Worker communities in online crowdsourcing markets

Master Thesis

C.P. van der Valk

Technische Universiteit Delft

Delft University of Technology

Challenge the future

Worker communities in online crowdsourcing markets

Master Thesis

by

C.P. van der Valk

in partial fulfillment of the requirements for the degree of

Master of Science
in Computer Science
Data Science and Technology Track

at the Delft University of Technology,
to be defended publicly on Monday January 30, 2017 at 15:00 PM.

Supervisors:	Dr. A. Bozzon	
	Dr. J. Redi	
Thesis committee:	Prof. dr. ir. GJ. Houben,	TU Delft
	Dr. A. Bozzon,	TU Delft
	Dr. J. Redi,	TU Delft
	Dr. P. Pawelczak,	TU Delft

This thesis is confidential and cannot be made public until January 31, 2017.

Delft University of Technology

Worker communities in online crowdsourcing markets

Author: Carlo van der Valk
Student id: 1308378
Email: cvandervalk@gmail.com

Abstract

Crowdsourcing Internet marketplaces such as Amazon Mechanical Turk are platforms where *human intelligence tasks* are posted by *requesters*, which *workers* can then choose to perform for a reward. It was assumed that workers choose and perform tasks independently, but recent work has shown that workers tend to collaborate and socialize by forming online communities. This thesis researches these online worker communities and their relationship with the crowdsourcing market. In order to perform this research we created a tool that gathers data from a crowdsourcing platform called Amazon Mechanical Turk (mTurk) and several worker community fora. With the aim of gaining a better understanding of worker communities we first studied worker community discussions on the fora to determine what types of discussions take place and by whom. The influence of the crowdsourcing market on worker community activity was determined to see what triggers discussion among workers. To see if workers communities can lead to change on the crowdsourcing market we determined the impact of worker discussions on tasks and requesters. Result of our analysis show that worker communities have a reciprocal relationship with the crowdwork market, which allowed us to create guidelines for the design crowdwork platforms and tasks.

Thesis Committee:

Chair:	Prof. dr. ir. G.J. Houben, Faculty EEMCS, TUDelft
University Supervisor:	Dr. A. Bozzon, Faculty EEMCS, TUDelft
University Supervisor:	Dr. J. Redi, Faculty EEMCS, TUDelft
Committee Member:	Dr. P. Pawelczak, Faculty EEMCS, TUDelft

PREFACE

This report documents my research of crowdsourcing and crowdworker communities. During my research of these topics I was able to test my technical and analytical skills, and learn some new skills along the way. This is due in no small part to the help I received along the way from supervisors, colleagues, and friends.

First I would like to thank Geert-Jan Houben for helping me choose this topic and putting me in contact with the right people. I would like to thank my supervisors, Alessandro Bozzon and Judith Redi, for their support, feedback, and presenting me with many exiting ideas and challenges during my Master's thesis project. I would also like to thank Jie Yang for his valuable help, feedback, and guidance. My gratitude also goes to Przemysław Pawełczak for being part of my thesis committee.

C.P. van der Valk
Delft, The Netherlands
January, 2017

Contents

1

Introduction

1.1. Motivation

The growth of the Internet provides many new possibilities and opportunities for obtaining and processing information. One data processing approach that has proven increasingly valuable in recent years is *crowdsourcing*. In crowdsourcing, a group of people operate Web tools to perform small or simple tasks, typically in exchange of small monetary rewards. By having people performing tasks that require human intelligence that computers cannot (or not yet) perform or complete reliably, complex problem can be solved. The employment of human processing power in tasks that are trivial to humans is known as *human computation*. Human computations tasks are often referred to as *Human Intelligence Tasks* (HITs). Some examples of such tasks are surveys, image annotation, and transcription of audio. Amazon Mechanical Turk (mTurk) is the most known example of human computation marketplace.

Literature about human computation assumes workers to anonymously and independently operate in online marketplaces. However recent studies have challenged these assumptions [1] [2], and show that workers form communities in Web fora to discuss and collaborate outside of such marketplaces.

In the case of Amazon Mechanical Turk, there exists a crowd community spread over different online fora; Turkopticon, TurkerNation, mTurkForum, mTurkCrowd, mTurkGrind and RedditHWTF(subreddit "Hits Worth Turking For"). These fora are known to be popular among workers [3], have different goals, such as providing a safeguard for workers, recommendations of tasks, and sharing of worker experiences and advice.

The existence and success of these fora show how the social component of human computation is central to a better understanding of the discipline. This consideration guided this work, which studies the community aspect of online worker communities, to better understand the dynamics within crowd fora, and their relationship with the human computation marketplaces.

1.2. Research Objectives

This thesis seeks answer to the following research questions:

- **RQ1**: *Which aspects of crowdwork are discussed in online fora, and by whom?*

  Observing the topics of the discussions by the workers is important for understanding the worker communities and the different community platforms. It allows us to determine the activities and organisation of worker communities.

- **RQ2**: *How is the activity of crowd worker communities in online fora influenced by the content (i.e. tasks) and by the dynamics (i.e. trends) of crowdsourcing markets?*

  Determining which aspects of the crowdsourcing market triggers discussion by the workers can provide some new insights into worker community activities.

- **RQ3**: *Does the activity of crowd workers in online fora influence the dynamics of crowdsourcing markets?*

  We hypothesise the existence of a relationship between discussions by the worker communities, and changes in the status of the crowdwork market. For instance, tasks could be completed more quickly, or more effectively, as a result of discussion on community platforms.

- **RQ4**: *Do ratings, reviews, and feedback from workers influence the behaviour of requesters?*

  We would like to determine if requesters are influenced by worker discussions and incorporate them as feedback when creating new tasks. Analysing requesters behaviour over time compared to worker evaluation might indicate a type of workers' influence. For instance, this could manifest in changes to task properties (e.g. reward, allotted completion time) that are due to feedbacks and discussions from the community.

1.3. CONTRIBUTIONS

Figure 1.1 shows how the socio-technical system under study. We adopted a data-driven approach, exploiting a novel *mTurk Market and Community Tracker* to extended an existing dataset [4] with more recent data extracted from mTurk, and with discussions crawled from crowdworker fora. The analysis on the dataset provided novel insights, that we collected and summarised in a set of guidelines for crowd-work design.

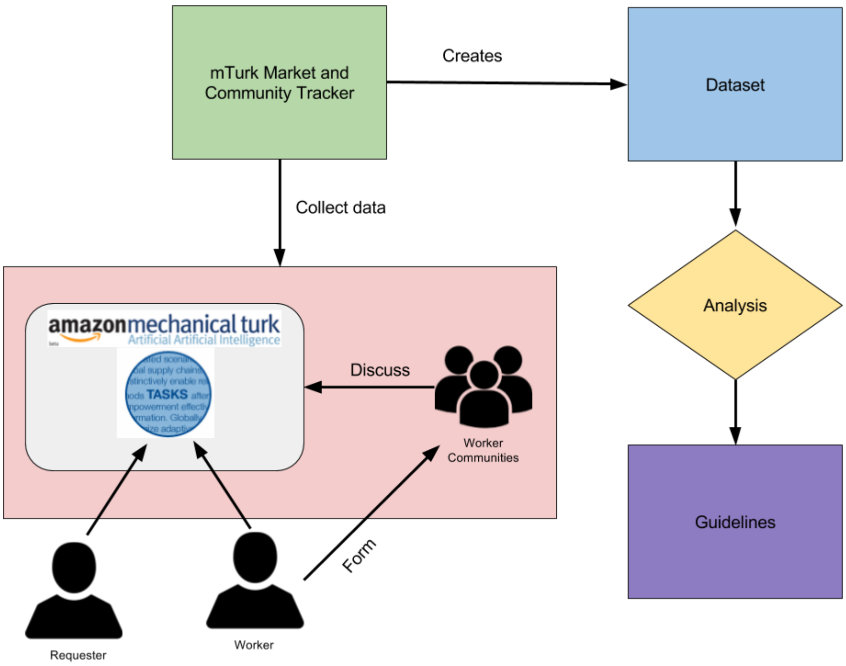


Figure 1.1: Model with the main activities and contributions of our work

The original contributions of this thesis are summarised below:

mTurk market & community tracker: We were able to construct a crawling platform using previous works of Panos Ipeirotis, who constructed the crawling platform that collects the data from the mTurk market. Furthermore, we expanded the tracker to also crawl worker community fora to collect worker discussion data.

