Noname manuscript No.
(will be inserted by the editor)

Crowd Anatomy Beyond the Good and Bad: Behavioral Traces for Crowd Worker Modeling and Pre-selection

Ujwal Gadiraju · Gianluca Demartini · Ricardo Kawase · Stefan Dietze

Received: date / Accepted: date

Abstract The suitability of crowdsourcing to solve a variety of problems has been investigated widely. Yet, there is still a lack of understanding about the distinct behavior and performance of workers within microtasks. In this paper, we first introduce a fine-grained data-driven worker typology based on different dimensions and derived from behavioral traces of workers. Next, we propose and evaluate novel models of crowd worker behavior and show the benefits of behavior-based worker pre-selection using machine learning models. We also study the effect of task complexity on worker behavior. Finally, we evaluate our novel typology-based worker pre-selection method in image transcription and information finding tasks involving crowd workers completing 1,800 HITs. Our proposed method for worker pre-selection leads to a higher quality of results when compared to the standard practice of using qualification or pre-screening tests. For image transcription tasks our method resulted in an accuracy increase of nearly 7% over the baseline and of almost 10% in information finding tasks, without a significant difference in task completion time. Our findings have important implications for crowdsourcing systems where a worker's behavioral type is unknown prior to participation in a task. We highlight the potential of leveraging worker types to identify and aid those workers who require further training to improve their performance. Having proposed a powerful automated mechanism to detect worker types, we reflect on promoting fairness, trust and transparency in microtask crowdsourcing platforms.

U. Gadiraju, S. Dietze
L3S Research Center, Leibniz Universität Hannover, Germany
E-mail: lastname@L3S.de

G. Demartini
School of ITEE, University of Queensland, Australia
E-mail: g.demartini@uq.edu.au

R. Kawase
mobile.de GmbH/eBay Inc., Berlin, Germany
E-mail: rkawase@team.mobile.de

Keywords Behavioral Traces · Crowdsourcing · Microtasks · Pre-selection · Pre-screening · Workers · Worker Typology

1 Introduction

"A worker may be the hammer's master, but the hammer still prevails. A tool knows exactly how it is meant to be handled, while the user of the tool can only have an approximate idea."

— *Milan Kundera*

A primary challenge in microtask crowdsourcing is quality assurance (Kittur et al., 2013). Various aspects can effect the quality of data collected (Ipeirotis et al., 2010), ranging from poor HIT (Human Intelligence Task) design to the presence of malicious activity (Gadiraju et al., 2015b). To improve crowdsourced data quality, early work has focused on aggregating multiple answers from different workers in the crowd by going beyond the simple majority vote (Demartini et al., 2012; Sheshadri and Lease, 2013; Venanzi et al., 2014). Other works have focused on modeling worker skills and interests to assign available HITs to them (Difallah et al., 2013; Bozzon et al., 2013). Authors have also proposed the use of gamification (Feyisetan et al., 2015a) and collaboration (Rokicki et al., 2015) to improve the effectiveness of the paradigm and balance the costs with benefits.

Rzeszotarski and Kittur, proposed to track worker activity to distinguish between *good* and *bad* workers according to their performance (Rzeszotarski and Kittur, 2011). Recently, Dang et al. built a framework called *mmm*Turkey, by leveraging this concept of tracking worker activity (Dang et al., 2016). Rzeszotarski et al. showed several benefits of their approach when compared to other quality control mechanisms due to aspects such as effort, skill and behavior that can be interpreted through a worker's activity, and eventually help in predicting the quality of work (Rzeszotarski and Kittur, 2011, 2012). While it is certainly useful to predict good versus bad quality of work, we argue that further benefits can be revealed by understanding worker activity at a finer level of granularity. For example, the knowledge that even *good* workers perform and operate in different ways to accomplish tasks, leads to the question of whether such differences can have practical implications.

With the rise in adoption of crowdsourcing solutions that leverage human input through microtask marketplaces, new requirements have emerged. Often it is not sufficient to predict the quality of work alone when there are additional constraints on costs (in terms of time and money). Moreover, a better understanding of how good workers differ in complex crowdsourcing tasks can lead to further benefits like improved HIT design or HIT assignment models.

Research Questions and Original Contributions.

This paper aims at filling this knowledge gap by contributing novel insights on worker behavior in microtask crowdsourcing. We aim to understand and identify the different types of workers in the crowd by focusing on *worker behavior*. Our objective is to advance the current understanding of different types of workers present in a crowdsourcing platform and leverage this for worker pre-selection, given a task to be completed. By combining quantitative analysis and proposing a supervised machine learning model, we seek to answer the following research questions.

RQ#1 How can requesters leverage worker behavioral traces and benefit from the knowledge of worker types at a fine granularity?

We collected activity tracking data from workers completing 1,800 HITs with varying length, type, and difficulty. We refined the existing understanding of worker types and extended it to multi-dimensional definitions within a *worker typology*. We experimentally showed that it is possible to automatically classify workers into granular classes based on supervised machine learning models that use behavioural traces of workers completing HITs. Leveraging such worker type classification, we can improve the quality of crowdsourced tasks by pre-selecting workers for a given task.

RQ#2 What is the impact of *task type* and *task complexity* on the behavior of crowd workers?

We considered the two different task types of *content creation* and *information finding*. We deployed tasks where workers had to transcribe images in case of the content creation tasks, or find the middle-names of personalities in case of the information finding tasks. We varied the task complexity of both these types of tasks in order to analyze the impact of task complexity on the behavior and performance of workers across both types of tasks. Based on our experiments and results we found that pre-selection based on worker types significantly improves the quality of the results produced, particularly in tasks with high complexity.

RQ#3 How effective is behavior-based pre-selection of crowd workers?

Our pre-selection method based on worker types yields an improvement of up to 10% compared with standard worker pre-selection techniques, without effecting the task completion time of workers. Predicting worker types can have important implications on promoting trust and transparency in crowd work. For example, workers can receive feedback and training that is personalized to their worker type. Workers can be made aware of their type and supported towards becoming more effective and efficient.

2 Related Literature

2.1 Modeling Crowd Workers

Crowd worker behavior is influenced by several aspects, some of which are inherent to the worker (such as trustworthiness of a worker (Gadiraju et al., 2015b)) and others that are induced by the nature of tasks (such as *task complexity* (Yang et al., 2016)). Workers can be categorized based on the quality of their work. Some categories proposed by (Gadiraju et al., 2015b) and (Kazai et al., 2011) include elite workers, competent workers, less-competent workers, and so forth. As described by (Eickhoff et al., 2012), *money-driven* workers are motivated by the monetary incentives, while *entertainment-driven* workers mainly seek diversion but readily accept the monetary rewards as additional extrinsic motivation.

Work on understanding and modeling worker behavior includes (Kazai et al., 2011), where authors proposed worker types based on outcomes of behavior, such as the amount of time spent on the task and quality of the work produced. Authors defined four classes of workers: diligent (workers completing the task carefully), competent (efficient and effective workers), incompetent (workers with a poor understanding of the task), and spammers.

Kazai et al. observed that varying task design properties (task difficulty and reward) has an impact on the type of crowd which completes the task, and workers' performance based on their interest and perceived challenge (Kazai et al., 2013). Authors found that workers who were bored underperformed and workers who found the task difficult exhibited a lower accuracy. Difallah et al. proposed building worker models by indexing Facebook pages that workers liked, to assign HITs to those workers whose skills and interests best fit the task at hand (Difallah et al., 2013). Vuurens and De Vries introduced a worker taxonomy focusing on different types of workers who performed poorly (Vuurens and De Vries, 2012). The authors compared diligent workers, sloppy workers (i.e., workers who were honest but provided low quality labels), random spammers (workers with an inter-worker agreement rate close to random), and uniform spammers (workers who repeated the same answer across the task). The authors proposed methods to automatically detect such workers based on comparing their responses with those from other workers. Recent work by Gadiraju et al. focused on the understanding of malicious behavior exhibited by workers in the crowd (Gadiraju et al., 2015b), where authors observed different malicious techniques used by workers to complete HITs with the sole purpose of maximizing their monetary rewards, without providing a quality response. Authors typecasted unreliable workers mainly as being either *fast deceivers* (i.e., workers who copy-paste the same response in multiple fields or provide quick and random responses to maximize their earnings), *smart deceivers* (i.e., workers who exert little effort by providing suboptimal responses without triggering underlying response validators), and *rule breakers* (i.e., workers who do not conform to the requirements laid down by requesters, thereby providing responses which are not entirely useful in the best case and completely useless in the worst case).

These prior works primarily focus on the outcomes of work completed to typecast workers. We advance the understanding of worker types by integrating the different dimensions considered in lone typologies in each case of previous work. The result is a holistic perspective of behavior, performance and motivation for each category in the proposed worker typology with a higher granularity of worker behavior. We discuss this further in Section 3.3.

2.2 Quality Control

One of the classic approaches to detect low quality work, is to compare worker responses against a gold standard dataset. Oleson et al. proposed the programmatic creation of gold standard data to provide targeted training feedback to workers and prevent common scamming scenarios. Authors found that it decreases the amount of manual work required to manage crowdsourced labor while improving the overall quality of the results (Oleson et al., 2011). Similarly, Wang et al. proposed to seamlessly integrate gold data (i.e., data with priorly known answers) for learning the quality of workers (Wang et al., 2011).

