do we leverage alternative measures of worker quality to improve crowdsourcing results?

Consistency is a measure of reliability in many domains, including healthcare (Chinn 1991), genomics (Misztal, Legarra, and Aguilar 2009), chemistry (Margolis and Duewer 1996), pervasive computing (Henricksen, Indulska, and Rakotonirainy 2002), machine learning (Rosten, Porter, and Drummond 2010), and human-computer interaction (Wilson et al. 2011; Hornbæk et al. 2014)). It evaluates an instrument's reliability based on its ability to yield the same output when repeatedly given a particular input under the same constraints. Deterministic algorithms, for example, can be described as *consistent* as they always produce the same output when given the same input. Despite its widespread application, only a handful of crowdsourcing literature has examined the utility of consistency (Cheng, Teevan, and Bernstein 2015; Sun and Stolee 2016) as a reliability metric, leaving many important questions unanswered: To what extent are workers capable of performing tasks consistently? How do characteristics of a repeated task affect a worker's consistency? Are consistency-based quality-control procedures viable alternatives to traditional methods of quality control?

In this paper, we explore the viability of measuring workers' reliability by their ability to generate the same response for a pair of duplicate tasks, which we call a *consistency probe*. First, we introduce *Deja Vu*, a mechanism for generating task queues with consistency probes to measure the task consistency of workers. Next, we present and report findings from an experiment to examine how certain characteristics of the consistency probe — such as placement, difficulty, and transformation — affect a workers task consistency. The experiment is conducted in the context of two object counting tasks that ask workers to locate a particular type of object in a set of ten images. We conclude with a discussion on the practicality of our findings and directions for future work.

Related Work

Characterization of Worker Reliability

Consensus Consensus-based reliability metrics are among the most common strategies for measuring worker reliability. These measures are often driven by comparing worker answers to consensus (*e.g.*, majority vote), which assumes the workers are equally reliable (Sheng, Provost, and Ipeirotis 2008; Sheshadri and Lease 2013). One common practice here is to score and filter workers by the proximity of their answer to the consensus (Ribeiro et al. 2011). Expectation Maximization (EM) algorithms go beyond the naive assumption of a perfect crowd and assume that workers have unknown errors that can be estimated simultaneously with ground truth (Dawid and Skene 1979; Demartini, Difallah, and Cudré-Mauroux 2012; Ipeirotis, Provost, and Wang 2010; Raykar and Yu 2012; Snow et al. 2008; Whitehill et al. 2009). Prior work offers alternative ways to weigh worker responses, including the use of Z-score from information retrieval (Jung and Lease 2011) to more advanced Bayesian approaches that model other worker and task characteristics, such as difficulty, approach to annotation, and expertise (Welinder et al. 2010).

Behavioral Measures of Reliability Behavioral measures capturing *how* workers perform tasks (*e.g.*, task fingerprinting) consisting of cognitive and motor actions, have been shown to approximate task performance and reliability (Rzeszotarski and Kittur 2011). Interactions with interface components critical to the task at hand have also been used to measure worker reliability (Buchholz and Latorre 2011). Such metrics have also been used to identify *curbstoning* (*i.e.*, falsification of survey data) (Birnbaum et al. 2013).

Consistency as as Measure of Reliabiliy Several prior works in crowdsourcing have explored consistency—the ability of a worker to conform to themselves when performing a task—as a measure of reliability. One study (Cheng, Teevan, and Bernstein 2015) evaluated the consistency of 40 workers performing a set of tasks, including emotion mapping and image categorization, and found that consistency between timed and untimed task variations could be a viable substitute for ground-truth data in objective tasks. In the context of online surveys, it was reported that 30% of the workers, when given the exact same survey twice, submitted inconsistent responses (Sun and Stolee 2016). Finally, Hata et al. found that workers can maintain consistent answer quality over long periods of time (Hata et al. 2017).

Our work is distinct from prior work in that we study the effects of characteristics of duplicate tasks and how they affect the task consistency of workers. We are not aware of prior work that has focused on either topic.

Effects of Task Characteristics on Reliability

Prior work has shown that various characteristics of a task sequence can affect the way workers perform tasks including task difficulty (Mao et al. 2013), contextual factors (Teevan, Iqbal, and von Veh 2016), and task ordering (Cai, Iqbal, and Teevan 2016). Other works (Chandler and Kapelner 2013; Newell and Ruths 2016) have shown that workers output can be strongly influenced by how the task is framed, either through an explicit message or by manipulating the content of preceding tasks. Closest to our work, Cheng et al. compared the error-time curves of workers performing a set of primitive tasks under two characterizations of quality, namely internal consistency and between-subject variation (Cheng, Teevan, and Bernstein 2015).

Deja Vu

Deja Vu is a mechanism for distributing calculated consistency probes and yielding a consistency-based reliability metric for a worker, composed of two components. The first component is a task router that distributes tasks to workers. The second component is a metric for assessing the quality of workers based on the consistency of their output for duplicates. We discuss each of these components along with the rationale behind their design in detail below.