Chapter 4 provides detail about the tracker architecture.

New dataset generated by our tracker: Our new tracker (mTurk market & community tracker) provided us with a new dataset that supplements the dataset already available. This dataset contains mTurk market data as well as message data from six worker community fora. The market data in this dataset was collected in a 1-month period between Apr. 11 and May 20 2016.

Chapter 5 starts with providing a statistical overview of the collected data.

Findings dynamics between market and community: Performing our quantitative analysis on the dataset provided us with answers to the research questions and new insights, which detail the dynamics between the mTurk market and worker communities. In short, by linking the crowdwork market data and worker community data we were able to study the nature of discussions, and measure the influence of the market on discussions and the impact of the discussions on the market.

In Chapter 3 we discuss an abstract overview of the data and conceptual model used when performing our study. And, in Chapter 5 we provide the analysis that were performed in our study that lead to our findings.

Guidelines: From our analysis results we were able to create guidelines for the creation of crowdwork platforms and crowdwork tasks. The guidelines are aimed at exploiting the relationship between the crowdwork platform and their communities to provide better exposure and throughput of HITs.

These guidelines can be found in Chapter 6.

1.4. THESIS OUTLINE

The remainder of this thesis is as follows. Chapter 2 starts with providing the explanation and background of crowdsourcing, the crowdsourcing platform mTurk, and worker community fora. Then we look at the previous work related market dynamics and crowd study. Chapter 3 starts with providing an overview of data that can be collected from mTurk and the crowd communities. Followed by a new conceptual model that explains the relationships between the data sources and how these can be studied. Chapter 4 provides how our *mTurk Market and Community Tracker* was constructed and functions. Chapter 5 starts with a statistical overview of our dataset. Then, we analyse the dataset to answer each research question. Chapter 6 provides some conclusions on our findings as well as some discussion on the possibilities of future work.

2

Background and Previous Work

In this chapter we introduce background knowledge about crowdsourcing. After a general introduction about crowdsourcing and its mains concepts, we describe the Amazon Mechanical Turk platform and the function and properties of several crowdsourcing communities and fora. Finally, we look at previous work relevant to the study of crowdsourcing, and the study of online communities.

2.1. Crowdsourcing

Crowdsourcing is defined as the activity performed by people(requesters) that outsource tasks to other people (workers) by means of online mediation tools, and for small monetary compensation. The term crowdsourcing was coined by Jeff Howe [5] and was described as using the Internet to "outsource work to the crowd". Crowdsourcing platforms facilitate the publishing and execution of crowdsourcing tasks. Tasks on crowdsourcing platforms include work that cannot, or not yet, be performed by machines or software. These types of tasks can range from picture annotation, survey, or even transcription of audio, and can take a few minutes to hours to complete. On crowdsourcing platforms these tasks are referred to as Human Intelligence Tasks, or HITs for short. Crowdsourcing gained great popularity in recent years [6], and fostered the creation of several online crowdwork marketplaces like Amazon Mechanical Turk.

2.1.1. Amazon Mechanical Turk

Amazon Mechanical Turk (mTurk or AMT), launched in 2005, has grown into one of the most popular Crowdsourcing platforms on the Internet. Workers on this platform are commonly referred to as Turkers. On mTurk tasks are posted by requesters in *HIT Groups* and a *HIT* refers to a single execution of such a task by a worker. Requesters provide for each HIT Group a description of the task which contains the type of work, reward in dollars and cents, time allotted for performing the task in minutes, expiration date, HITs available (batch size of the HIT Group), and qualifications for workers. A Turker, a registered and signed in worker on mTurk, is provided a selection of tasks athat are available. Workers can then accept a tasks, perform and submit their work. The requester then approves the work which results in the compensation being provided to the worker.

Figure 2.1 shows the conceptual model of how mTurk functions. It shows the basic principle crowdsourcing actors and their relationships. Requesters publish tasks on mTurk, these tasks are referred to as Human Intelligence Task Groups or HIT Group. The execution of a HIT Group is called a HIT. HIT Groups have one or more HITs. Workers perform HITs on mTurk to earn monetary reward. The requester approves or rejects the results of a HIT, and provides the reward when the work is approved.

Figure 2.2 shows how workers view HIT Groups on the mTurk market. From this figure we can observe that HIT Groups show the requester name, the task title, when the task expires, reward, and how much time requesters have allotted for workers to perform the task.

One interesting phenomenon that has occurred with the growth of this crowdsourcing platform is the formation and growth of a community of workers online in fora such as Turkopticon and mTurkForum for example. It was believed that workers are independent and autonomous when choosing and performing tasks, but recent work has challenged these assumptions and show that workers collaborate outside of mTurk using worker community platforms [1] [2].

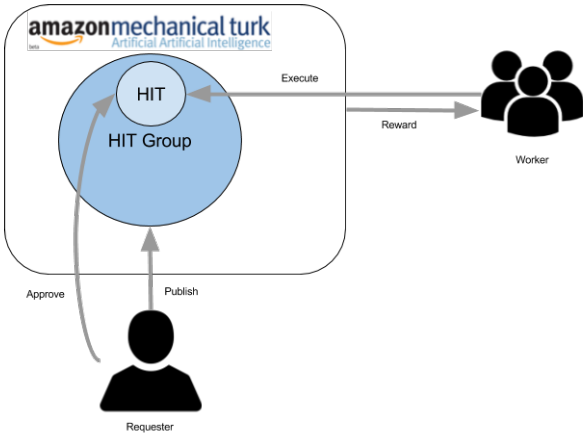


Figure 2.1: Model of crowdwork on Amazon Mechanical Turk

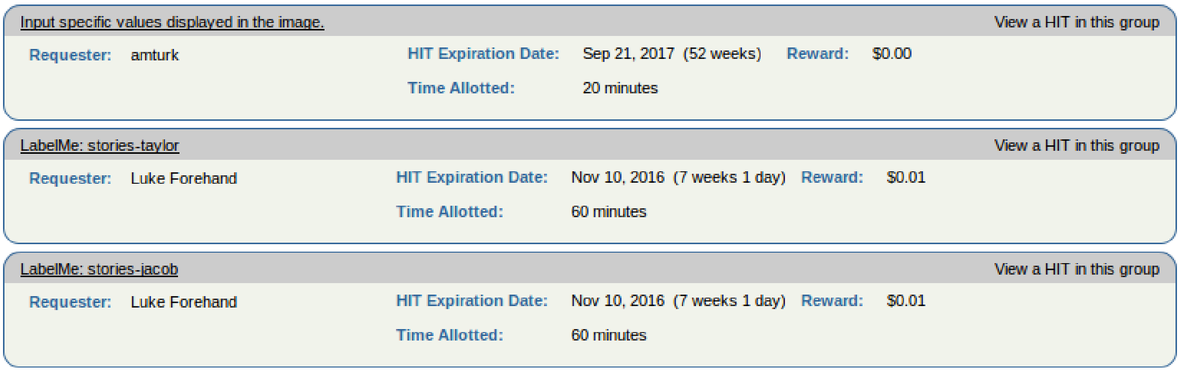


Figure 2.2: HIT Groups on Amazon Mechanical Turk

2.1.2. MTURK TRACKER

Because mTurk has grown into such a popular crowdsourcing platform, it has attracted attention of many researchers. One result of this is the *mTurk Tracker* developed by Panos Ipeirotis [1], which gathers data about tasks and requesters from mTurk. It is a valuable tool for studying the crowdsourcing platform market and the source code is available on GitHub [2]. Panos Ipeirotis published two papers on crowdsourcing [7] [4] using the mTurk tracker. The *mTurk Tracker* functions by periodically collecting data of the visible HIT Groups on the mTurk market and following the progress of active HIT Groups present in its database. *mTurk Tracker* has a front-end web component that allows for interactive visualizations of the gathered data in an Internet browser. The source code of the mTurk tracker linked here is a version that requires Google App Engine services to run, which provides a server and data storage for the tracker platform.