Another traditional way to increase label quality generated by means of microtask crowdsourcing is to rely on redundancy; by assigning the same task to a number of workers and then aggregating their responses. Sheshadri and Lease have been benchmarked such techniques over a set of crowd generated labels, comparing state of the art methods over the classic majority vote aggregation method (Sheshadri and Lease, 2013). More recently, Venanzi et al. proposed an advanced response aggregation technique that weights crowd responses based on measures of workers similarity, showing a significant improvement in label accuracy (Venanzi et al., 2014).

In other works, Marshall and Shipman proposed the use of psychometric tests to ensure reliability of responses from workers (Marshall and Shipman, 2013). Rzeszotarski and Kittur looked at worker tracking data with the purpose of distinguishing between high and low performing workers (Rzeszotarski and Kittur, 2011). Additionally, the authors presented visual analytics tools that allow requesters to observe worker performance and identify low performers to be filtered out (Rzeszotarski and Kittur, 2012). Kazai et al. proposed to look at worker demographics and personality traits as indicators of work quality (Kazai et al., 2012). Qualification tests and pre-screening methods have also been adopted in order to select appropriate workers for a given task. Recent work by Gadiraju et al. has proposed the use of worker self-assessments for pre-selection (Gadiraju et al., 2017b).

A limitation of prior works on quality control based on worker typologies is the absence of prior knowledge about worker types in typical scenarios, and the lack of automated methods that go beyond identifying *good* and *bad* performing workers. Our work is complementary to aforementioned prior works, in that we aim to improve the quality of work that is produced by workers.

In addition, by relying on a more granular understanding of worker types, we afford pre-selection of desired workers in the absence of any prior information about workers. We extract behavioral features and propose a supervised machine learning model, that automatically detects worker types, thus going beyond the good/bad binary classification problem.

3 Methodology and Setup

3.1 Methodology

To address the research questions stated earlier, we consider the task types of *Content Creation* and *Information Finding* (Gadiraju et al., 2014). A recent study on the dynamics of crowdsourced tasks on Amazon's Mechanical Turk (AMT) showed that content creation tasks have been the most popularly crowdsourced tasks over the last 5 years, while information finding tasks have depicted the most growth over the last 3 years (Difallah et al., 2015).

In a seminal work on task complexity (Wood, 1986), the author suggested *component complexity* to be a type of task complexity. Wood posited that component complexity is a direct function of the number of distinct acts that need to be executed while performing the task, and the number of distinct information cues that must be processed in order to do so. Wood suggested that as the number of acts required to be carried out increase, the knowledge and skill requirements for a task also increase, simply because there are more activities and events that an individual needs to be aware of and able to perform.

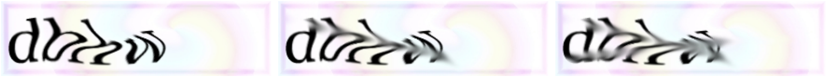
One can model *task complexity* (Yang et al., 2016) from a worker's point of view, where worker competence for example, could play a role in determining how complex a given task is. This is logically sound, since one worker can find a given task to be difficult while another can find the same task to be simple. However, including inherent worker traits in task complexity modeling would make it subjective.

To define task complexity from a purely objective standpoint, we consider the characteristics of the task alone. Herein, we model task complexity as a function of (i) the objective difficulty-level of the task and (ii) the length of the task.

3.1.1 Microtask Design - Content Creation

Due to its popularity, we choose to use *image transcription* as the content creation task in our experiments. For this purpose, we use a dataset of captchas[1] where crowd workers are asked to decipher characters from a given image (Gadiraju et al., 2015a). To cater to varying task complexity and observe consequent behavior of workers participating in the tasks, we consider tasks with

[1] `http://www.captcha.net/`



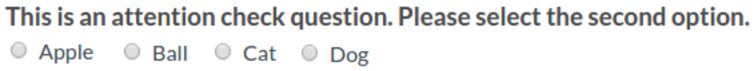
(a) Difficulty Level I (`no-stroke`)

(b) Difficulty Level II (`one-stroke`)

(c) Difficulty Level III (`two-strokes`)

Fig. 1: Progressive difficulty levels in the content creation task of transcribing captcha images.

lengths of 20, 30, and 40 units respectively. In each unit a worker is asked to transcribe a captcha. Apart from this, to model the difficulty-level aspect of task complexity, we use the objective notion of smudging the captchas with `no-stroke`, `one-stroke`, and `two-strokes` to indicate a progressively increasing difficulty-level of tasks (as shown in Figure 1). Thus, we aim to replicate the objective reality of image transcription tasks where some images can be easier to transcribe than others. This follows Wood's explanation of component complexity, since the distinct act of identifying and transcribing a captcha increases due to decreasing legibility of the characters across each level.



This is an attention check question. Please select the second option.

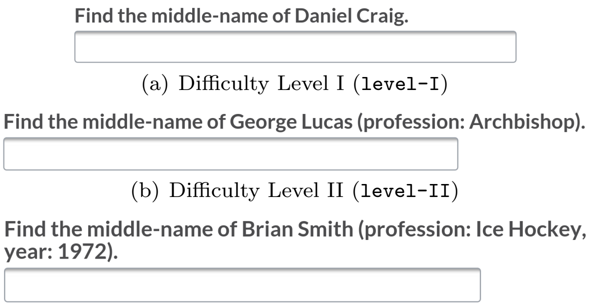
○ Apple ○ Ball ○ Cat ○ Dog

Fig. 2: Question to assess *trustworthiness* of workers.

To deduce the *trustworthiness* of a worker as demonstrated in (Gadiraju et al., 2015b), we intersperse multiple choice questions between the image transcription units at a regular interval of 25% of total units. We explicitly ask workers to pick a given option, (as shown in Figure 2), and due to the fact that this is a change in question format (from transcribing an image in a text field to answering a multiple choice question) we believe that it is not possible for a trustworthy worker to miss the direct and clear instruction. We consider workers who answer one or more of these questions incorrectly to be *untrustworthy.*

3.1.2 Microtask Design - Information Finding

For the information finding type, we adopt the task of finding the middle-names of famous people. To investigate the effect of varying task complexity on worker behavior, we consider tasks with length of 10, 20, and 30 units (since this type of task requires more time for completion in comparison to the content creation task of image transcription). In each unit, a worker is asked to find the middle-name of a given person. We model the task difficulty objectively into 3 levels, wherein workers need to consider an additional aspect in each progressively difficult level as shown in Figure 3.



Find the middle-name of Daniel Craig.

(a) Difficulty Level I (`level-I`)

Find the middle-name of George Lucas (profession: Archbishop).

(b) Difficulty Level II (`level-II`)

Find the middle-name of Brian Smith (profession: Ice Hockey, year: 1972).

(c) Difficulty Level III (`level-III`)

Fig. 3: Progressive difficulty levels in the Information Finding task of finding the middle-names of famous persons.

In `level-I`, workers are presented with unique names of famous persons, such that the middle-names can be found using a simple Google or Wikipedia search. In `level-II` workers are additionally provided with the profession of the given person. We manually selected the names such that there are at least two different individuals with the given names in `level-II`, and the distinguishing factor that the workers need to rely upon is their profession. In `level-III` workers are presented names of persons, their profession, and a year during which the persons were active in the given profession. There are multiple distinct individuals with the given names, associated with the same profession in `level-III`. The workers are required to identify the accurate middle-name by relying on the year in which the person was active in the profession. The progressively difficult levels in the information findings tasks are analogous to Wood's definition of component complexity, where the number of distinct information cues that must be processed increase by a factor of one and the number of acts that need to be executed (the number of units) increase by 10 units. We use the same method adopted in the content creation tasks to determine the *trustworthiness* of workers in these information finding tasks.

3.2 Experimental Setup

We deployed 9 tasks of the content creation type, with varying combinations of length (20, 30, 40 units) and objective difficulty-levels (no-stroke, one-stroke, two-strokes) on CrowdFlower[2]. Similarly we deployed a further 9 tasks of the information finding type, with varying combinations of length (10, 20, 30 units) and objective difficulty (level-I, level-II, level-III). For each of these 18 tasks, we gathered responses from 100 distinct workers resulting in a total of 1,800 HITs. We deployed tasks of the same type and difficulty-level concurrently, in

[2] `http://www.crowdflower.com/`

order to avoid potential learning biases. Workers were paid in accordance to the task complexity of a given task (10, 20, 30 USD cents per unit).

3.2.1 Tracking Worker Activity

In the context of crowdsourcing, user tracking techniques have been used for a variety of reasons. Examples include work done by Cheng et al., where authors logged browser window focus changes to understand interruptions (Cheng et al., 2015). Feyisetan et al. used mouse tracking to generate heatmaps over HITs to see which part workers focused on (Feyisetan et al., 2015b). More recently, Kazai et al. used behavioral data to compare experts and crowd workers on HITs pertaining to relevance judgments (Kazai and Zitouni, 2016).