Task Router

The Task Router component automatically constructs a queue of tasks with consistency probes to capture the task consistency of a worker. Formally, a *consistency probe* is defined as a task-set containing an original task and its corresponding D duplicates. For the purpose of this study, we only consider the simplest scenario in which there is only one duplicate (*i.e.*, D=1) following the original task; in practice, the original task can be followed by multiple duplicates to accommodate more complex scenarios. Both the size of the queue and the number of consistency probes can be specified by the managing requester.

Routing Dimensions

There are three dimensions of a consistency probe that can be configured: placement, transformation, and difficulty.

Placement: The task router can systematically select *where* the original task and its duplicates appear in the task queue. As shown in Figure 1, placement can be specified using two parameters:

- $position_{orig}$: the position of the original task
- $offset_{orig,dup}$: the number of tasks between the original and duplicate task

Transformation: The task router can apply a transformation to a duplicate task (*i.e.*, flipping a image on the Y-axis).

- *transform*: the transformation applied to the duplicate task

Difficulty: The task router can, optionally, adjust the difficulty of consistency probes by tuning three parameters:

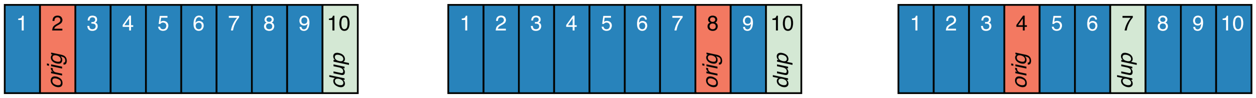
- $difficulty_{dup}$: the difficulty of the duplicate task
- $difficulty_{< orig}$: the average difficulty of the tasks before the original task
- $difficulty_{orig,dup}$: the average difficulty of the tasks between the original task and the duplicate task

Each of the ***Difficulty*** parameters is bound by the availability of information that can be used as proxies for task difficulty. For example, the difficulty of object counting tasks can be approximated by the number of objects in the image. In many cases, this information is initially unknown in the context of crowdsourcing, and the parameters are therefore not required for routing consistency probes.

As these parameters are the most basic dimensions that describe tasks and how they are served to workers, the *Deja Vu* task routing mechanism is task-agnostic and applicable to any task routing scenario.

A Baseline Measure of Consistency

The second component of *Deja Vu* is a measure of consistency. The simplest such measure is the absolute difference



(a)
$position_{orig} = 2$; $offset_{orig,dup} = 7$

(b)
$position_{orig} = 8$; $offset_{orig,dup} = 1$

(c)
$position_{orig} = 4$; $offset_{orig,dup} = 2$

Figure 1: Task queues with varying `Placement` parameters.

between a worker's outputs for the original task and its duplicate:

$$output_{orig,dup} = |output_{orig} - output_{dup}| \quad (1)$$

where $output_{orig}$ is the output for the original task and $output_{dup}$ is the output for the duplicate. If $output_{orig,dup}$ equals zero, the worker is perfectly consistent while a nonzero value indicates the worker is observably inconsistent.

Study Design

We conducted an experiment on Amazon Mechanical Turk[1] to examine the ability of workers to yield consistent output for a particular task. In this study, we vary the key parameters for placement and transformation to determine how each parameter affects the task consistency of workers, as measured by our baseline metric. The effects of difficulty are examined post-hoc as task queues were generated randomly to minic real-world crowdsourcing scenarios.

Task and Procedure

Object detection is among the most common types of commercial and scientific crowdsourced tasks (Ipeirotis 2010; Simpson, Page, and De Roure 2014), and recent work has reinforced the importance of people in reliable methods for object detection in images (Sarma et al. 2015) and computer vision research (Forsyth and Ponce 2002). In this work, we focus specifically on counting tasks, where the input is an artifact (*e.g.*, an image) and the output is the count of a certain object found in the artifact. While participants do annotate objects in an image, the output of the task is limited to the numeric count of an object as determining which object was counted by comparing coordinates is challenging, particularly when objects occlude one another (Sarma et al. 2015).

Our study focuses on counting tasks in two unique domains: (1) counting flowers in images of Herbarium records and (2) counting Greek taus in images of ancient papyrus manuscripts. We refer to these tasks as the Flower task and the Tau task respectively.

Figure 2 illustrates the counting interface for both tasks. In the interface, users can locate and count objects in the image by clicking on the image to make an annotation. As each annotation is created or removed, the interface automatically increments or decrements the count for the object next to the object's label to the right of the image. For the purpose of this study, we assume that each image contained at least one identifiable object for the task and structured the the interface to prevent workers from submitting a task with a reported count of zero. Both the annotation interface and the *Deja Vu* mechanism are implemented within the CrowdCurio research-oriented crowdsourcing platform[2] (Law et al. 2013; Willis et al. 2017).

In order to quantify the difficulty of each image and study participant accuracy, ground-truth counts for each task were collected from experts or public datasets. For the Flower task, four specialists with a background in biology, who are currently employed at the herbarium of an R1 research institution, were recruited to locate the flowers in each herbarium records. The median count of the recruited specialists was taken as the ground-truth. For the Tau task, ground-truth counts for each papyrus manuscript were retrieved from published, peer-reviewed transcriptions (Society 1908).