2.2. CROWDSOURCING COMMUNITIES

Workers interact with one another using online fora, which now host large communities. Workers began discussing and recommending tasks and requesters in many different ways. Communication between workers ranges from advice on how to earn money efficiently, recommending tasks, to warning each other of bad requesters or tasks. Some of the most popular online communities for mTurk are; Turkopticon, TurkerNation, mTurkCrowd, mTurkForum, mTurkGrind, and RedditHWTF(the subreddit HitsWorthTurkingFor). Not all of these communities have the exact same structure, but they do have many things in common as community platforms. All of them facilitate discussion about crowdwork in some way, but can have different intentions(mission statements). For example, Turkopticon aims to raise awareness of the unaccountability of requesters, while TurkerNation provides discussion and recommendations of tasks to the benefit of workers.

[1] http://www.mturk-tracker.com/
[2] https://github.com/ipeirotis/mturk-tracker-gae

The common structure of the community platforms consists of Threads, Messages, Posters and content. Threads are a section where users can post messages. A forum can contain many threads and organize them in some structured way, such as by discussion topic, organized by day or even task. Figure 2.3 shows an example of how threads are shown on one of the community platforms. Messages indicate who wrote it in a thread and have some form of content. Content can range from simple text, images, or links. For example, see Figure 2.4 which shows a message where the poster and content are highlighted. A common occurrence in some of the discussion platforms is a link to the average Turkopticon rating of a requester, as highlighted in the content of the message in Figure 2.4.

Figure 2.5 shows a timeline of these different community platforms indicating when they were active to illustrate the differences in fora lifetimes.

Figure 2.3: An example of threads on a forum

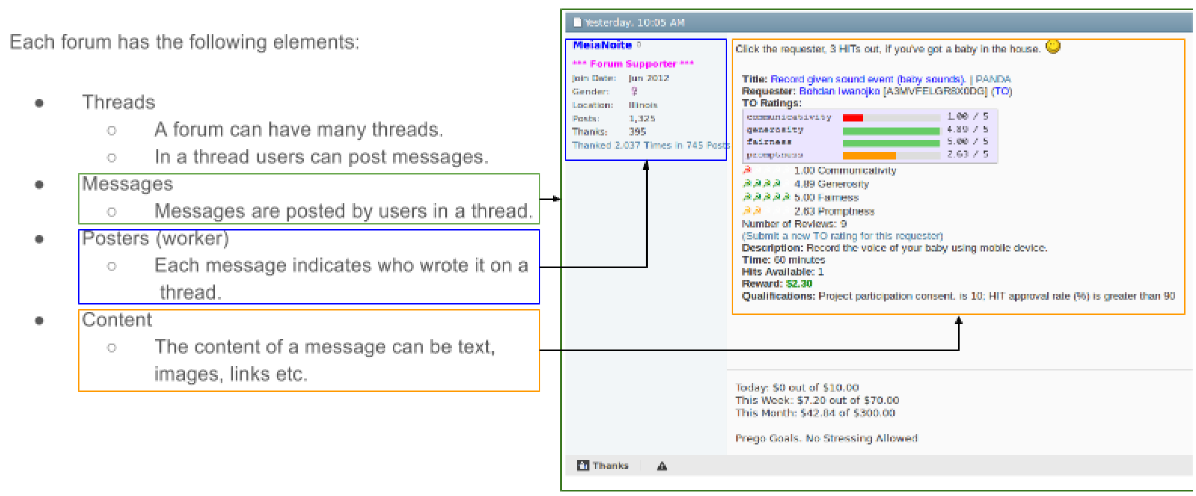


Figure 2.4: An example of a message and its contents on a forum

2.3. COMMUNITY PLATFORMS

In this section we give a short description for each of the chosen crowdsourcing community platforms. Some of these platforms have similar structure, rules and activities. However, some have differing mission statements about the type and format of messages and threads that may be posted on their forum.

Each message on these fora is identified with the poster username (as opposed to anonymous messages) and a number. Some fora have etiquette rules when posting about HIT Groups and requesters, some of these are written rules published on the website and are enforced by moderators. For example when workers reference a task, they are obliged to format their message in a certain way by providing a link to the task and a review on Turkopticon (much like in Figure 2.4). Table 2.1 shows an overview of the message characteristics

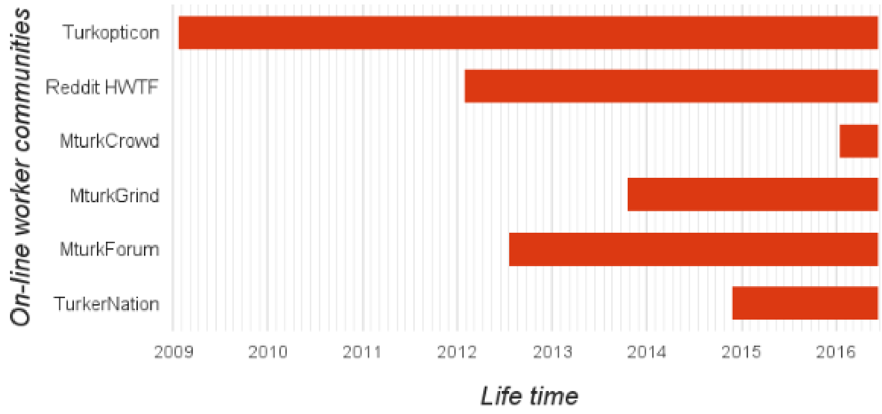


Figure 2.5: A graph indicating during which times the fora were active

	MTurkcrowd	**MTurkforum**	**MTurkgrind**	**TurkerNation**	**Reddit**	**Turkopticon**
Username	Link to user profile	Link to user profile	Link to user profile	Link blocked, login required	Link to user profile	Link to user post history
URL to HIT or HIT review	TO information of HIT with links	TO information of HIT with links	TO information of HIT with links	TO information of HIT with links	Link to HIT preview	Link to requester and HIT group on mTurk
Quote, link or reference to other message	Quote	Quote	Quote	Quote	Nested replies	Reply comment or flag review
Hit title	Included in TO Information	Included in TO Information	Included in TO Information	Included in TO Information	Provided in thread title	Provided by user in comment
Requester name or Group ID	Included in TO Information	Included in TO Information	Included in TO Information	Included in TO Information	Requester name provided in thread title	In review description
Messages	Counted as replies of thread starter	Counted as replies of thread starter	Counted as replies of thread starter	Counted as replies of thread starter	All messages counted	Not counted
Layout	Flat	Flat	Flat	Flat	Nested	Nested under review
Timestamp	Date	Date and Time	Date	Date and Time	Time since submit	Date
Order	Date and Number, Older first	Date and Number, Older first	Date and Number, Older first	Date and Number, Older first	Vote popularity	Date, newer first

Table 2.1: Message characteristics per forum (TO = Turkopticon)

per forum. This table shows for each forum the function of username links, how tasks should be referenced on the forum, how messages replies are shown, how the message number (that identifies the message) is counted, message layout on the forum, how the timestamp of the message is formatted, and in which order messages are shown.

2.3.1. TURKOPTICON

As seen in Figure 2.5, Turkopticon [8] is the oldest crowdsourcing community of mTurk having started in 2009. On their website it states “Turkopticon helps the people in the ’crowd’ of crowdsourcing watch out for each other’s because nobody else seems to be. Turkopticon lets you REPORT and AVOID shady employers.”, which means that the mission of this website is to help workers choose tasks from reliable requesters and report which ones are not.

Turkopticon functions by having workers review Requesters by providing a rating from 1 to 5 for each of the following attributes: Communicativity, Fairness, Generosity, and Promptness. Workers can also provide, in addition to the ratings, a short written report about the task(s), behaviour, and experiences relating to the requester. Workers can also reply to a review with a comment, which is then nested under the review. Turkopticon is also available as a browser plug-in that allows workers to view and contribute reviews while browsing the mTurk website.

2.3.2. REDDITHWTF

RedditHWTF is a subsection of Reddit, the subreddit HitsWorthTurkingFor, where workers post links to good paying tasks which they recommend to other workers.

This subreddit describes itself as "A Community for Sharing Good HITs - Users post links to good paying tasks, called HITs (Human Intelligence Tasks), that are available to be completed on Amazon's crowdsourcing service, Amazon Mechanical Turk". This subreddit was formed in 2012 and now reports to consist of roughly 35K Turkers. Threads are shown in typical reddit fashion, which is a list where the most popular voted threads are at the top and the latest 1000 threads are visible for browsing. Messages on this forum are organised in a tree structure, where replies to a message are nested under that message. Also present on this forum is a bot, called hit_bot, that periodically checks the posted tasks and posts messages about its availability.