3.2.2 Mousetracking Implementation:

We implemented mousetracking using Javascript and the JQuery library, and logged user activity data ranging from mouse movements to keypresses. We took measures to distinguish between workers that use a mouse and those who use a touchpad. We also distinguish between worker mannerisms with respect to scrolling behavior; use of scrollbar as opposed to the mousewheel. In this way, we gathered worker activity data from each of the experimental tasks deployed on CrowdFlower. Apart from this data, we use a Javascript implementation[3] of *browser fingerprinting* (Eckersley, 2010) in order to identify workers that participate in tasks multiple times (*'repeaters'*) by virtue of using different `worker-ids` (Gadiraju and Kawase, 2017). We take measures to avoid privacy intrusion of workers by hashing various browser characteristics such as the user agent, cookies settings, screen resolution, and so forth, results in a 64-bit browser fingerprint. We do not retain any worker-specific browser traits other than the resulting fingerprint to identify repeaters.

3.3 Modeling Worker Behavior

We present a worker typology by building on prior works described in Section 2.1 in an inductive and data-driven fashion prescribed by Berg et al. (Berg, 2004). To summarize, Kazai el al. (Kazai et al., 2011), Gadiraju et al. (Gadiraju et al., 2015b), and Vuurens and De Vries (Vuurens and De Vries, 2012) proposed worker typologies based on worker behavior and performance, while Eickhoff et al. (Eickhoff et al., 2012) categorized workers based on their motivation. We propose to combine behavior, motivation, and performance rather than looking at each aspect individually to typecast workers from a holistic standpoint. Based on the responses provided by workers in the 1,800 HITs described earlier, we computed their performance. We explicitly gathered information from workers regarding their motivation for participation. Finally,

[3] `http://github.com/Valve/fingerprintjs`

based on the low-level worker activity that we logged, we were able to analyze worker behavior.

To categorize workers based on their performance (accuracy and task completion time), motivation, and behavior we used a data-driven and inductive approach. This means that the categories we thereby derived were grounded in the data from which they emerged, as suggested by Denzin (Denzin, 1978) as well as Glaser and Strauss (Strauss and Glaser, 1967). We manually inspected workers' responses to the 1,800 HITs and built rubrics around their task completion time, trustworthiness, and performance to assign appropriate labels. The rubrics were such that worker types could be assigned without clashes between the classes. Three authors of this paper acted as experts and designed a coding frame according to which we could decide which category in the typology a worker belonged to. In case, the characteristics exhibited by workers did not fit any existing category, a new one was created. After resolving disagreements on the coding frame every worker was labeled with a category. We followed the guidelines suggested by (Strauss, 1987; Berg, 2004) while conducting the open-coding of behavioral data, collected over the 1,800 HITs run on CrowdFlower, leading to the following categories[4]. We also describe the rubrics used to categorize workers into the respective category.

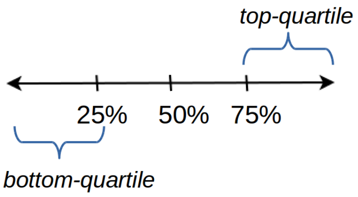


Fig. 4: Quartiles of task completion time and accuracy of workers used within rubrics for categorization of workers in the proposed typology. The first quartile is the bottom-quartile, followed by the 2^{nd}, 3^{rd} and the 4^{th} quartile, which is also called the top-quartile.

- **Diligent Workers (DW).** These crowd workers may be *money-driven* or *entertainment-driven*. They make sure to provide high quality responses and spend a long time to ensure good responses.

  Rubric used to categorize DW: trustworthy workers who have high to very high task completion times (i.e., 3^{rd} and 4^{th} quartiles of task completion times among all workers in the given task), and high to very high accuracy (i.e., 3^{rd} and 4^{th} quartiles of accuracy among all workers in the given task).
- **Competent Workers (CW).** These crowd workers may be *money-driven* or *entertainment-driven*. They possess skills necessary to complete tasks in a quick and effective manner, producing high quality responses.

[4] Note that worker types describe session-level behavior of the workers rather than properties of a person.

Rubric used to categorize CW: trustworthy workers who have very low to low task completion times (i.e., first 2 quartiles of task completion times among all workers in the given task), and high to very high accuracy (i.e., 3^{rd} and 4^{th} quartiles of accuracy among all workers in the given task).

– **Fast Deceivers (FD).** These crowd workers are *money-driven*, and attempt to complete a given task in the fastest possible way to attain the rewards offered. Due to this, *fast deceivers* provide poor responses by copy-pasting content and taking advantage of loopholes in the task design (such as weak or missing validators).

Rubric used to categorize FD: untrustworthy workers[5] who have low to very low task completion times (i.e., first 2 quartiles of task completion times among all workers in the given task), and very low accuracy (i.e., the bottom quartile of accuracy among all workers in the given task).

– **Smart Deceivers (SD).** These crowd workers are *money-driven* and aware of potential validators and checks that task requesters may be using to flag workers (such as minimum time spent on a question). They provide poor responses without violating validators, and thereby exert less effort to attain the incentives.

Rubric used to categorize SD: trustworthy workers who have high task completion times (i.e., 3^{rd} quartile of task completion times among all workers in the given task), and very low accuracy (i.e., the bottom quartile of accuracy among all workers in the given task).

– **Rule Breakers (RB).** These crowd workers may be *money-driven* or *entertainment-driven*. They provide mediocre responses that fall short of the expectations of a requester (eg., providing 3 keywords where 5 are required).

Rubric used to categorize RB: trustworthy workers who have high task completion times (i.e., 3^{rd} quartile of task completion times among all workers in the given task), and high accuracy (i.e., the 3^{rd} quartile of accuracy among all workers in the given task).

– **Less-competent Wokers (LW).** These crowd workers may be *money-driven* or *entertainment-driven*. They appear to have a genuine intent to complete a given task successfully by spending ample time on it, but lack the necessary skills to provide high quality responses.

Rubric used to categorize LW: trustworthy workers who have very high task completion times (i.e., $4th$ quartile of task completion times among all workers in the given task), and low accuracy (i.e., the 2^{nd} quartile of accuracy among all workers in the given task).

– **Sloppy Workers (SW).** These crowd workers may be *money-driven* or *entertainment-driven*. They complete tasks quickly and perform with an average or below average accuracy. Sloppy workers (Kazai et al., 2011) appear to err due to their speed within the task.

Rubric used to categorize SW: trustworthy workers who have very low task completion times (i.e., first quartile of task completion times among all workers

[5] Untrustworthy workers are those workers who failed to pass at least one attention check question.

in the given task), and low accuracy (i.e., the 2^{nd} quartile of accuracy among all workers in the given task).

4 Categorization of Workers

4.1 Worker Types in CC and IF Tasks

Based on the responses of individual workers in each of the 18 different tasks, 3 authors of this paper acted as experts and manually categorized workers into different classes of the worker typology presented earlier. In the 9 content creation tasks of image transcription, the overall inter-rater agreement on the expert annotations was found to be 80.1% according to percent agreement, while that in case of the 9 information finding tasks of finding middle-names was found to be 89.1%. Following a phase of discussion between the experts, the instances with disagreements were resolved in order to ensure accurate categorization of workers.

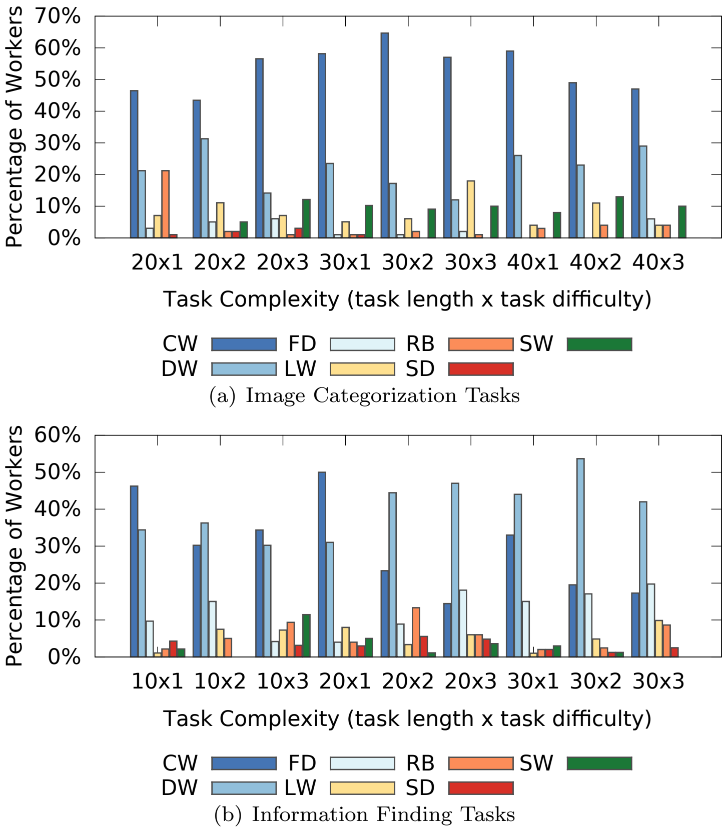
Figure 5(a) presents the distribution of different worker types based on manual annotations in the *content creation* (CC) tasks of image transcription with varying task complexity.