For each task, participants were asked to report counts for a series of 10 images. A dataset of 30 randomly-selected images with varying ground-truth counts, ranging from 2 to 88 objects in a single image, was used to generate task queues for workers in each task. Task queues were generated by randomly selecting 8 tasks from the dataset of 30 images. An additional image was randomly selected from the remaining 22 images as a consistency probe and subsequently inserted into the task queue at two particular locations. If specified, a transformation was applied to the second instance of the task selected as the probe.

All participants were recruited from Amazon Mechanical Turk and paid $2.00 for completing the task. Before beginning the task, workers were required to watch a training video explaining how to use the interface to correctly perform the task. Additionally, workers were asked to complete a pre-questionnaire that indicated experience relevant to the task (*i.e.*, familiarity with plant sciences or the Greek language). The experiment concluded with a post-questionnaire that first asked them if they realized they were given a duplicate image and to identify the duplicate if they believed they had seen one. The post-questionnaire also included included questions from the Intrinsic Motivation Inventory (Ryan 1982) to assess workers' enjoyment, effort, and competence for the task.

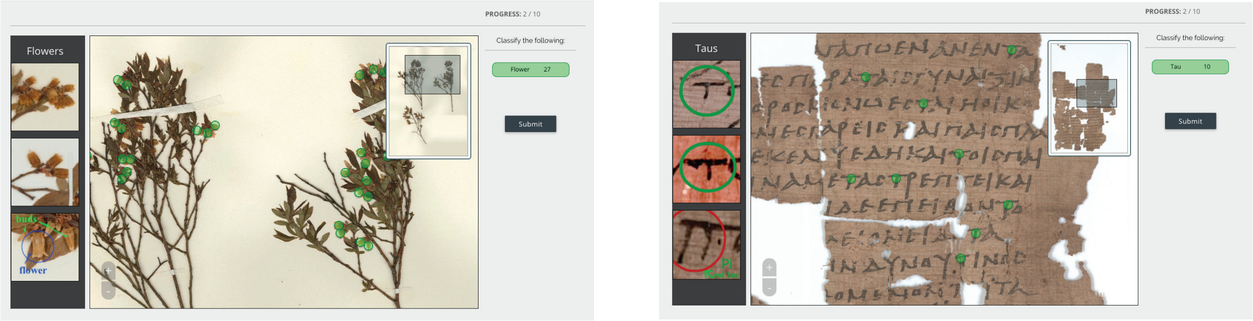
Experimental Conditions

To investigate the effects of placement and transformation on worker consistency, we created the following set of conditions for each task, totaling in 8 conditions:

Low Offset, No Transformation Original and duplicate task are separated by 1 task. No transformation applied to the duplicate task.

[1] https://www.mturk.com

[2] https://www.crowdcurio.com



(a) Counting flowers.

(b) Counting Greek taus.

Figure 2: Annotation interface for counting objects.

Low Offset, Duplicate Flipped Original and duplicate task are separated by 1 task. The duplicate is flipped on the Y-axis.

High Offset, No Transformation Original and duplicate task are separated by 6 tasks. No transformation applied to the duplicate task.

High Offset, Duplicate Flipped Original and duplicate task are separated by 6 tasks. The duplicate is flipped on the Y-axis.

In all conditions, the duplicate tasks and the remaining 8 images in the task queue were chosen at random. While the placement of the duplicate was controlled in each condition, the arrangement of the 8 non-duplicate tasks was randomly generated to eliminate ordering effects. In all conditions, $position_{orig}$ was assigned a value of 2, meaning the first instance of the consistency probe occurred directly after the first task. Therefore, our examination on the effects of placement focuses entirely on $offset_{orig,dup}$. However, identifying an optimal $position_{orig}$ is a key direction of future work. As the effects of placement and transformation are unknown, we regard none of the 8 conditions as a baseline and focus on reporting observable differences between the conditions.

Research Questions and Hypotheses

Our study aims to answer four research questions (Q1-Q4).

`Q1`: What is the relationship between consistency and the placement of the duplicate? First, we hypothesized that consistency improves as number of tasks between the original task and duplicate task increases:

[H1] $output_{orig,dup}$ increases as $offset_{orig,dup}$ increases.

`Q2`: What is the relationship between consistency and difficulty of the repeated task in the consistency probe? We hypothesized that consistency decreases as difficulty increases:

[H2] $output_{orig,dup}$ increases as $difficulty_{dup}$ increases.

`Q3`: What is the relationship between consistency and applying a transformation to the duplicate task? We hypothesized that consistency increases when a transformation (*i.e.*, flipped on the Y-axis) is applied to the duplicate task:

[H3] $output_{orig,dup}$ decreases when *transform* is applied.

Workers may, for example, accredit the presence of a duplicate task to a systematic error, and may operate under the assumption their data has already been collected for the duplicate task. A transformation may disguise the duplicate, overcoming biases associated with performing a task twice.

`Q4`: What is the relationship between worker consistency and recognition of the consistency probe? We hypothesized that consistency decreases when the consistency probe is recognized:

[H4] $output_{orig,dup}$ increases when recognition occurs.

The intuition behind this hypothesis is that workers may remember prior answers. Alternatively, workers may recognize the task as a retest for reliability and strategically game the system.