2.3.3. GENERAL PURPOSE FORA

The other chosen fora are mTurkForum, mTurkGrind, TurkerNation, and mTurkCrowd.

Each of these fora have a similar statement; mTurkForum "Great HITs" section states "The original great hits forum. Share the best hits on amazon mechanical turk and start earning dollars!". mTurkGrind Awesome HITS section states "Everyday there is a HITS thread where workers come to post HITs and discuss the work we do. This is the go to place to make the most out of MTurk!" TurkerNation "Daily mTurk HIT Threads" section states "Please share any great HITs you see here!". mTurkCrowd [9] has a Daily Work Threads section without explicitly providing a statement, but the intention seems to be similar to the other fora.

These fora are fairly similar in structure, with threads and messages organised in the same manner. Threads are organised by day where users share and discuss tasks with each other. Threads are closed after the day is over, to enforce that messages are organised by day. Messages are shown in chronological order with the oldest message at the top. Workers can reply to each other with messages by citing or quoting the message of the other worker. Some fora allow workers to friend or follow each other. Table 2.2 shows an overview of what the profile page of a worker shows per forum. This table shows if the forum profile page displays the post history of the worker (number of messages and lists these messages), reputation systems (i.e. points, likes, etc.), the join date of the worker, date that the worker was last active, a friend or follower list, and the presence of a title of class system on the forum.

From this table we can observe that all fora show the post history of workers on their profile page, but not all fora have a reputation system, indication of join date and last activity. The general purpose fora tend to have a title or class system, for example forum moderators have the moderator class while other workers are labeled as "users".

	MTurkcrowd	MTurkforum	MTurkgrind	TurkerNation	Reddit	Turkopticon
Post History	Number and list	Number and list	Number and list	Number	List	List
Reputation	Likes and Trophy Points	Thanks	Post Ratings	Thanks	Link karma and comment karma	No
Join Date	Yes	Yes	Yes	Yes	Yes	No
Last Activity	Yes	Yes	Yes	No	Yes	Yes
Friends/Follower List	Following and Followers List	Friend list	Following and Followers List	No	No	No
User title or class	Yes	Yes	Yes	Yes	No	No

Table 2.2: Profile characteristics per forum.

2.4. PREVIOUS & RELATED WORK

In this section we introduce previous work in the areas of crowdsourcing market dynamics and crowdwork communities. There have been many studies about crowdsourcing from the aspects of market dynamics and crowdwork communities. Both of these aspects were studied in isolation, and the purpose of our study is to connect these two aspects of crowdsourcing and characterize the relations between them. To achieve this purpose we adopted some of the methods applied in previous work or used them as inspiration to form our own experimental methodology and analysis.

2.4.1. MARKET DYNAMICS

There are studies centred around the market dynamics of mTurk that introduce methods of gathering and analysing market data. These studies focus on the analysis of market attributes over time, which our study can draw from to perform our own analysis. By combining the analysis of the mTurk market with data collected from crowdwork communities we can answer **RQ2** and **RQ3** to gain a more comprehensive understanding of crowdwork.

Ipeirotis [7] performed an analysis on the mTurk market by gathering data of tasks and requesters. To gather data from the mTurk market Ipeirotis constructed the *mTurk Tracker*. The source code of the *mTurk Tracker* is publicly available on Github. This tracker produced a dataset of mTurk market data containing 2.56M distinct HIT groups and 130M HITs, collected from 2009 to 2014. From their data they determined that mTurk is a heavy tailed market, which means most of the activity was produced by a small fraction of the users. Also observed was most activity was centred around small tasks, meaning tasks with a small reward.

Difallah et al. [4] performed an analysis to determine which factors shape the dynamic of the mTurk market. This study was conducted using the *mTurk Tracker* dataset spanning 5 years. Several trend were observed by their analysis, such as the popularity of certain types of tasks, the growth of the amount of requesters etc. A predictive model was proposed for task throughput, indicating that batch (amount of HITs) throughput can be predicted by the batch size and task freshness. A strong weekly periodicity was observed in the market for both supply and demand.

For our study we wanted to compare market dynamics for some temporal relationship with community discussions. For this we chose to use Granger causality test. Granger [10] proposed testing causality between two time series. Given the time series X and another time series Y, Granger causality test determines if past values of X helps significantly predict Y. When performing such a test there is also a "lag" parameter when considering past values to determine optimal delay for prediction.

2.4.2. ONLINE BEHAVIOUR

Online community behaviour analysis is an important research, with the Web supporting social interactions that are growing in size and complexity . Our research uses these studies as inspiration to construct our own analysis of user behaviour within worker communities. By analysing online behaviour we can determine which roles users take and the nature of their discussions, we used such analyses to answer **RQ1**. Combining online behaviour studies with market data results in new perspective of worker behaviour effects, which allows us to examine the effect of crowdwork discussions on requesters and answer **RQ4**.

Butler et al. [11] study the formation, participation and maintenance of online communities and the roles of its members. It states that some members have a leadership roles and other member contribute more depending on different benefits. There is also a distinction between active participants and silent participants in these communities.

Cheng et al. [12] state that evaluations that users provide on content contributed by other users results in complex social feedback effects. The authors studied the effect of ratings on a piece of content affect its author's future behaviour. Their methodology was to first evaluate the quality of content that users contributed. Then, find two contributions of similar quality by two different authors, where one was positively and the other negatively evaluated by users. Finally, then comparing how the quality and rating of these users' content changed over time. This study found that negative feedback resulted more content but also in content of lower quality by that user.

2.4.3. CROWD STUDY

There has been recent studies about crowdsourcing communities that show and characterise workers collaboration. The studies performed on the workers was done to show that workers collaborate, our research expands on this notion to provide qualitative results of worker collaboration and its effects using several worker fora.

Gray et al. [1] examine online communities with the goal to show that workers collaborate for fulfil technical and social needs. Four large crowdsourcing platforms were studied, with datasets containing interview, survey data, and participant observations. This study showed that workers collaborate to manage the administrative overhead involved in doing crowdwork, such as choosing tasks and avoiding scams. Also examined were how workers communicated via fora, chat, phone and even in person to discuss crowdwork, which showed that workers actually turn to each other to perform work much like in a conventional work environment.

Yin et al. [3] show that the "crowd" is not as set of independent workers that work in isolation but that workers are interconnected and collaborate using online community platforms. With interviews and reports of workers was determined that there are indeed many workers that work independently, however there is a rich topology of networked workers present. Which means there is a network of interconnected workers present within the crowd.

Martin et al. [2] conducted an ethnomethodological study on the crowdworker community active on TurkerNation. Researchers observed worker discussions on the forum for a period of seven months, analyzing public threads towards better understanding (mTurk) workers' motivation and their attitude towards the different actors in the marketplace (especially requesters). Their work clearly pointed out that workers regard their activity on mTurk as paid work (and, often, main source of income), and as such, they strive for efficiency in work execution and fairness, and transparency in the way the work is evaluated and rewarded. Most discussions in the forum are centered on requester evaluation (praises as well as criticism), mutual support and opportunities for collective actions.

Laplante and Silberman [9] provide some insight to worker collaboration and trust using the worker community platform, mTurkCrowd, developed by the first author.

Irani and Silberman [8] present Turkopticon and examines the role and its effects on mTurk. Describing Turkopticon is a activist system that allows workers more visibility into the mTurk market.

Silberman et al. [13] reviews some of the problem of crowdwork from the perspective of a worker. Workers face problems such as no pay, spam, bugs etc. that are not addressed by crowdwork platforms.

3

Experimental Methodology

In this chapter we look at the data that can be obtained from the the mTurk market as well as the community platforms in details. Next, we elaborate on the research questions and examine how these research questions can be addressed using the described data.