We note that in tasks with the length of 20 units, the percentage of sloppy workers (SW) and fast deceivers (FD) increases with an increase in task difficulty, while that of rule breakers decreases. In the tasks with length 30 units, we observe an increase in the number of less-competent workers (LW) with an increase in difficulty level. This indicates that as the complexity of a task increases, the competence or skill of a worker plays a more decisive role in the worker's performance. In tasks with a length of 30 and 40 units we note a high fraction of sloppy workers (SW) on average.

Figure 5(b) presents the distribution of different worker types based on manual annotations in the *information finding* (IF) tasks of finding middle-names with varying task complexity. We can see that with an increasing task complexity there is a decrease in the number of CW and increase in the number of DW. This indicates that complex tasks can go beyond the competence of workers and therefore workers tend to require more time to complete the task in order to perform accurately. We also note an increase in the number of FD with increasing task complexity.

Figure 6 presents the average accuracy of different types of workers and their task completion time in each of the CC tasks with varying task complexity. Across all the tasks, by definition we note that competent workers (CW) and diligent workers (DW) exhibit the highest levels of accuracy. However, CW take significantly lesser time than DW to complete the tasks ($p<.001$). We also note that with increasing task complexity, DW take more time to complete.

As shown in Table 1, less-competent workers (LW) also take a long time for task completion, but exhibit a much lower accuracy. Fast deceivers (FD) and smart deceivers (SD) exhibit lowest accuracies and task completion times



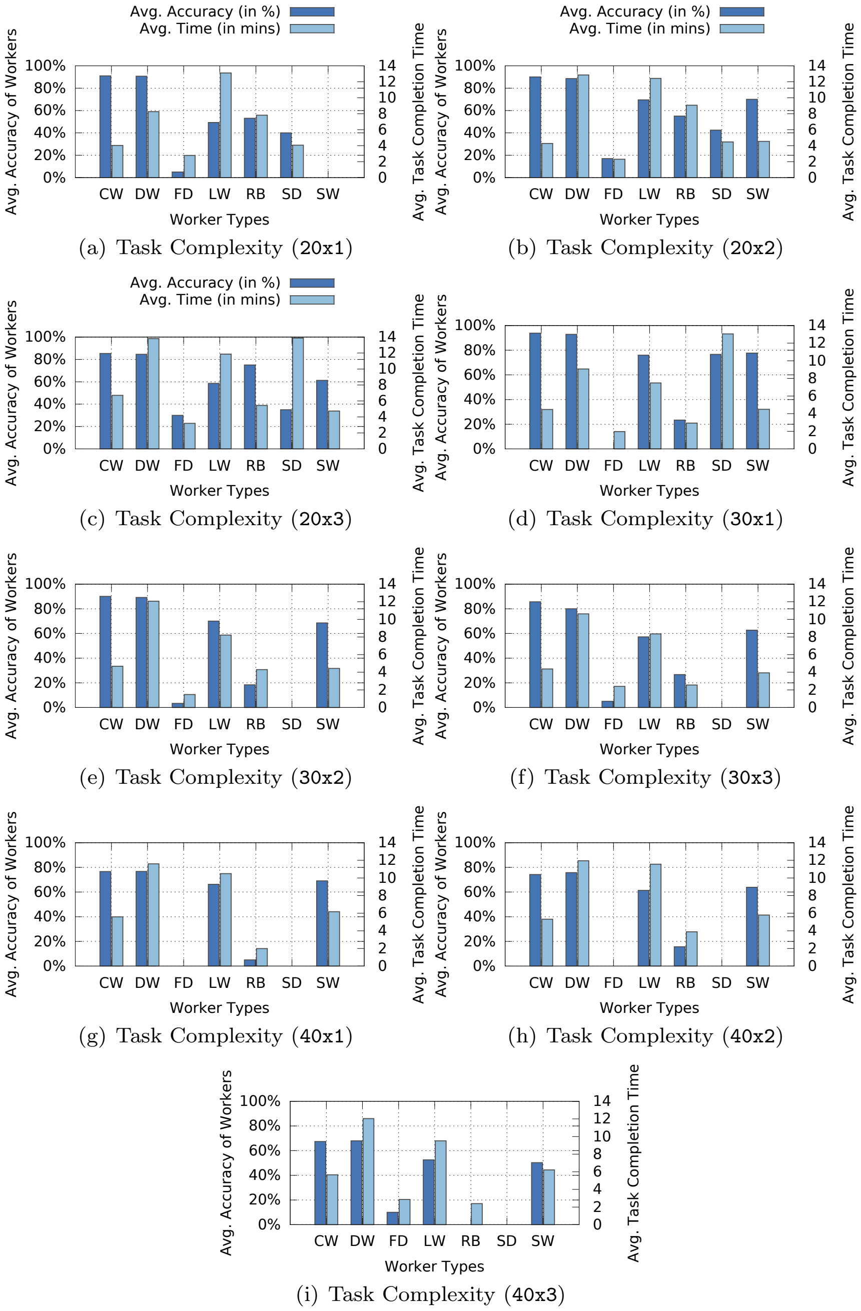
(a) Image Categorization Tasks

(b) Information Finding Tasks

Fig. 5: Distribution of worker types in the (a) content creation (CC) tasks and (b) information finding (IF) tasks, with varying task complexity. The different worker types presented here are as follows. `CW`: Competent Workers, `DW`: Diligent Workers, `FD`: Fast Deceivers, `LW`: Less-competent Workers, `RB`: Rule Breakers, `SD`: Smart Deceivers, `SW`: Sloppy Workers.

on average across all tasks, indicating their reward-focused intentions. Rule breakers (RB) perform with a low accuracy across all the CC tasks, indicative of their behavior resulting in partial responses.

Figure 7 presents the average accuracy and task completion time of different types of workers in the IF tasks with varying task complexity. Once again we notice that CW and DW exhibit the highest accuracies across the different tasks, with CW taking significantly lesser time to complete the tasks (*p<.001*). Table 2 presents the overall accuracy of different types of workers and their corresponding task completion times in the IF tasks. We observe that on average DW fractionally outperform CW (but this difference is not statistically significant). FD and SD exhibit the lowest accuracies and task completion times due to their behavior. LW spend a considerable amount of time on the tasks but fail to attain a high level of accuracy.



(a) Task Complexity (20x1)

(b) Task Complexity (20x2)

(c) Task Complexity (20x3)

(d) Task Complexity (30x1)

(e) Task Complexity (30x2)

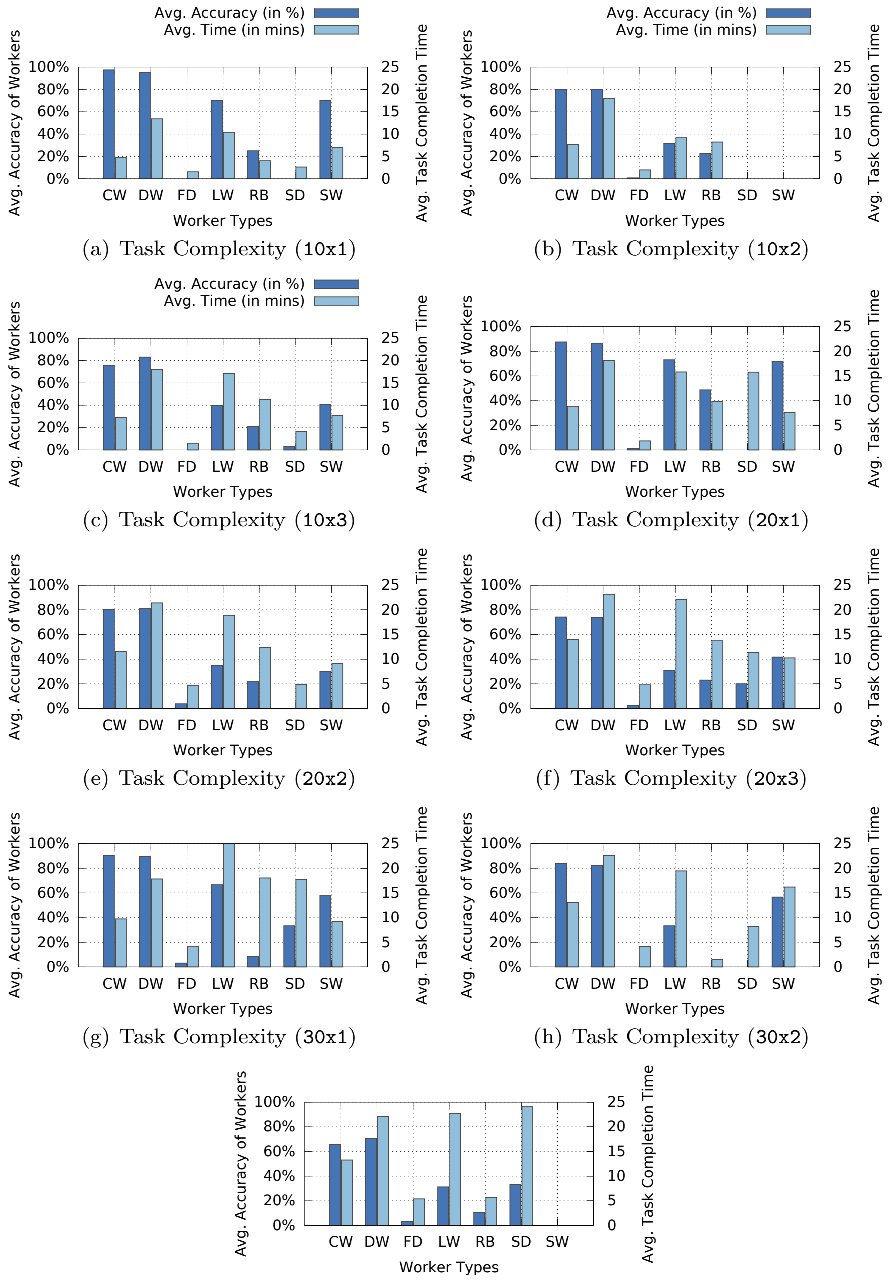
(f) Task Complexity (30x3)