Analysis Methods

In our study, the output of each task is the number of detections (*i.e.*, counts) for flowers or taus reported by workers. We used the number of detectable objects in the image as a proxy for difficulty, and consider two scenarios: *S1*, when actual difficulty of each task is known, *i.e.*, can be determined based on expert (median) consensus count, and *S2* the realistic scenario when the task difficulty is unknown, but can be estimated based on worker consensus count.

Dependent Variables: In both *S1* and *S2*, our simple consistency metric $output_{orig,dup}$ is the dependent variable.

Independent Variables: In *S1*, the independent variables consist of the following: the number of tasks between the original task and duplicate task ($offset_{orig,dup}$), a binary variable representing whether the transformation was applied to the image (*transform*), a binary variable representing whether the worker noticed the duplicate task (*Noticed*), a binary variable represent whether the worker was able to correctly recall and identify the duplicate task (*Identify*), the difficulty of the duplicate task ($difficulty_{dup}$), the average difficulty of the tasks before the duplicate ($difficulty_{< orig}$), and the difficulty of the tasks between the original and duplicate task ($difficulty_{orig,dup}$). In *S2*, the independent variables are identical, but exclude the difficulty-related parameters as the assumption is that ground-truth information is unavailable. In both scenarios, we also considered two interaction terms of interest: *transform* × *Noticed* and *transform* × *Identify*.

Statistical Methods: The dependent variable, $output_{orig,dup}$, is an integer, which is typically modeled using a log-linear (Poisson) regression model. However, as illustrated by Figure 3, we observed that the distribution of $output_{orig,dup}$ is skewed toward zero, indicating that most workers are capable of being perfectly consistent (*i.e.*, $output_{orig,dup} = 0$). The excess observed zero counts could cause ordinary Poisson models to poorly fit the data, and a zero-inflated Poisson (ZIP) model would be more suitable (Lambert 1992). Validating these concerns, Table 1 shows the predictive performance of both types of models.

To account for the zero inflation caused by $output_{orig,dup}$, the proposed ZIP models for both *S1* and *S2* specify perfect consistency as a binary random variable that depends on $offset_{orig,dup}$, whereas $output_{orig,dup}$ follows a Poisson distribution whose mean depends on all the independent variables as mentioned earlier. Each ZIP model used a logistic model to describe the probability of perfect consistency, and a Poisson regression to model the non-zero counts. Under the Vuong Non-Nested hypothesis test (Vuong 1989), all ZIP models are shown to be significantly better than the corresponding ordinary Poisson regression model. The final ZIP models were selected through the likelihood ratio test (Wasserman 2003), which compares the models with and without the interaction terms. Residual plots were checked to ensure there was no clear violation of the model's assumptions.

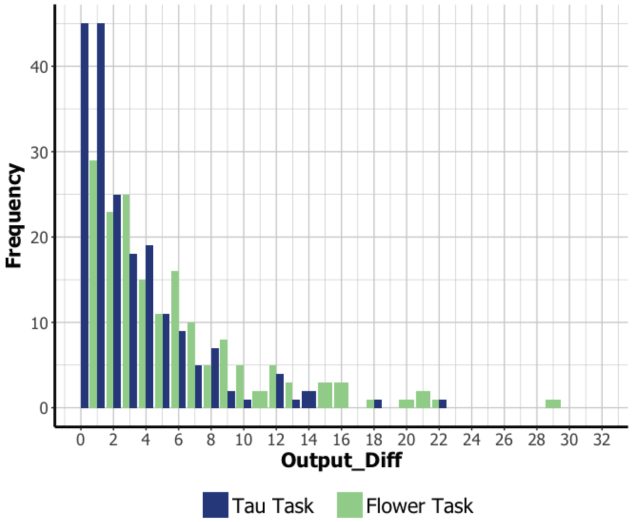


Figure 3: The distribution of $output_{orig,dup}$ is highly skewed, such that most workers exhibit a high degree of consistency for both tasks (*i.e.*, $output_{orig,dup} = 0$).

Results

A total of 402 workers were recruited through Amazon Mechanical Turk as participants for the study with approximately 50 workers assigned to each condition. On average, workers who completed the Flower task finished each image in 83.9 seconds ($\sigma = 60.0$) while workers who completed the Tau task finished each image in 97.9 seconds ($\sigma = 121.3$). 14 workers (6 Flower; 8 Tau) were filtered for apparent spammer behavior (*i.e.*, reporting a count of 1 for each image).

General Observations Table 2 reports the results for the ZIP models. The resulting models for both scenarios were similar within each task, but differed between the two tasks. Under the likelihood ratio test (Wasserman 2003), the models that contained only the main effect were sufficient for the Flower task, whereas the models that include the interaction terms of interest were more appropriate for the Tau task. This is not surprising since the tasks are different both nature and necessary domain expertise, and hence, the effect of the interaction terms differed.