3.1. mTurk Market Data

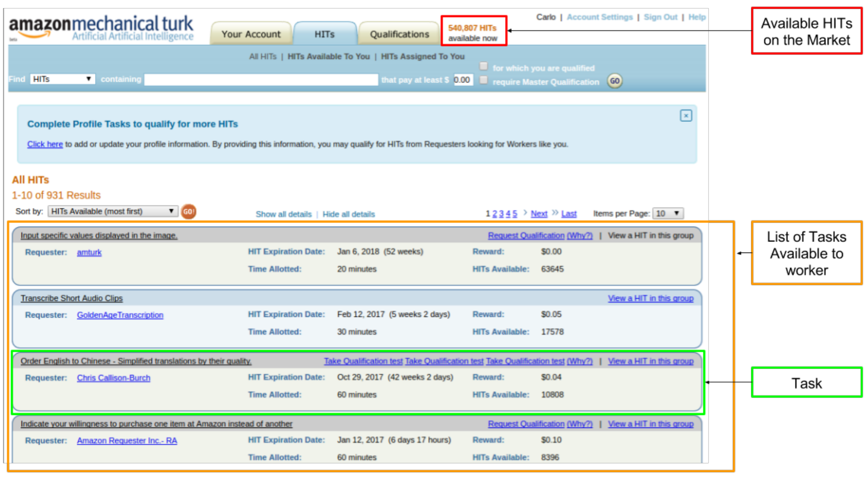


Figure 3.1: Worker view of Amazon Mechanical Turk

The Mturk Market provides the workers a list of tasks that are currently available as shown in Figure 3.1. Each task has a set of attributes show in the following Table 3.1. Most of these tasks attributes are visible to workers on the market just as shown in 2.2 and 3.1 (highlighted in green). After gathering the general attributes of a task, its progress has to be monitored by periodically checking to see if it is active and recording the changed attributes. This results is data as shown in Table 3.3, containing the GroupID, timestamp, and HITs Available. General market statistics, which are the number of HIT Groups and HITs available on the market, are reported on the mTurk website (highlighted red in Figure 3.1) and are recorded periodically in the format of Table 3.2.

This is the data that the *mTurk Market and Community Tracker* (detailed in Chapter 4) is designed to collect in relation to tasks on the mTurk market.

3.2. Community Data

Community data consist of scraped data from the community fora detailed in Chapter 2. The gathered data follows the general structure of the community platforms; threads and messages. As explained in Chapter 2,

Attribute	**Description**
Title	The Title of the task
RequesterName	Name of the requester that made the task
RequesterId	Unique identifier of requester
GroupID	Unique identifier of HIT Group
HitsAvailable	Amount of HITs that are currently available for this HIT Group
Reward	The amount of monetary compensation for successful completion of this task
ExpirationDate	The date which this task will be removed from the market
TimeAllotted	The amount of time workers are given to complete the task once they open the task
Description	A description what the task entails and how it is to be performed
Keywords	Words used to describe and organize tasks on the market
Qualifications	Qualities and accomplishments workers most have before they can accept this task

Table 3.1: HIT Group Attributes and Descriptions

Attribute	**Description**
HITs Available	The amount of HITs available on the entire mTurk market
HIT Groups Available	The amount of HIT Groups available on the entire mTurk market

Table 3.2: Market Statistics Attributes and Descriptions

Attribute	**Description**
GroupID	Unique identifier of HIT Group
Timestamp	The date which this entry was recorded
HITs Available	The amount of HITs available on the entire mTurk market
HITs Diff	The difference in the amount of HITs Available in relation to the last entry of this HIT Group
Rewards Available	Amount of reward that are currently available for this HIT Group, calculated by multiplying Reward with HITs Available
Rewards Diff	The difference in the amount of Rewards Available in relation to the last entry of this HIT Group

Table 3.3: HIT Group Tracking Attributes and Descriptions

each forum has some characteristics in common resulting in the scheme in Table 3.4 for threads and Table 3.5 for messages.

Figure 2.3 shows an example of how threads can be displayed on fora. It shows that the thread Title is visible, while the threadId and URL are derived from the link to the content of the thread. It is also worth noting that the author of a thread, as shown in the example figure, is the author of the first message in a thread. This is because a thread must contain at least one message.

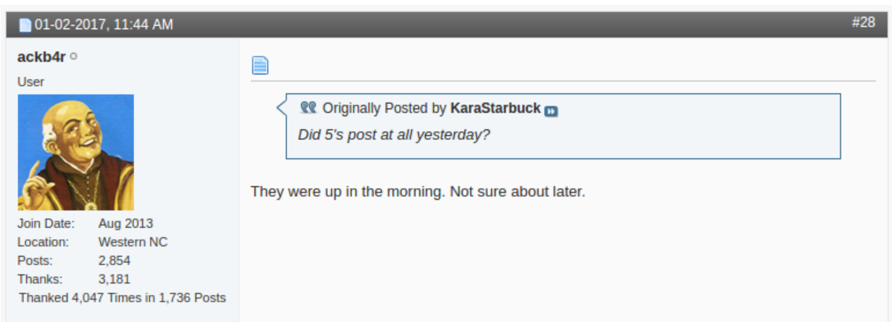


Figure 3.2: Example of a message that quotes another message

Figure 2.4 shows an example of a message on one of the fora. The main attributes described in table 3.5 are visible in this figure. The thread title, author name, content, and timeposted are visible while some attributes are contained in the links provided in the message. The *reply_to* attribute differs from the other attributes, because it is derived from the message content or context to indicate a reply. Figure 3.2 shows an example of a user replying to another user, where the message the user is replying to is quoted in the reply message. These replies contain quotes of or are nested under earlier messages to indicate that the message was a response to that earlier message. Because users can posts replies to each following the forum structure where the message is posted, we included the *reply_to* attribute to indicate the relationship of a message and reply. This *reply_to* attribute is the username for which the reply is meant, and indicates a relationship between forum users.

Attribute	**Description**
ThreadId	Unique identifier for a thread
Thread Title	Title of the thread
URL	URL adress of the thread

Table 3.4: Thread Attributes and Descriptions

Attribute	**Description**
ThreadId	Unique identifier of the thread which this message belong
Thread Title	Title of the thread where the message belongs
PostId	Unique identifier of this message post
author	Username of the worker that authored this message
authorUrl	URL adress of the author profile page
reply_to	The username(s) of the worker to which this message was replied
content	The content of the message. This could be text, images, or links.
timeposted	The server time when this message was posted
likes	The username(s) of the workers that liked (or something similar, i.e. thanks) this post

Table 3.5: Message Attributes and Descriptions

3.3. Market and Community Study

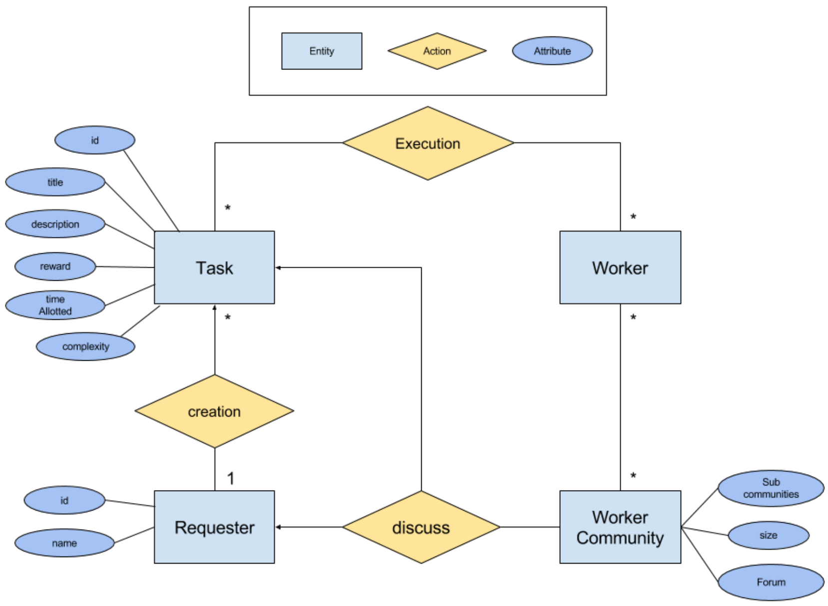


Figure 3.3: Conceptual model that shows the relationships between requester, task, worker, and worker community

By collecting data from both the crowdwork market and worker communities we can study many different aspects of crowdwork. Using previous work detailed in Chapter 2 about crowd study, we can verify their findings and expand on the understanding of the crowdwork relationships and dynamics.

By observing all the data in the context crowdwork and their communities in tables 3.1 and 3.5 we can construct a new conceptual model of crowdsourcing as shown in Figure 3.3, which expands on the model shown in Figure 2.1 in Chapter 2. This model shows that requester create tasks for workers to execute and adds that workers discuss requesters and tasks in worker communities.

When studying these relationships we can consider amazon mturk and the crowd community as a socio-technical system as shown in Figure 3.4.

The models illustrated in Figures 3.3 and 3.4 set the foundation for our research questions.