(g) Task Complexity (40x1)

(h) Task Complexity (40x2)

(i) Task Complexity (40x3)

Fig. 6: Average accuracy (scaled on the y-axis) and task completion time (scaled on the y2-axis) of different types of workers in **image transcription tasks** with varying task complexity. CW: Competent Workers, DW: Diligent Workers, FD: Fast Deceivers, LW: Less-competent Workers, RB: Rule Breakers, SD: Smart Deceivers, SW: Sloppy Workers.



(a) Task Complexity (10x1)

(b) Task Complexity (10x2)

(c) Task Complexity (10x3)

(d) Task Complexity (20x1)

(e) Task Complexity (20x2)

(f) Task Complexity (20x3)

(g) Task Complexity (30x1)

(h) Task Complexity (30x2)

(i) Task Complexity (30x3)

Fig. 7: Average accuracy (scaled on the y-axis) and task completion time (scaled on the y2-axis) of different types of workers in **information finding tasks** with varying task complexity. `CW`: Competent Workers, `DW`: Diligent Workers, `FD`: Fast Deceivers, `LW`: Less-competent Workers, `RB`: Rule Breakers, `SD`: Smart Deceivers, `SW`: Sloppy Workers.

Table 1: Overall average accuracy and task completion time of different types of workers in CC tasks.

Worker Type	Avg. Acc (in %)	Avg. Time (in mins)
CW	83.79	5.01
DW	82.94	11.37
FD	7.81	1.89
LW	62.30	10.34
RB	30.23	4.49
SD	21.57	3.94
SW	59.14	4.48

Table 2: Overall average accuracy and task completion time of different types of workers in IF tasks.

Worker Type	Avg. Acc (in %)	Avg. Time (in mins)
CW	73.62	18.39
DW	75.51	26.79
FD	1.75	3.03
LW	43.28	20.27
RB	18.51	11.57
SD	10	9.57
SW	41	6.90

5 Automatic Categorization

To use the proposed worker typology in practice, in this section we present the results of an experimental evaluation of supervised machine learning models used to automatically classify worker types based on behavioral traits. Among different possible supervised models (i.e., Naive Bayes, Support Vector Machines, Neural Networks, and Decision Trees) random forest classifiers were chosen as being the best performing in terms of accuracy over an initial pilot dataset used for validation. In order to automatically categorize workers using such supervised models, we leveraged several behavioral signals: we started from a large set of signals indicating worker behavior and identified the most informative features by means of information gain. We then build decision trees (which are part of the random forest model) which place features which are most discriminative of different worker types closer to the root node. Next, we describe the final set of features used to train the model resulting from our pilot experiments.

5.1 Features Indicating Behavioral Traces

We study the mousetracking data (including keypresses) generated by crowd workers in 1,800 HITs through the 9 content creation and 9 information finding tasks, in order to determine features that can help in the prediction of a worker type. Some of the important features are presented below. A complete list of features used can be found here[6].

[6] Shortened URL - `https://goo.gl/jjv0gp`

- `time`: The task completion time of a worker.
- `tBeforeLClick`: The time taken by a crowd worker before responding to the multiple choice demographic questions in the tasks.
- `tBeforeInput`: The time taken by a crowd worker before entering a transcription in the content creation task or a middle-name in the information finding task.
- `tabSwitchFreq`: No. of times that a worker switches the tab while working on a particular task.
- `windowToggleFreq`: No. of times that a worker toggles between the current and last-viewed window while working on a particular task.
- `openNewTabFreq`: No. of times that a worker opens a new tab while working on a particular task.
- `closeCurrentTabFreq`: No. of times that a worker closes the current tab while working on a task.
- `windowFocusBlurFreq`: No. of times that the window related to the task goes in and out of focus until task completion by the crowd worker.
- `scrollUp/DownFreq`: No. of times that a worker scrolls up or down while working in a task respectively.
- `transitionBetweenUnits`: No. of times a worker moves the cursor from one unit to another in the task.
- `totalMouseMoves`: The total no. of times that a worker moves the cursor within the task.

5.2 Predicting Worker Types

By exploiting the expert annotated HITs and the features defined based on worker behavioural traces described earlier, we train and test a random forest classifier to predict worker types at the end of a completed task. We distinguish models for tasks with and without 'gold questions' (i.e., questions with known answers used to check for work quality). We study the effectiveness of our supervised models to predict worker type in CC and IF tasks with varying task complexity. Tables 3 and 4 present Accuracy and F-Measure (to account for unbalanced classes) of our supervised worker type classifiers evaluated using 10-fold cross validation over IF and CC tasks.

We can observe that it is easier to predict worker types when gold questions are available in the task. We also observe higher accuracy of automatic worker type classification for IF in comparison to CC tasks. Moreover, as *longer* tasks typically provide more behavioral signals, they lead to better automatic classification of workers in our typology. A similar conclusion can be drawn for *less difficult* tasks where worker types can be better distinguished. Due to the imbalance in the different worker types, we also ran undersampling and oversampling experiments, that yielded similar results.

Additional results from the supervised classification evaluation showed that the easiest worker types to be predicted are CW (91% accuracy) and DW (87% accuracy) for CC tasks and DW (88.7% accuracy) and FD (86.6% accuracy)

Table 3: Supervised worker type classification evaluation for IF tasks with varying task complexity.

	with Gold Questions		w/out Gold Questions	
HIT Length	**Accuracy**	**F-Measure**	**Accuracy**	**F-Measure**
10	77.3	0.748	73.6	0.679
20	74	0.701	74	0.691
30	**81.4**	**0.786**	**79.8**	**0.763**
HIT Difficulty	**Accuracy**	**F-Measure**	**Accuracy**	**F-Measure**
Level-I	**82.3**	**0.779**	**80.5**	**0.754**
Level-II	79.4	0.77	74.6	0.718
Level-III	72.3	0.691	64.2	0.587

Table 4: Supervised worker type classification evaluation for CC tasks with varying task complexity.

	with Gold Questions		w/out Gold Questions	
HIT Length	**Accuracy**	**F-Measure**	**Accuracy**	**F-Measure**
20	69.02	0.671	58.6	0.532
30	**84.5**	**0.828**	75.6	0.712
40	80.3	0.768	**78.7**	**0.729**
HIT Difficulty	**Accuracy**	**F-Measure**	**Accuracy**	**F-Measure**
Level-I	74.7	0.714	**70**	**0.643**
Level-II	**77.5**	**0.746**	67.4	0.611
Level-III	72.5	0.696	64.5	0.59

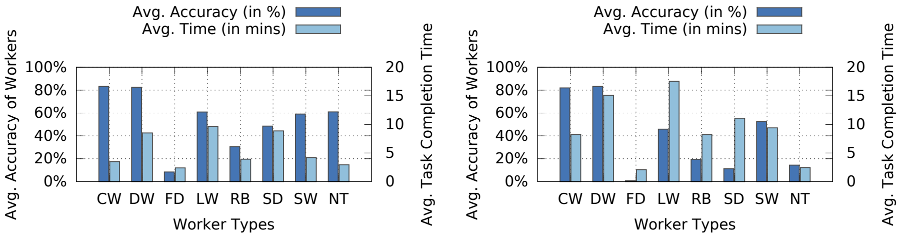
for IF tasks. Most confused worker types by our models are SW classified as CW for CC tasks and CW classified as DW for IF tasks. Feature selection by Information Gain shows that the most predictive features to automatically predict the worker type are mouse movement, windows focus frequency, the task completion time, the score, and tipping point[7] computed from gold questions (when available).

6 Evaluation and Implications

6.1 Benefits of Worker Type Information

In this section we investigate the potential benefit of automatically classifying workers as per the granular typology introduced in this paper. We analyze the average accuracy of the first 5 workers of each type who submit their responses (where the worker type is considered according to the expert annotations). In typical crowdsourcing tasks where redundancy is required, 5 judgments has been considered the norm (Vuurens and De Vries, 2012). By comparing this to the classic setting where worker type is unknown (`No Type`), i.e., the first 5 responses overall without considering worker types, we can measure the weight of worker type information.