Effects of Placement The effect of offset appeared to differ from task to task and from scenario to scenario in our study. When the workers counted flowers, an increase in offset decreases consistency (*i.e.*, increases $output_{orig,dup}$) significantly, which agrees with H1. This negative impact on consistency, although small, was consistent in both *S1* and *S2*. The similarity did not apply to the Tau-counting task: when ground-truth was available, offset had no significant effect. However, when ground truth was not available, an increase in offset significantly increased consistency, which is a complete opposite effect than than in the Flower task. The difference may be surprising, but when the magnitudes ($\hat{\beta}$) were considered, we see that the positive and inverse relationship between offset and consistency is very small (< 0.1). Thus, the statistical significance we observe may not be at all practical.

Effects of Difficulty Here, we restrict our analysis to only *S1* models, where the difficulty information is available. All three measures of difficulty: $difficulty_{dup}$, $difficulty_{< orig}$, and $difficulty_{orig,dup}$, were included into both *S1* models. As hypothesized in H2, the difficulty of the duplicate task showed a significant inverse relationship with consistency in both

Scenario	***S1***		***S2***	
Task	**Flower**	**Tau**	**Flower**	**Tau**
Deviance over df	4.05	2.87	5.12	3.38
Observed Zero Counts	26	40	26	40
Regular Poisson regression model				
Expected Zero Counts	5	15	2	11
ZIP regression model				
Expected Zero Counts	25	40	26	40

Table 1: Both ordinary Poisson and zero-inflated Poisson (ZIP) models were created for each scenario in each task. The expected number of zero counts for the regular Poisson regression model and the ZIP regression model are presented for each task. The ZIP model expectations are closer to the observed number of zero counts in all scenarios.

variable	$\hat{\beta}$	$Std.Error$	t	p
S1				
offset$_{orig,dup}$	0.04	0.01	2.91	**
transform (No)	0.04	0.07	0.54	
Noticed (No)	-0.30	0.10	-2.86	**
Identify (No)	0.12	0.18	0.67	
difficulty$_{dup}$	0.02	0.001	13.66	***
difficulty$_{< orig}$	0.002	0.002	1.12	
difficulty$_{orig,dup}$	0.01	0.002	2.82	**
transform (No) × *Noticed* (No)	NA	NA	NA	NA
transform(No) × *Identify*) (No)	NA	NA	NA	NA
S2				
offset$_{orig,dup}$	0.06	0.01	4.53	***
transform (No)	0.06	0.06	0.95	
Noticed (No)	-0.30	0.10	-2.96	**
Identify (No)	0.11	0.18	0.59	
transform (No) × *Noticed* (No)	NA	NA	NA	NA
transform) (No) × *Identify* (No)	NA	NA	NA	NA

(a) Models for counting flowers.

variable	$\hat{\beta}$	$Std.Error$	t	p
S1				
offset$_{orig,dup}$	-0.02	0.02	-1.05	
transform (No)	-3.92	0.72	-5.47	***
Noticed (No)	-0.87	0.27	-3.21	**
Identify (No)	-0.72	0.20	-3.70	***
difficulty$_{dup}$	0.02	0.002	8.23	***
difficulty$_{< orig}$	0.00	0.003	-1.32	
difficulty$_{orig,dup}$	0.01	0.003	1.79	.
transform (No) × *Noticed* (No)	2.48	0.62	3.98	***
transform (No) × *Identify*	1.32	0.52	2.56	*
S2				
offset$_{orig,dup}$	-0.07	0.02	-3.78	***
transform (No)	-4.53	0.71	-6.42	***
Noticed (No)	-0.81	0.26	-3.13	**
Identify (No)	-1.35	0.17	-7.85	***
transform (No) × *Noticed* (No)	2.49	0.60	4.14	***
transform (No) × *Identify* (No)	2.03	0.48	4.19	***

(b) Models for counting Greek taus.

Table 2: Zero-inflated Poisson (ZIP) regression models for counts for the Flower task and the Tau task. We report two models for each object: one that includes task difficulty (defined as expert consensus) (*S1*) and one that does not (*S2*).

Flower and Tau tasks (*i.e.*, as the difficulty of the duplicate task increased, consistency decreased). Similarly, there was strong evidence that an increase in the average difficulty of the tasks in between the original and duplicate tasks also decreased consistency when the workers were counting flowers. The same difficulty measures, however, had no significant impact on consistency among the workers who were counting Greek taus.

Effects of Transformation The relationship between consistency and applying a transformation to the duplicate task is not as straightforward as hypothesized in H3. In both scenarios considered, transformation had no significant effect on consistency when workers were counting flowers; but workers who were assigned to the Tau task had significantly higher consistency when no transformation was applied to the duplicate tasks. The high consistency is not unexpected as letters may appear differently and may be more challenging to recognize than flowers when flipped on the Y-axis. The effect of transformation remains insignificant with or without the workers recognizing or identifying the duplicates in the Flower task. In fact, the interaction terms involving transformation had negligible contribution to the variability of consistency measure. The interaction terms showed strong significance in both models for the Tau task, indicating the effect of transformation is dependent on workers' ability to recognize if there were a duplicate task, and workers' ability to correctly identify the duplicate. Further discussion occurs in the next section.