Figure 3.3 shows the relationships between requester, task, worker, and worker community. A requester can create many tasks, and each task has one requester. One or more workers can execute a task, and one or more tasks can be executed by a worker. One or more workers can form a worker community, and a worker can belong to one or more worker communities. Worker communities can discuss requester and tasks.

Figure 3.4 depicts the hypothesis that worker community and crowdwork platform form a socio-technical system, where mTurk triggers discussion within worker communities and worker communities in turn influence mTurk.

The following subsections explain the setup for each aspects relating to the research questions provided in Chapter 1.

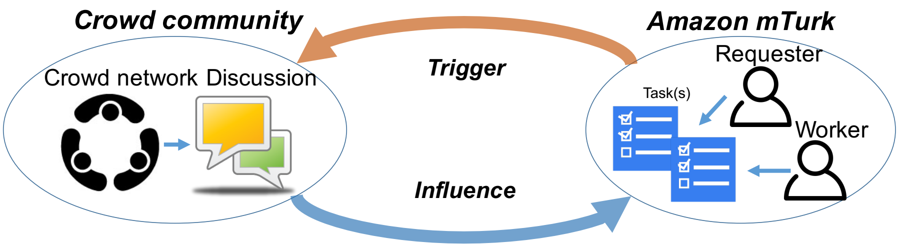


Figure 3.4: Bird's eye view of the socio-technical system that is built around Amazon Mechanical Turk

3.3.1. Community Discussions

Comparing Figure 3.4 with the old model of crowdwork, we can see that the community is a new and unknown component. To better understand the composition and role of crowdwork communities the composition and roles within these communities must be studied.

Using previous research from section 2.4.2 relating to online behaviour (Butler et al. [11]) as inspiration, we can analyse the formation, participation, and maintenance of these worker communities. Specifically we can determine who are the most important users according to their message posting frequency and lifetime on fora. Followed by examining the relationships between users to determine which are more central and valuable to discussions.

3.3.2. Linking Market and Community Data

In the online community discussions workers often reference tasks on the market using a link to a rating on Turkopticon or a link to the task on the market (for an example see Figure 2.4). These links contain the unique identifier known as the GroupId of a HIT Group, and sometimes also contain the unique identifier of a requester known as the RequesterId. By extracting these unique identifiers from messages we can link the community data and market data. Linking this data can be organized in different ways; linking messages to HIT Groups, or linking messages to Requesters.

For example. A link in a message that points to a specific HIT Group on mTurk looks like this:

`https://www.mturk.com/mturk/preview?groupId=37ZHE2JT1D250TKIB56TICN73UQ88W`

In this URL we see that the GroupID is defined at the end as "37ZHE2JT1D250TKIB56TICN73UQ88W". So now we have a message that has referenced a HIT Group, but not the HIT Group details. These details have to be gathered from mTurk at some earlier point in time by a tracker platform and stored like in table 3.1. If it is indeed the case that the data of the HIT Group from the mTurk market was collected, we have as a result a detailed instance where a message and a HIT Group are linked.

This process can also be done with requesters with a link such as:

https://www.mturk.com/mturk/searchbar?selectedSearchType=hitgroups&requesterId=A123A4M8WUYB09"

This URL links to a page on mturk that shows the HIT Groups the requester has available. Also commonly posted in messages by workers are links to Turkopticon such as:

https://turkopticon.ucsd.edu/A2ZA0WM6BTXZ4Q

Where the requesterID is also visible, and links to a page with Turkopticon reviews of the requester.

Once links between messages and tasks are established, the analysis of the data can provide some new insights. We can check to see which kind of tasks are mentioned more frequently, which requesters are related to the most discussion etc.

These links between the messages and mturk provide the basis for analysing the relationship and effects between crowdsourcing platform and worker communities which lay at the heart of the research questions **RQ2**, **RQ3**, and **RQ4**.

3.3.3. Market Dynamics Effect on Discussion

With the Statistics of Table 3.2 we can construct a time series of the market dynamics. And, along with a time series of the number of mentions of tasks, we can analyze these two time series to see if the market dynamics affect discussions on fora. Figure 3.5 shows a plot of such two time series. Using Granger causality explained in section 2.4.1, these two time series can be analysed for causality and provide some insight for **RQ2**.

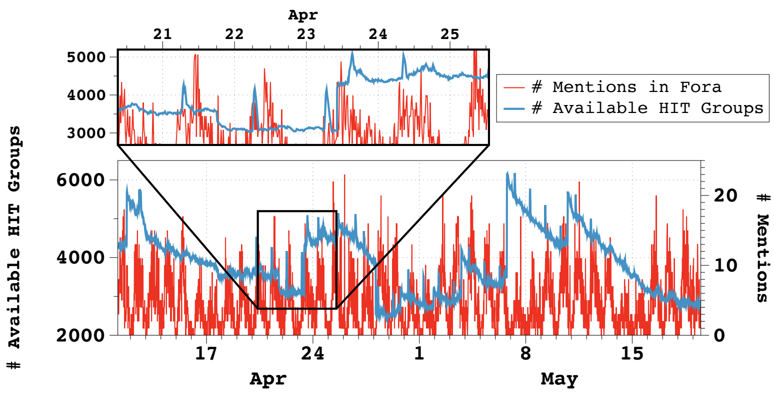


Figure 3.5: The dynamics of the crowdsourcing marketplace (mTurk) and discussion in the fora. The blue curve shows the # Available HIT groups in the market, and red curve shows the #Mention over time by crowd communities. A small period of the two time series (Apr. 21 – 25, 2016) is magnified to show the relation between them.

3.3.4. Discussion Impact

As stated before, the worker discussions can mention two things: tasks and/or requesters. It follows that we can study the impact on both of these aspects of the mTurk market.

With the data of HIT Groups of table 3.1 and 3.3, specifically the HITs Available attribute, we can study change in the number executions of a task over time once this task is discussed in the forum to answer **RQ3**. This would generate a similar plot as Figure 3.5, but with HITs available and number of mentions of a single task. Followed by a Granger causality test to determine the presence of some temporal relationship and then measurement of impact on HIT throughput. From this we can learn if there is some form of influence between tasks that are mentioned and the consumption of HITs.

For studying requester behaviour relating to **RQ4**, we must look at their tasks posted by the requester and the ratings they received on Turkopticon. Looking at the data of 3.1 filtered by RequesterID, we can examine how HIT Group attributes change over time alongside the ratings over time belonging to the same requester. Taking the study Cheng et al. [12] (described in section 2.4.2) as inspiration of user feedback online, we can determine how positive or negative evaluation affect requester behaviour. It is worth noting that ratings on Turkopticon do not specifically reference tasks (such as with a HITGroupId), and we worked under the assumption that ratings posted on Turkopticon after a task was published on mTurk are related to that task.

4

mTurk Market and Community Tracker

In this chapter we present the tool we created for collecting data from the mTurk market and crowdwork community fora, the *mTurk Market and Community Tracker*. This tracker platform is based on the work and tracker by Panos Ipeirotis [7].

Having our own tool that collects data from the mTurk market as well as worker community fora grants some advantages regarding to dataset creation, tracker functionality and customization. Having our tool would allows for the creation of a dataset that contains more recent data from the mTurk market that can supplements data that was collected during previous work. The additional functionalities allows for the enrichment of the dataset, for example the storage of HTML source code of the HIT Group web pages allows the possibility to analyse tasks in more detail. Using the source code available on GitHub [1], we were able to create a tool that can run on our own servers. Running the *mTurk Market and Community Tracker* on our own servers allowed us to collecting data the mTurk market and worker community fora. After running the the *mTurk Market and Community Tracker* from Apr. 11 and May 20 2016 were able to create a dataset with 46K HIT Groups of the mTurk market and more than 3M messages from 28K workers in the collective community. The collected data is explained and analysed in detail in Chapter 5.

This chapter is structured as follows, first we look at the requirements of the *mTurk Market and Community Tracker*. Next, we explain and justify the design choices we made. Then, we discuss the challenges encountered while creating the *mTurk Market and Community Tracker*. Finally, we provide an overview of the functionalities and explain the workflow of the *mTurk Market and Community Tracker*.

4.1. Requirements

The *mTurk Market and Community Tracker* has to fulfill the requirements of data collection and storage as detailed in Sections 3.1 and 3.2. The following list are the requirements that the *mTurk Market and Community Tracker* aims to meet.