[7] First point at which a worker provides an incorrect response after having provided at least one correct response (Gadiraju et al., 2015b).



(a) Image Transcription Tasks

(b) Information Finding Tasks

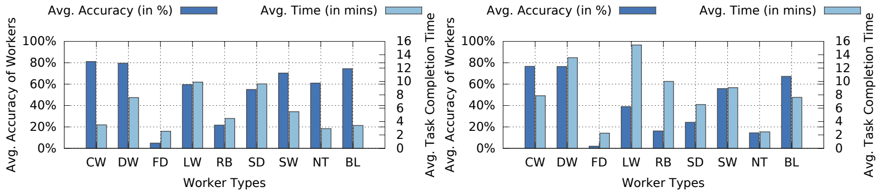
Fig. 8: Average accuracy (scaled on the y-axis) and task completion time (scaled on the y2-axis) of the first 5 judgments received from different worker types in the (a) image transcription and (b) information finding tasks. `CW`: Competent Workers, `DW`: Diligent Workers, `FD`: Fast Deceivers, `LW`: Less-competent Workers, `RB`: Rule Breakers, `SD`: Smart Deceivers, `SW`: Sloppy Workers, `NT` (No Type): First 5 judgments without considering worker type.

Figure 8 depicts the benefit of having prior knowledge of worker types. We see that in both image transcription and information finding tasks `CW` and `DW` outperform the `No Type` setting. Moreover, in case of `CW` a high level of accuracy is observed with a fairly low task completion time. This makes the competent workers (CW) preferable when compared to diligent workers (DW) who tend to take more time. We also note that the average performance of `CW` (*M=83.24, SD=8.08*) across all image transcription tasks (Fig. 8(a)) is significantly better than `No Type` (*M=60.9, SD=17.62*) with *t(8)=4.5, p<.001*. Note that the other worker types apart from `CW` and `DW` can be considered detrimental, and automatically detecting these workers is an effective way to separate them from the worker pool.

Similarly, in case of the information finding tasks (Fig. 8(b)), we note that the average performance of `CW` (*M=81.96, SD=8.33*) is significantly better than `No Type` (*M=14.7, SD=22.1*) with *t(8)=5.04, p<.001*. We allude the poor performance of `No Type` in case of the information finding tasks to the inherent task complexity of the tasks. Since these tasks require relatively more time for completion the first responses tend to be submitted by workers who complete tasks very quickly (and with low accuracy, for e.g., `FD` or `SW`). In a typical crowdsourced task, requesters finalize units when a certain number of judgments are received. Thus, we observe an adverse effect on the quality of responses in the absence of pre-selection.

6.2 Results: Automatic Worker Classification for Pre-selection

Here, we assess the impact of worker type predictions made by the proposed ML models described earlier. Once again we consider the first 5 judgments submitted by workers of each type (worker type as predicted by the classifer). We compare our proposed worker type based pre-selection method with the standard approach of using qualification tests which we refer to as the `Baseline`. In the `Baseline` method, we consider the first 5 responses from



(a) Image Transcription Tasks

(b) Information Finding Tasks

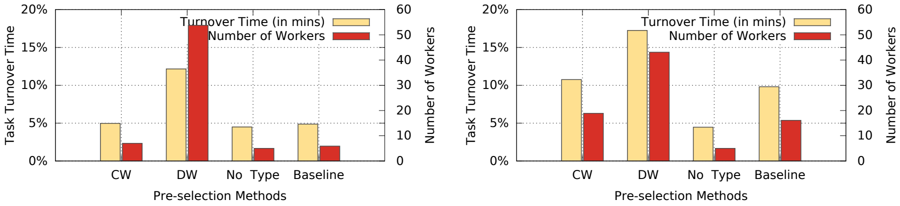
Fig. 9: Average accuracy (scaled on the y-axis) and task completion time (scaled on the y2-axis) of the first 5 judgments received from different automatically predicted worker types in the (a) image transcription and (b) information finding tasks. The different worker types presented here are as follows. `CW`: Competent Workers, `DW`: Diligent Workers, `FD`: Fast Deceivers, `LW`: Less-competent Workers, `RB`: Rule Breakers, `SD`: Smart Deceivers, `SW`: Sloppy Workers, `NT` (No Type): First 5 judgments without considering worker type, `BL` (Baseline): First 5 judgments from workers who passed the standard pre-selection test.

each worker to be a part of the qualification test. Only workers who achieve an accuracy of $\geq 3/5$ in the qualification test are considered to have passed the test. This follows our aim to replicate a realistic pre-screening scenario[8]. To compare the `Baseline` method with our proposed approach of worker type based pre-selection, we consider the first 5 judgments submitted by workers who passed the qualification test.

Figure 9 presents the results of our evaluation for the two task types. In case of the image transcription tasks (Fig. 9(a)) we note that on average across all tasks, `CW` (M=81.03, SD=8.52) significantly outperform workers in the `No Type` setting (M=60.9, SD=18.69) with $t(8)$=5.04, p<.0005. Interestingly, the task completion time (in mins) of `CW` (M=3.5, SD=0.85) is slightly more than that of `No Type` (M=2.93, SD=0.48) with $t(8)$=1.86, p<.05. `CW` also perform significantly better than the `Baseline` method (M=74.41, SD=14.06) with $t(8)$=1.86, p<.05. The differences in task completion time between `CW` and the `Baseline` method were not statistically significant, indicating that worker type based pre-selection of `CW` can outperform existing pre-selection methods in terms of quality without a negative impact on the task completion time.

For the information finding tasks (Fig. 9(b)), we note that on average across all tasks `CW` (M=76.59, SD=11.34) significantly outperform workers in the `No Type` setting (M=14.44, SD=23.6) with $t(8)$=5.04, p<.0005. In addition, we also observe that `CW` significantly outperform workers that are pre-selected using the `Baseline` method (M=67.26, SD=14.92) with $t(8)$=1.86, p<.05. The task completion time (in mins) of `CW` (M=7.87, SD=3.56) is not significantly different from that of the `Baseline` method (M=7.62, SD=3.45).

[8] CrowdFlower suggests a min. accuracy of 70% by default.



(a) Image Transcription Tasks

(b) Information Finding Tasks

Fig. 10: Task turnover time and the number of workers required for task turnover on average across all (a) image transcription and (b) information finding tasks. `CW`: Competent Workers, `DW`: Diligent Workers, `No Type`: First 5 judgments without considering worker type, `Baseline`: First 5 judgments from workers who passed the standard pre-selection test.

6.3 Task Turnover Time

The amount of time required to acquire the full set of judgments from crowd workers, thereby completing and finalizing a task considering pre-defined criteria (such as qualification tests or pre-selection) is called the *task turnover time*. We additionally evaluated the task turnover time of the different image transcription and information finding tasks when using the proposed typology-based worker pre-selection in comparison to the `Baseline` and `No Type` methods. Figure 10 presents our findings on average across all the tasks of the (a) image transcription tasks and (b) information finding tasks. We note that the average turnover time of the image transcription tasks where `CW` are pre-selected (*M=4.97, SD=1.47*) is negligibly longer than in case of the baseline method (*M=4.98 , SD=1.51*), with no statistically significant difference. These observations also hold for the information finding tasks where we did not find a significant difference between the turnover times corresponding to using the `CW` (*M=10.76 , SD=4.96*) and `Baseline` (*M=9.8, SD=4.17*) methods. Although we see that the `No Type` method results in a significantly lower turnover time when compared to `CW` with $p<.05$, as described earlier the accuracy of results when type information is not considered for pre-selection is relatively much lower. We note that the number of workers that were required before the task turnover was not significantly different between `CW` and `Baseline` methods across the different tasks in our experiments.

We present the turnover times and the number of workers required for task turnover when `DW` are pre-selected for the sake of comparison. `DW` pre-selection results in significantly higher turnover times and requires more workers for task turnover ($p<.001$). In tasks without time constraints, requesters can consider pre-selecting `DW` in addition to `CW` due to their high result accuracy.

7 Discussion and Caveats

Over the last 3 years there has been a surge in the number of new task requesters on the Amazon MTurk platform (over 1,000 new requesters per month) (Difallah et al., 2015). Tasks designed by less experienced requesters can be easy targets for fast deceivers (FD) and smart deceivers (SD) alike. In this paper, we have shown that FD and SD take the least amount of time to provide responses despite of task complexity. There are two adverse effects of this behavior; (i) FD and SD can access a lot of available work that is susceptible to their behavior in the marketplace due to their quick task completion times. (ii) Due to the fact that responses provided by FD and SD cannot be easily distinguished from genuine workers on the fly, requesters accept the validity of their responses, thereby depriving other more suitable workers from participating in the task. Requesters thus face the dual-curse of getting sub-optimal returns for their investment in terms of response quality, and would require to deploy the tasks once again on discovering poor quality through a post-hoc analysis. In this context, automated pre-selection of workers based on their behavior, as proposed in this paper can help requesters in improving their costs-benefits ratio while assuring the reliability and speed of produced results.