Effects of Recognition and Recall 95% of the participants in the high-offset conditions self-reportedly did not recognize the duplicate compared to 90% of the participants in the low-offset conditions. As shown in Table 2, the ability to recognize that there were duplicates significantly decreased consistency (*i.e.*, increased *output*$_{orig,dup}$) in all cases. This can be attributed to workers doubting their initial count, or workers recognizing the task as a retest for reliability and strategically gaming the system. There is also strong evidence that the Tau-counting workers remember prior answers as the significant interaction term (*transform* × *Noticed*) showed that when the image was not transformed, the workers who recognized that there were duplicates had significantly higher consistency (*i.e.*, lower *output*$_{orig,dup}$) than those who did not recognize the presence of duplicates. The same argument can be applied to workers' ability to correctly recall and identify the duplicate task when performing the Tau task: consistency decreases when workers could identify the duplicated task. This finding does not apply to the Flower task. Furthermore, when presented with duplicates that were not transformed, the Tau-counting workers who were able to identify the duplicates correctly had significantly higher consistency, as shown by the significant interaction term *transform* × *Identify*. Recognition was initially thought to be inversely related to consistency (H4), but the results from our data set suggest this relationship depends on the type of task and workers' ability to recall and recognize the tasks they performed.

Relationship between Consistency and Accuracy

The assumption behind consistency as a measure of reliability is that a consistent worker is likely "good" worker who produces high quality results. Here, we challenge this assumption by analyzing the relationship between consistency and accuracy.

Scoring Workers We represent each worker with two scores. First, *consistency score* (*i.e.*, *output*$_{orig,dup}$) measures the distance between the worker's reported counts for the original and duplicate tasks. The second score, *consensus score*, is the distance between the worker's reported count for an image and the median count from all workers who processed that image (Ribeiro et al. 2011). Consensus methods, ranging from simple majority-vote approaches to sophisticated machine-learning based models, have become a standard for ensuring data quality when true measures of it are otherwise absent (Ipeirotis, Provost, and Wang 2010; Jung and Lease 2011). We chose consensus as a point of comparison because it is a frequently used proxy for reliability. Finally, we define *error* as the absolute difference between reported count and ground truth count for an image.

Methods Various linear regression models were used to model the relationship between error and both consensus scores and consistency scores. Task difficulty and its interaction with the scores were also included in the respective models to account for blocking effect and to improve estimation. Similar to the ZIP models, we used separate linear models for the Flower and Tau tasks and verified there were no clear violations of the model's assumptions by checking the residual plots.

Results In general, both the relationship between error and consensus, and between error and consistency differed by task. Workers, who tend to agree with the majority, are more accurate for both tasks (Flower: $\hat{\beta} = 1.02, t(1767) = 58.59, p < 0.001$; Tau: $\hat{\beta} = 0.71, t(1767) = 4.11, p < 0.001$). The interaction effects suggest that, given the same consensus score, workers' accuracy decreases significantly when task difficulty level increases. Although consistent workers showed significantly higher accuracy when counting flowers ($\hat{\beta} = 0.68, t(1767) = 9.42, p < 0.001$), their accuracy at counting Greek taus is not affected by their consistency ($\hat{\beta} = 0.04, t(1767) = 0.58, p = 0.56$). Similar to the effect of the consensus scores, given the same consistency score, workers' accuracy at the Flower task decreased significantly when task difficulty level increased. The same cannot be said for the Tau task.

Discussion

Consistency as a Measure of Reliability

Consistency—characterizing workers based on how well they agree with themselves—can be an effective supplement to existing measures of reliability. Our work contributes to the design of consistency-based reliability measures by studying the effects of task characteristics on worker consistency. Alongside our examination of task characteristics, we presented *Deja Vu*, a general and simple mechanism for controlling the distribution of duplicate tasks and studying the consistency of workers. Collectively, our work introduces open questions on developing more complex consistency-based metrics that take into account the nuanced effects of factors that influence consistency, and machine learning methods for statistically modeling consistency in workers. Exploring consistency across other common types of crowdsourcing tasks, within complex crowdsourcing workflows, and in worker populations that are intrinsically motivated to participate (*e.g.*, citizen scientists) are all important directions of future work.

Fatigue and Learning Effects

Our results show that the number of tasks between the original and duplicate task (*i.e.*, $\textit{offset}_{\textit{orig,dup}}$) may influence consistency (*i.e.*, $\textit{output}_{\textit{orig,dup}}$) in both tasks. One interpretation of these results is that learning can be influential factor for consistency. However, we provided both a training video and referential material for detecting the objects during the task, a standard practice that is thought to mitigate such a learning curve (Doroudi et al. 2016). An alternative interpretation of these results is that workers may become fatigued and perform inadequately, which is supported by prior literature in crowdwork (Chandler and Kapelner 2013; Franklin et al. 2011; Rzeszotarski et al. 2013). As we only examined task queues of one size, we cannot confidently conclude this is the case.

Explanations aside, these findings collectively suggest that a trade-off exists for determining *when* to route the original task. Distributing the task before the worker has had proper experience performing the task may facilitate experiential learning until the duplicate has been dispatched. Conversely, the likelihood of worker experiencing fatigue and exhaustion grows by delaying the presentation of the duplicate. Prior work (Bragg, Mausam, and Weld 2016) has addressed the problem of scheduling validation tasks for maintaining data quality, but not in the context of performing the same task *again*. A promising direction for future work is to develop a solution that optimizes the learning-fatigue tradeoff in routing duplicate tasks.