1. Collect data of recently posted HIT Groups on mTurk
   The recently posted HIT Groups on mTurk should be recorded according to the schema from table 3.1 in Section 3.1.

2. Update data of the recorded active HIT Groups on mTurk
   The HIT Groups that are still marked as active should be looked up on mTurk by the tracker and record any changes to the HIT Group attribute HitsAvailable along with a timestamp according to the schema of table 3.3 in Section 3.1.

3. Collect data of worker community discussions
   The threads and messages from each of the community fora should be recorded according to the schema of tables 3.4 and 3.5 respectively from Section 3.2.

[1] https://github.com/ipeirotis/mturk-tracker-gae

4. Scheduling of data collection actions
   The tracker should allows for the configuration of a schedule for the collection and tracking of HIT Group data as well as the collection of data from the worker community fora.

5. Store all the source code of the HIT Group web page
   When the tracker encounters a new HIT Group, it should also store the source code of the web page of the task.

6. Take screenshot of HIT Group web page
   Using a headless browser (a browser without a graphical user interface), it should store a graphical representation of the task to the disk. This saves effort of reconstructing the web page from the source code.

4.2. Design Choices

In this section we look at some of the design choices we made when implementing the *mTurk Market and Community Tracker*.

- Decoupling from the Google App Engine
  The source code provided by Panos Ipeirotis on Github stated at the beginning of this chapter links to a version of the mTurk Tracker that runs on the Google App Engine. The decision of decoupling the application of the Google App Engine was made so that we had more control over the application and the dataset it generates.

- Servlet structure
  Having modified the application from the existing source code it was needed to keep the underlying servlet structure of the *mTurk Tracker*. Keeping the servlet structure was important for several reasons. Firstly, straying from a servlet structure would require a large amount of code to be rewritten. Secondly, the servlet structure allows us to add more functionalities by simply adding another servlet.

- Schedule frequency
  After some experimentation with scheduling, we settled on collecting and tracking HIT Group data from the mTurk market every 15 minutes and collecting data from the worker community fora once a day. For HIT Group data collection and tracking, we noticed that 15 min intervals was sufficient for several reasons; the amount of new HIT Groups found in smaller intervals was quite small, smaller intervals in HIT Group data tracking showed no changes for the majority of the active HIT Groups, and smaller intervals would provide enough time for the tracker to finish the data collection and tracking of the previously scheduled round. In short, smaller intervals yielded no new data and would not perform adequately according to such a packed schedule. For the worker community fora data collection, there is no need to collect the data in small intervals because most of the fora retain messages and threads for some time. Also, collecting data too frequently from worker communities could get the tracker banned from the fora.

- Removal of web front-end
  The *mTurk Tracker* provided by Panos Ipeirotis included a web front-end that shows statistics of the collected data in real time. We decided to not include that front-end into the *mTurk Market and Community Tracker* for several reasons. Firstly, our approach consists of collecting data to supplement another dataset and then perform our analysis separate from the tracker. This means that, from our approach, there was no need for an interface that shows graphical representations of the data in real time. Secondly, because we added a whole new aspect to the tracker, worker community data, we decided to mainly focus on data collection rather than expanding the web front-end.

4.3. Challenges

Creating the *mTurk Market and Community Tracker* provided some unique challenges stemming from the fact that it used existing code of Panos Ipeirotis' mTurk Tracker and different server system.

- Decoupling from the Google App Engine
  The Google App Engine is a platform as a service cloud computing environment, which means that

the programming and hosting of the application are coupled. Decoupling the *mTurk Tracker* provided a challenge, but was possible due the common underlying servlet architecture of applications on the Google App Engine. This allowed us to convert the application to run on Apache Tomcat.

- Including worker community fora
  As stated above, the mTurk Tracker consists of servlets. Adding the functionality of collecting data from the worker community fora had to be implemented by creating servlets for each forum Although the fora did have many things in common, and some created with the same forum software, there were enough differences that there had to be a specialized implementation (servlet) for each of the fora. Code was reused for the parts the fora had in common, for example mTurkForum and TurkerNation are both powered by vBulletin (a popular forum software) so some part of the HTML they generated was the same.

- Adding extra functionalities
  As stated before, adding extra functionalities is done by creating additional servlets. However, the nature of the extra functionalities, task screenshot and task web page content storage, provided interesting challenges. For the task screenshot functionality, it was required to use a headless browser, a browser without a graphical user interface. A headless browser allowed us to automate the loading of the task web page, render the page in memory, and then store it to disk. However, implementing such a headless browser (we chose PhantomJS) into a servlet required a workaround. For the task of web page source code storage we were able to reuse code from Friso Abcouwer's version of the *mTurk Tracker* (the *Extended MTurk Tracker*). His version had the content retriever running on an external server, and we decided to incorporate it into the *mTurk Market and Community Tracker*.

- Gaining access to data sources
  Several web pages the *mTurk Market and Community Tracker* visits, such as some of the worker community fora, required a login (and sometimes permission from the admins) before any data could be scraped. It was a challenge to painstakingly create a login script for each of these pages.

- Location differences
  mTurk is meant for workers from the United States and India, and visiting the market page from the Netherlands resulted in some complications when trying to collect data. Not all the data is shown on the pages when accessed from the Netherlands because we here are not considered eligible workers. We were able to create a workaround that required logging into mTurk to collect data.

4.4. SYSTEM OVERVIEW

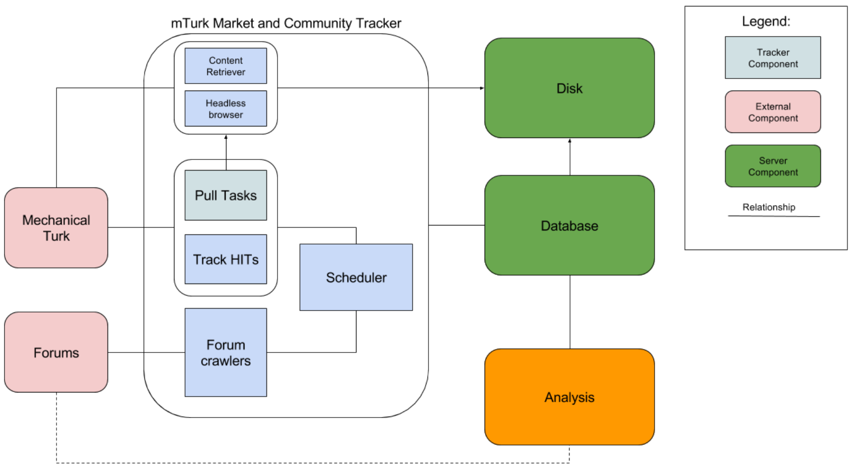


Figure 4.1: Tracker Overview

Using the source code provided by Panos Ipeirotis [7], we were able to create a version of *mTurk Tracker* which allowed us to run our own tracker to collect data from mTurk. Figure 4.1 shows an overview of the design of the *mTurk Market and Community Tracker*. The tracker consists of three main internal parts; the market data collection & tracking, forum crawler, and a scheduler. The core functionalities correspond with the requirements detailed earlier in this chapter:

1. Pull Tasks:
   This servlet collects data of recently posted HIT Groups on mTurk and stores it in the database. When this servlet encounters a new HIT Group, it sends a request to the *Content Retriever* and *Headless Browser* servlet to collect supplemental data.

2. Track HITs:
   This servlet first iterates thought the HIT Groups that are marked as active in the database. For each of these HIT Groups it tries to visit the corresponding HIT Group web page on mTurk and makes an entry in the database if there are any changes to the HIT Group. If it cannot find a web page, then the HIT Group is marked as inactive in the database.

3. Forum Crawlers:
   This represent the collection of servlets that collect data of worker community discussions. There is a servlet for each community forum. In general these servlets function by loading the forum pages and scanning the thread. If there are any changes to the threads or if there are any new threads, it opens those thread pages and updates the database. When a thread page is opened, it scans though the messages of the thread and stores any new messages in the database. The schema in which the thread and message data are stored is detailed in Section 3.2.

4. Scheduler:
   This servlet facilitates activation of the other servlets according to the configured schedule. The schedule configuration consist of a list of cron (time-based job scheduler) jobs for each servlet. The scheduler also keeps track of which servlet are active so that it does not issue requests to already active servlets.

5. Content Retriever:
   The content retriever downloads all the source code of the requested web page and stores all the source code of the HIT Group web page to the disk. This servlet was originally created by Friso Abcouwer.