We also investigated the effect of task complexity on worker behavior. From our experiments, we found that with increasing task complexity the fraction of underperforming workers increases. In complex tasks it is therefore all the more important to pre-select workers who are capable of performing accurately as exhibited by competent and diligent workers (`CW`, `DW`).

The importance of distinguishing between `CW` and `DW` is realized when requesters need to account for cost-bound constraints (time, money). In such cases `CW` are more desirable. Although workers of other task types are found to be detrimental, detecting each type of workers can facilitate personalized feedback and training that can improve the overall effectiveness of crowd work in the long run. Thus, we argue in favor of the typology-based prediction and pre-selection of workers, more so in tasks with high complexity due to the clear benefits in quality. At the same time, the automated detection of worker types provides an opportunity to identify less-competent workers `LW` and help them improve their performance. Prior works have shown that providing feedback to workers regarding their performance and helping them to reflect on instances where they were wrong or provided suboptimal responses, allows workers to improve their performance (Dow et al., 2012; Gadiraju et al., 2015a; Taras, 2002). Thus, by automatically detecting less-competent workers (LW), one can provide additional training and feedback to these workers and help them improve their performance. In this way, a less-competent worker can become more competent overtime after acquiring sufficient support, preventing his/her alienation in the crowd through automatic classification.

Finally, in our previous work we found that the device type of workers can potentially influence their performance in crowdsourced microtasks (Gadiraju et al. (2017a)). In similar settings, we found less than 7% of workers to be using mobile devices. Coupled with the high number of distinct workers who completed our tasks in each task configuration, we believe the device type would not have a significant impact on the overall comparison in the analysis presented in this work.

8 Conclusions and Future Work

We collected worker activity data in 1,800 HITs with varying length, type, and difficulty. We refined the existing understanding of worker types and extended it by considering the dimensions of motivation, performance and behavior within a *worker typology*.

We experimentally showed that it is possible to automatically classify workers into granular classes based on supervised machine learning models that use behavioral traces of workers completing HITs. Leveraging such worker type classification, we can improve the quality of crowdsourced tasks by preselecting workers for a given task. Thus, we found that crowd worker behavioral traces can be leveraged to classify workers in a fine-grained worker typology that can be used for better worker pre-selection (**RQ#1**).

We modeled task complexity and studied the impact of task complexity on worker behavior across two different task types; content creation tasks and information finding tasks. Based on our experiments and results we have shown that pre-selection based on worker types significantly improves the quality of the results produced, especially in tasks with high complexity (**RQ#2**).

For image transcription tasks our method yielded an accuracy increase of nearly 7% over the baseline and of almost 10% in information finding tasks, without a significant difference in task completion time of workers (**RQ#3**). Since our approach is based on gathering behavioral signals from a worker during the pre-screening phase, no prior information about a worker is required. This has important implications on structuring workflow.

In this paper, we highlighted clear benefits of distinguishing beyond *good* and *bad* workers in image transcription and information finding tasks. This is not just useful for requesters to attain better and faster results from crowdsourcing platforms but can also be leveraged to support crowd workers by helping them to understand their performance and contributions better, and improve over time.

Prior work has discussed that requesters should consider the context in which workers are embedded while contributing work in online labour markets (Martin et al., 2014; Gadiraju and Gupta, 2016). The work environments may not always be appropriate, and the devices that workers use to complete tasks may not be ergonomically suitable. Recent work has brought to light the influence of task clarity on the quality of work that is produced (Gadiraju et al., 2017c). Supporting such previous works that reflect on the wide landscape of quality in crowdsourced microtasks, our results show clear benefits in automatically typecasting workers in the pre-selection phase. However, employing such mechanisms should not alienate or discriminate against less-competent workers (LW). On the contrary, such workers should be supported in a manner that allows them to learn and transform into more effective and capable contributors (Dow et al., 2012). Power asymmetry between workers and requesters in crowdsourcing marketplaces has been acknowledged as an issue, and addressed by recent works (Irani and Silberman, 2013; Gaikwad et al., 2016). Thus, it is important to consider other factors that promote fairness and transparency in the marketplace. Aiding, helping and training workers to learn and improve their performance in microtasks (Gadiraju et al., 2015a; Gadiraju and Dietze, 2017) can have a positive impact on the mutual trust between workers and task requesters. Our results suggest that there is a need to support workers so that they become more effective and efficient (especially those who complete tasks while exerting genuine effort, such as the less-competent workers). One way to achieve this is to provide constructive feedback to workers who do not pass the pre-selection phase.

In the imminent future, we will also investigate the use of worker behavioral analytics to support workers in crowdsourcing tasks. We will also evaluate the use of worker type based pre-selection in other types of crowdsourcing tasks.

References

Berg, Bruce Lawrence (2004). *Methods for the social sciences. Qualitative Research Methods for the Social Sciences.* Boston: Pearson Education.

Bozzon, Alessandro; Marco Brambilla; Stefano Ceri; Matteo Silvestri; and Giuliano Vesci (2013). Choosing the Right Crowd: Expert Finding in Social Networks. *EDBT'13. Joint 2013 EDBT/ICDT Conferences, Proceedings of the 16th International Conference on Extending Database Technology, Genoa, Italy, March 18-22, 2013.* New York: ACM Press, pp. 637–648.

Cheng, Justin; Jaime Teevan; Shamsi T Iqbal; and Michael S Bernstein (2015). Break it down: A comparison of macro-and microtasks. *CHI'15. Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, Seoul, Republic of Korea, April 18-23, 2015.* New York: ACM Press, pp. 4061–4064.

Dang, Brandon; Miles Hutson; and Matthew Lease (2016). MmmTurkey: A Crowdsourcing Framework for Deploying Tasks and Recording Worker Behavior on Amazon Mechanical Turk. *HCOMP'16. Proceedings of the 4th AAAI Conference on Human Computation and Crowdsourcing (HCOMP): Works-in-Progress Track, Austin, Texas, USA, 30 October-3 November, 2016.* AAAI Press, pp. 1–3.

Demartini, Gianluca; Djellel Eddine Difallah; and Philippe Cudré-Mauroux (2012). ZenCrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. *WWW'12. Proceedings of the 21st World Wide Web Conference 2012, Lyon, France, April 16-20, 2012.* New York: ACM Press, pp. 469–478.

Denzin, Norman K (1978). *The research act: A theoretical orientation to sociological methods*, Vol. 2. New York: McGraw-Hill.

Difallah, Djellel Eddine; Gianluca Demartini; and Philippe Cudré-Mauroux (2013). Pick-a-crowd: tell me what you like, and i'll tell you what to do. *WWW'13. Proceedings of the 22nd International World Wide Web Conference, Rio de Janeiro, Brazil, May 13-17, 2013.* New York: ACM Press, pp. 367–374.

Difallah, Djellel Eddine; Michele Catasta; Gianluca Demartini; Panagiotis G Ipeirotis; and Philippe Cudré-Mauroux (2015). The dynamics of micro-task crowdsourcing: The case of amazon mturk. *WWW'15. Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, May 18-22, 2015.* New York: ACM Press, pp. 238–247.

Dow, Steven; Anand Kulkarni; Scott Klemmer; and Björn Hartmann (2012). Shepherding the crowd yields better work. *CSCW'12. Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work, Seattle, WA, USA, February 11-15, 2012.* New York: ACM Press, pp. 1013–1022.

Eckersley, Peter (2010). How unique is your web browser? *PETS'10. Proceedings of the 10th International Symposium on Privacy Enhancing Technologies Symposium, Berlin, Germany, July 21 - 23, 2010.* Heidelberg: Springer, pp. 1–18.

Eickhoff, Carsten; Christopher G Harris; Arjen P de Vries; and Padmini Srinivasan (2012). Quality through flow and immersion: gamifying crowdsourced relevance assessments. *SIGIR'12. Proceedings of the 35th International ACM SIGIR conference on research and development in Information Retrieval, Portland, OR, USA, August 12-16, 2012.* New York: ACM Press, pp. 871–880.

Feyisetan, Oluwaseyi; Elena Simperl; Max Van Kleek; and Nigel Shadbolt (2015)a. Improving paid microtasks through gamification and adaptive furtherance incentives. *WWW'15. Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, May 18-22, 2015.* New York: ACM Press, pp. 333–343.

Feyisetan, Oluwaseyi; Markus Luczak-Roesch; Elena Simperl; Ramine Tinati; and Nigel Shadbolt (2015)b. Towards hybrid NER: a study of content and crowdsourcing-related performance factors. *ESWC'15. Proceedings of The Semantic Web. Latest Advances and New Domains - 12th European Semantic Web Conference, Portoroz, Slovenia, May 31 - June 4, 2015.* Heidelberg: Springer, pp. 525–540.

Gadiraju, Ujwal; Alessandro Checco; Neha Gupta; and Gianluca Demartini (2017)a. Modus operandi of crowd workers: The invisible role of microtask work environments. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)*, vol. 1, no. 3, pp. 49:1–49:29.

Gadiraju, Ujwal; Besnik Fetahu; Ricardo Kawase; Patrick Siehndel; and Stefan Dietze (2017)b. Using worker self-assessments for competence-based preselection in crowdsourcing microtasks. *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 24, no. 4, pp. 30:1–30:26.