Recognition and Carryover Effects

Our results show that both the recognition and the transformation of the duplicate had a strong effect on consistency, indicating the possible presence of the carryover effect (*i.e.*, remembering the answer from the original task). If malicious workers are able to identify a duplicate, they may shift their strategy—realizing that they have already passed the *consistency test*, they may complete the remainder of the tasks in a hurry, without due attention to quality. As our results suggest the transformation had a significant effect for only one of the studied tasks, future research can investigate the influence of carryover effects on worker consistency, develop additional methods for transforming and obfuscating the duplicate task (*e.g.*, blurring part of an image), and better understand recognition in the context of larger task queue sizes.

Limitations

Our work is grounded in object counting tasks within two, unique domains where workers were given task queues in limited length. Our study did not examine the effects of consistency in larger task queue sizes, nor does it draw conclusions related to other, more complex task types. Our work also studied consistency in the narrow context of probes with a only single duplicate. Our work does not study the advantages and disadvantages of assessing worker consistency with a larger number of duplicates in worker task queues. However, we regard these topics as important and necessary directions of future work for informing the design and establishment of consistency-based reliability metrics in crowdwork.

Conclusion

Consistency can be a powerful supplement to traditional quality control methods. In this work, we introduced *Deja Vu*, a mechanism for routing duplicate tasks and assessing workers based on their ability to complete duplicate tasks consistently. We presented findings from an experiment that showed how the duplicate's difficulty, position, and transformation affect worker consistency in two, unique counting tasks. Future work includes investigating consistency as a reliability metric in other contexts, developing more complex metrics and investigating machine learning techniques for modeling consistency, understanding how consistency can be measured within complex workflows (*i.e.*, hierarchical tasks), and studying other factors (*i.e.*, carryover effects) that may affect consistency.

Acknowledgments

We acknowledge and thank the participants recruited from Amazon Mechanical Turk for participating in our study. We also acknowledge the Egypt Exploration Society for providing access to the dataset of papyri images used in the experiment. This research was funded by an NSERC Discovery grant (RGPIN-2015-04543) and an NSF-DBI grant (EF1208835).

References

Birnbaum, B.; Borriello, G.; Flaxman, A. D.; DeRenzi, B.; and Karlin, A. R. 2013. Using behavioral data to identify interviewer fabrication in surveys. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2911–2920. ACM.

Bragg, J.; Mausam; and Weld, D. S. 2016. Optimal testing for crowd workers. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, 966–974. International Foundation for Autonomous Agents and Multiagent Systems.

Buchholz, S., and Latorre, J. 2011. Crowdsourcing preference tests, and how to detect cheating. In *INTERSPEECH*, 3053–3056.

Cai, C. J.; Iqbal, S. T.; and Teevan, J. 2016. Chain reactions: The impact of order on microtask chains. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, 3143–3154. New York, NY, USA: ACM.

Chandler, D., and Kapelner, A. 2013. Breaking monotony with meaning: Motivation in crowdsourcing markets. *Journal of Economic Behavior & Organization* 90:123–133.

Cheng, J.; Teevan, J.; and Bernstein, M. S. 2015. Measuring crowdsourcing effort with error-time curves. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 1365–1374. ACM.

Chinn, S. 1991. Statistics in respiratory medicine. 2. repeatability and method comparison. *Thorax* 46(6):454–456.

Dawid, A. P., and Skene, A. M. 1979. Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied statistics* 20–28.

Demartini, G.; Difallah, D. E.; and Cudré-Mauroux, P. 2012. Zencrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In *Proceedings of the 21st international conference on World Wide Web*, 469–478. ACM.

Doroudi, S.; Kamar, E.; Brunskill, E.; and Horvitz, E. 2016. Toward a learning science for complex crowdsourcing tasks. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 2623–2634. ACM.

Forsyth, D. A., and Ponce, J. 2002. *Computer vision: a modern approach*. Prentice Hall Professional Technical Reference.

Franklin, M. J.; Kossmann, D.; Kraska, T.; Ramesh, S.; and Xin, R. 2011. Crowddb: answering queries with crowdsourcing. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, 61–72. ACM.

Hata, K.; Krishna, R.; Fei-Fei, L.; and Bernstein, M. 2017. A glimpse far into the future: Understanding long-term crowd worker accuracy. In *CSCW: Computer-Supported Cooperative Work and Social Computing*.

Henricksen, K.; Indulska, J.; and Rakotonirainy, A. 2002. Modeling context information in pervasive computing systems. In *Pervasive Computing*. Springer. 167–180.

Hornbæk, K.; Sander, S. S.; Bargas-Avila, J. A.; and Grue Simonsen, J. 2014. Is once enough?: on the extent and content of replications in human-computer interaction. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, 3523–3532. ACM.

Ipeirotis, P. G.; Provost, F.; and Wang, J. 2010. Quality management on amazon mechanical turk. In *Proceedings of the ACM SIGKDD workshop on human computation*, 64–67. ACM.

Ipeirotis, P. G. 2010. Analyzing the amazon mechanical turk marketplace. *XRDS: Crossroads, The ACM Magazine for Students* 17(2):16–21.