Gadiraju, Ujwal; Besnik Fetahu; and Ricardo Kawase (2015)a. Training workers for improving performance in crowdsourcing microtasks. *EC-TEL'15. Design for Teaching and Learning in a Networked World - Proceedings of the 10th European Conference on Technology Enhanced Learning, Toledo, Spain, September 15-18, 2015.* Heidelberg: Springer, pp. 100–114.

Gadiraju, Ujwal; Jie Yang; and Alessandro Bozzon (2017)c. Clarity is a Worthwhile Quality – On the Role of Task Clarity in Microtask Crowdsourcing. *HT'17. Proceedings of the 28th ACM Conference on Hypertext and Social*

Media, Prague, Czech Republic, July 4-7, 2017. New York: ACM Press, pp. 5–14.

Gadiraju, Ujwal; Ricardo Kawase; Stefan Dietze; and Gianluca Demartini (2015)b. Understanding malicious behavior in crowdsourcing platforms: The case of online surveys. *CHI'15. Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, Seoul, Republic of Korea, April 18-23, 2015.* New York: ACM Press, pp. 1631–1640.

Gadiraju, Ujwal; Ricardo Kawase; and Stefan Dietze (2014). A taxonomy of microtasks on the web. *HT'14. Proceedings of the 25th ACM Conference on Hypertext and Social Media, Santiago, Chile, September 1-4, 2014.* New York: ACM Press, pp. 218–223.

Gadiraju, Ujwal; and Neha Gupta (2016). Dealing with Sub-optimal Crowd Work: Implications of Current Quality Control Practices. *International Reports on Socio-Informatics (IRSI), Proceedings of the CHI 2016 - Workshop: Crowd Dynamics: Exploring Conflicts and Contradictions in Crowdsourcing,* Vol. 13. pp. 15–20.

Gadiraju, Ujwal; and Ricardo Kawase (2017). Improving Reliability of Crowdsourced Results by Detecting Crowd Workers with Multiple Identities. *ICWE'17. Proceedings of the 17th International Conference, Rome, Italy, June 5-8, 2017.* Heidelberg: Springer, pp. 190–205.

Gadiraju, Ujwal; and Stefan Dietze (2017). Improving Learning Through Achievement Priming in Crowdsourced Information Finding Microtasks. *LAK'17. Proceedings of the Seventh International Learning Analytics & Knowledge Conference, Vancouver, BC, Canada, March 13-17, 2017.* New York: ACM Press, pp. 105–114.

Gaikwad, Snehalkumar Neil S; Durim Morina; Adam Ginzberg; Catherine Mullings; Shirish Goyal; Dilrukshi Gamage; Christopher Diemert; Mathias Burton; Sharon Zhou; Mark Whiting; et al. (2016). Boomerang: Rebounding the consequences of reputation feedback on crowdsourcing platforms. *UIST'16. Proceedings of the 29th Annual Symposium on User Interface Software and Technology, Tokyo, Japan, October 16-19, 2016.* New York: ACM Press, pp. 625–637.

Ipeirotis, Panagiotis G; Foster Provost; and Jing Wang (2010). Quality management on amazon mechanical turk. *HCOMP'10. Proceedings of the ACM SIGKDD workshop on Human Computation.* New York: ACM Press, pp. 64–67.

Irani, Lilly C; and M Silberman (2013). Turkopticon: Interrupting worker invisibility in amazon mechanical turk. *CHI'13. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Paris, France, April 27 - May 2, 2013.* New York: ACM Press, pp. 611–620.

Kazai, Gabriella; Jaap Kamps; and Natasa Milic-Frayling (2011). Worker types and personality traits in crowdsourcing relevance labels. *CIKM'11. Proceedings of the 20th ACM International Conference on Information and Knowledge Management, Glasgow, United Kingdom, October 24-28, 2011.* New York: ACM Press, pp. 1941–1944.

Kazai, Gabriella; Jaap Kamps; and Natasa Milic-Frayling (2012). The face of quality in crowdsourcing relevance labels: Demographics, personality and labeling accuracy. *CIKM'12. Proceedings of the 21st ACM International conference on Information and Knowledge Management, Maui, HI, USA, October 29 - November 02, 2012.* New York: ACM Press, pp. 2583–2586.

Kazai, Gabriella; Jaap Kamps; and Natasa Milic-Frayling (2013). An analysis of human factors and label accuracy in crowdsourcing relevance judgments. *Information retrieval*, vol. 16, no. 2, pp. 138–178.

Kazai, Gabriella; and Imed Zitouni (2016). Quality Management in Crowdsourcing Using Gold Judges Behavior. *WSDM'18. Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, San Francisco, CA, USA, February 22-25, 2016.* New York: ACM Press, pp. 267–276.

Kittur, Aniket; Jeffrey V Nickerson; Michael Bernstein; Elizabeth Gerber; Aaron Shaw; John Zimmerman; Matt Lease; and John Horton (2013). The future of crowd work. *CSCW'13. Proceedings of the 16th ACM Conference on Computer Supported Cooperative Work, San Antonio, TX, USA, February 23-27, 2013.* New York: ACM Press, pp. 1301–1318.

Marshall, Catherine C; and Frank M Shipman (2013). Experiences surveying the crowd: Reflections on methods, participation, and reliability. *Proceedings of the 5th Annual ACM Web Science Conference.* pp. 234–243.

Martin, David; Benjamin V Hanrahan; Jacki O'Neill; and Neha Gupta (2014). Being a Turker. *CSCW'14. Proceedings of the 17th ACM conference on Computer Supported Cooperative Work & Social Computing, Baltimore, MD, USA, February 15-19, 2014.* New York: ACM Press, pp. 224–235.

Oleson, David; Alexander Sorokin; Greg P. Laughlin; Vaughn Hester; John Le; and Lukas Biewald (2011). Programmatic Gold: Targeted and Scalable Quality Assurance in Crowdsourcing. *HCOMP'11. Papers from the 2011 AAAI Workshop on Human Computation, San Francisco, California, USA, August 8, 2011.* AAAI Press, pp. 43–48.

Rokicki, Markus; Sergej Zerr; and Stefan Siersdorfer (2015). Groupsourcing: Team competition designs for crowdsourcing. *WWW'15. Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, May 18-22, 2015.* New York: ACM Press, pp. 906–915.

Rzeszotarski, Jeffrey; and Aniket Kittur (2012). CrowdScape: interactively visualizing user behavior and output. *UIST'12. Proceedings of the he 25th Annual ACM Symposium on User Interface Software and Technology, Cambridge, MA, USA, October 7-10, 2012.* New York: ACM Press, pp. 55–62.

Rzeszotarski, Jeffrey M; and Aniket Kittur (2011). Instrumenting the crowd: using implicit behavioral measures to predict task performance. *UIST'11. Proceedings of the 24th annual ACM symposium on User Interface Software and Technology, Santa Barbara, CA, USA, October 16-19, 2011.* New York: ACM Press, pp. 13–22.

Sheshadri, Aashish; and Matthew Lease (2013). SQUARE: A Benchmark for Research on Computing Crowd Consensus. *HCOMP'13. Proceedings of the First AAAI Conference on Human Computation and Crowdsourcing, November 7-9, 2013, Palm Springs, CA, USA.* AAAI Press, pp. 156–164.

Strauss, Anselm; and Barney Glaser (1967). *Discovery of grounded theory.* Chicago: Aldine.

Strauss, Anselm L (1987). *Qualitative analysis for social scientists.* Cambridge: Cambridge University Press.

Taras, Maddalena (2002). Using assessment for learning and learning from assessment. *Assessment & Evaluation in Higher Education*, vol. 27, no. 6, pp. 501–510.

Venanzi, Matteo; John Guiver; Gabriella Kazai; Pushmeet Kohli; and Milad Shokouhi (2014). Community-based bayesian aggregation models for crowdsourcing. *WWW'14. Proceedings of the 23rd International World Wide Web Conference, Seoul, Republic of Korea, April 7-11, 2014.* New York: ACM Press, pp. 155–164.

Vuurens, Jeroen BP; and Arjen P De Vries (2012). Obtaining high-quality relevance judgments using crowdsourcing. *IEEE Internet Computing*, vol. 16, no. 5, pp. 20–27.

Wang, Jing; Panagiotis G Ipeirotis; and Foster Provost (2011). Managing crowdsourcing workers. *WCBI'11. Proceedings of the Winter Conference on Business Intelligence, Salt Lake City, Utah, USA, March 12-14, 2011.* Citeseer, pp. 10–12.

Wood, Robert E (1986). Task complexity: Definition of the construct. *Organizational Behavior and Human Decision Processes*, vol. 37, no. 1, pp. 60–82.

Yang, Jie; Judith Redi; Gianluca Demartini; and Alessandro Bozzon (2016). Modeling Task Complexity in Crowdsourcing. *HCOMP'16. Proceedings of the Fourth AAAI Conference on Human Computation and Crowdsourcing, Austin, Texas, USA, 30 October-3 November, 2016.* AAAI Press, pp. 249–258.