Jung, H. J., and Lease, M. 2011. Improving consensus accuracy via z-score and weighted voting. In *Proceedings of the 2011 AAAI Workshop on Human Computation*.

Lambert, D. 1992. Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics* 34(1):1–14.

Law, E.; Dalton, C.; Merrill, N.; Young, A.; and Gajos, K. Z. 2013. Curio: A platform for supporting mixed-expertise crowdsourcing. In *Proceedings of the First AAAI Conference on Human Computation and Crowdsourcing*.

Mao, A.; Kamar, E.; Chen, Y.; Horvitz, E.; Schwamb, M. E.; Lintott, C. J.; and Smith, A. M. 2013. Volunteering versus work for pay: Incentives and tradeoffs in crowdsourcing. In *Proceedings of the First AAAI Conference on Human Computation and Crowdsourcing*.

Margolis, S. A., and Duewer, D. L. 1996. Measurement of ascorbic acid in human plasma and serum: stability, intralaboratory repeatability, and interlaboratory reproducibility. *Clinical Chemistry* 42(8):1257–1262.

Misztal, I.; Legarra, A.; and Aguilar, I. 2009. Computing procedures for genetic evaluation including phenotypic, full pedigree, and genomic information. *Journal of Dairy Science* 92(9):4648–4655.

Newell, E., and Ruths, D. 2016. How one microtask affects another. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, 3155–3166. New York, NY, USA: ACM.

Raykar, V. C., and Yu, S. 2012. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *Journal of Machine Learning Research* 13(Feb):491–518.

Ribeiro, F.; Florêncio, D.; Zhang, C.; and Seltzer, M. 2011. Crowdmos: An approach for crowdsourcing mean opinion score studies. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2416–2419. IEEE.

Rosten, E.; Porter, R.; and Drummond, T. 2010. Faster and better: A machine learning approach to corner detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 32:105–119.

Ryan, R. M. 1982. Control and information in the intrapersonal sphere: An extension of cognitive evaluation theory. *Journal of Personality and Social Psychology* 43(3):450.

Rzeszotarski, J. M., and Kittur, A. 2011. Instrumenting the crowd: Using implicit behavioral measures to predict task performance. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, 13–22. New York, NY, USA: ACM.

Rzeszotarski, J. M.; Chi, E.; Paritosh, P.; and Dai, P. 2013. Inserting micro-breaks into crowdsourcing workflows. In *Proceedings of the First AAAI Conference on Human Computation and Crowdsourcing*.

Sarma, A. D.; Jain, A.; Nandi, A.; Parameswaran, A.; and Widom, J. 2015. Surpassing humans and computers with jellybean: Crowd-vision-hybrid counting algorithms. In *Third AAAI Conference on Human Computation and Crowdsourcing*.

Sheng, V. S.; Provost, F.; and Ipeirotis, P. G. 2008. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 614–622. ACM.

Sheshadri, A., and Lease, M. 2013. Square: A benchmark for research on computing crowd consensus. In *Proceedings of the First AAAI Conference on Human Computation and Crowdsourcing*.

Simpson, R.; Page, K. R.; and De Roure, D. 2014. Zooniverse: observing the world's largest citizen science platform. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, 1049–1054. International World Wide Web Conferences Steering Committee.

Snow, R.; O'Connor, B.; Jurafsky, D.; and Ng, A. Y. 2008. 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*, 254–263. Association for Computational Linguistics.

Society, E. E. 1908. *Graeco-Roman Memoirs*. Number v. 9 in Graeco-Roman Memoirs.

Sun, P., and Stolee, K. T. 2016. Exploring crowd consistency in a mechanical turk survey. In *Proceedings of the 3rd International Workshop on CrowdSourcing in Software Engineering*, 8–14. ACM.

Teevan, J.; Iqbal, S. T.; and von Veh, C. 2016. Supporting collaborative writing with microtasks. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, 2657–2668. New York, NY, USA: ACM.

Vuong, Q. H. 1989. Likelihood ratio tests for model selection and non-nested hypotheses. *Econometrica* 57(2):307–333.

Wasserman, L. 2003. *All of Statistics: A Concise Course in Statistical Inference*. Berlin: Springer-Verlag.

Welinder, P.; Branson, S.; Belongie, S. J.; and Perona, P. 2010. The multidimensional wisdom of crowds. In *Advances in Neural Information Processing Systems*, volume 23, 2424–2432.

Whitehill, J.; Wu, T.-f.; Bergsma, J.; Movellan, J. R.; and Ruvolo, P. L. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in Neural Information Processing Systems*, 2035–2043.

Willis, C. G.; Law, E.; Williams, A. C.; Franzone, B. F.; Bernardos, R.; Bruno, L.; Hopkins, C.; Schorn, C.; Weber, E.; Park, D. S.; and Davis, C. C. 2017. CrowdCurio: an online crowdsourcing platform to facilitate climate change studies using herbarium specimens. *New Phytologist*.

Wilson, M. L.; Mackay, W.; Chi, E.; Bernstein, M.; Russell, D.; and Thimbleby, H. 2011. Replichi-chi should be replicating and validating results more: discuss. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, 463–466. ACM.