6. Headless Browser:
   This servlet creates a call to a PhantomJS (our chosen headless browser) script, which loads the requested page, renders that page in memory and stores the information to the disk.

5

Analysis

In this chapter we look at the analysis performed on market and community data.

First, we begin by providing an high-level overview of the data gathered using the *mTurk Market and Community Tracker* as described in Chapter 4. This tracker was made with the purpose of gathering data from both the mTurk market as well as worker community fora. Second, we analyse worker community discussions to determine what workers discuss and which workers are more central and valued in discussions. Third, we determine the influence of the crowdwork market on worker discussions. Fourth, we measure the impact of worker discussions on tasks on the crowdwork market. Inspired by the results of these analyses, we discuss some guidelines for task design and crowdwork platform design. Next, we provide a summarize discussion on our findings. Finally, we discuss some threats to the validity of the work performed in this chapter.

5.1. Overview On the Collected Data

The market data consist of a dataset obtained during previous studies, which is also supplemented with data gathered by our own tracker. The end result is a dataset of market data that spans 6 years. The initial dataset from [4] contains 2.56M HIT groups and 130M HITs from 2009 until 2014. This is then supplemented with data from our own tracker which contained 46K HIT groups and 1.9M HITs, from the period between Apr 11 until May 20 2016.

Worker community data was obtained by our modified tracker (See Chapter 3 and 4 for details) which collected data that was available on the worker community fora. Table 5.1 shows some statistics of the worker community fora we were able to collect. This table shows the data we collected from the fora mTurkForum, mTurkGrind, TurkerNation, mTurkCrowd, Turkopticon, and Reddit HWTF. For more details about these fora see Chapter 2. The table shows for each of these fora the following attributes (and for some attributes the total of the worker community) ; the start date, meaning the earliest message retrieved from the forum; The number of members recorded per forum, which is the number of distinct message authors within each forum; the number of threads recorded on the forum; the number of messages; the number of replies, which are messages that quote another message (for an example see Figure 2.4); the average of messages per member, indicating the amount of activity per member; the average amount of messages per thread; the average reply per member, indicating a measure of discussion activity among members; the average message per reply; and the purpose of each forum.

From Table 5.1 we can make some interesting observations. These observations will be useful to explain some results from analysis in the following sections.

- We can see that Turkopticon is the oldest forum with the most members, but has the a unique purpose of facilitating evaluations of requesters. As stated before in Chapter 2, Turkopticon is unique in its purpose and platform structure and can be observed by the statistics we observe in this table.

- It is worth noting that the Reddit HWTF start date is march 2016 while the subreddit exists since 2012 (explained in Chapter 2), this is because of the limitations of the data crawling of the reddit pages. There is however a dataset by other parties that contains the older data, but that can be left for future work.

- The reddit community also sets itself apart with a unique purpose: the advertisement of HITs. From the reply count and message per thread we can observe that threads merely contain the suggestion of a task with rarely a response. Indicating that it is a rare occasion that actual discussions occur.

- In general we can see that the older fora have more members, threads, and messages. However on the youngest forum, mTurkCrowd, members tend to reply to each other quite often according to the number of replies and the replies per member. Showing that their members tend to engage each other discussions relatively often.

Forum	**Start**	**#Mem**	**#Thr**	**#Msg**	**#Rep**	**Msg/Mem**	**Msg/Thr**	**Rep/Mem**	**Msg/Rep**	*Purpose*
mTurkForum	07/12	4,926	1,889	1,427,856	705,249	289.9	755.9	143.2	2.02	General
mTurkGrind	10/13	3,217	943	948,775	441,660	294.9	1006.1	137.3	2.15	General
TurkerNation	11/14	572	563	173,942	73,407	304.1	309	128.3	2.37	General
mTurkCrowd	01/16	616	131	177,669	105,588	288.4	1356.3	171.4	1.68	General
Turkopticon	01/09	18,640	310,129	371,750	56,919	19.2	5.1	8.4	2.27	Evaluation of requesters
Reddit HWTF	03/16	930	3,490	17,843	7,849	19.9	1.2	3.1	6.53	HITs advertisement
Total		28,901	317,145	3,117,835	1,390,672	202.7	572.3	98.6	2.84	

Table 5.1: Descriptive statistics of the six targeted fora. Legends: **Start** – earliest crawled message; **#Mem** – number of community members; **#Thr** – number of forum threads; **#Msg** – number of messages in the forum; **#Rep** – number of replies; **Msg/Mem** – average messages per member; **Msg/Thr** – average messages per thread; **Rep/Mem** – average replies per members; **Msg/Rep**: average replies per message.

5.2. Discussion on Worker Fora

Determining what is being discussed and by whom allows us to better understand worker communities. In this section we aim to answer **RQ1**. To do so we provide and overview of what is being discussed by by whom.

To determine what worker discuss we decided to classify the content of the messages on fora according to what we observed in the fora. We observed different types of messages such as messages where users ask each other about crowdwork, or share their experiences etc. By first annotating a sample of messages ourselves we were able use an automated process to classify messages of the worker communities within an acceptable range of certainty.

We looked at worker histories and relationships to better understand what types of workers are part of the online worker communities. First, we studied the amount of messages workers post as well as their lifetime on fora. Then, we examined community structure by analysing worker interaction on the fora to determine which workers are more central and valuable in discussions.

5.2.1. Classification of Message Content

Looking at the content of messages we can get a better understanding of what workers discuss on crowdwork community fora. To achieve this goal, we set up a content classification pipeline, where messages are categorised in one or more of the following 6 types:

- Ask or Answer: messages that include questions asked by a worker, or answers to previous questions. This category included messages inquiring for general purpose issues with mTurk or fora (e.g. how to obtain qualifications), or seeking for explanations about tasks.
- Comment: messages that include general comments about the task, such as its availability, requirements, or presence of bugs (e.g. lack of completion code).
- Experience: messages that report the experience of a worker in the execution of a task, e.g. the amount of time spent on a task, or the amount of rewarded bonus.
- Judgement: messages where workers explicitly express compliment or criticisms about tasks or requesters.
- Rating: messages that include a reference to Turkopticon rating, or rating in other fora. Rating messages often serve as recommendation from workers to the community, as only tasks worthy of discussion are mentioned.
- Social: messages where workers address the community with general-purpose topics, such as greetings and jokes.

	Type	Content	Forum
Message	Ask or Answer	Anyone able to withdraw?	`mTurkCrowd`
	Comment	Can't be on mobile.	`Reddit HWTF`
	Experience	Projected Earnings for Today $70.00	`mTurkGrind`
	Judgement	$0.60 cent one is good, 0.36 hit sucks	`Reddit HWTF`
	Rating	This requester has actually joined Opticon just to flag negative reviews and accuse them of blackmail.	`Turkopticon`
	Social	Turtles for days Happy new year!	`mTurkGrind`

Table 5.2: Examples of threads and messages of each type.

Messages can belong to multiple categories, for example a worker can start with social banter and then go on to provide a rating and judgement about a task in the same message. Table 5.2 shows the types of message classifications that were made along with some examples.

These message type are a result of a combination of previous work and the card sorting technique. Card sorting is a technique widely used in the design of information architecture to create mental models and derive taxonomies from input data [14]. There are several card sorting techniques, and we used open and closed card sorting. Open card sorting has no predefined categories where the participants have to sort the cards info groups and describe each group, which allows for the creation of new categories and information organizations. While closed card sorting has predefined categories and asks participants to sort cards into these groups, which is useful in classifying information in existing categories and structures. We obtained an unbiased sample by randomly selecting 10% threads for each worker community, and of these threads we picked a random sample of at most 50 messages. In the case of Turkopticon, which has a different structure compared to the other fora, we randomly selected a sample 500 threads for classification. We elicited messages types that occur across all fora following the example of Martin et al. [2]. Then, using the open card sorting technique we were able to synthesize and define the six message types described above. Open card sorting was employed in order to create message categories that fit the collective worker community. The next step was to classify messages according to the established message types using closed card sorting. We did this by having three participant manually assign cards to message types. The participants reviewed each others work to reduce bias and strengthen the result.

5.2.2. AUTOMATIC MESSAGE CONTENT CLASSIFICATION

Classifying all the messages in the dataset allows us to get an overview of the types of discussions workers have. Having collected more than $5M$ messages, it would take a large amount of time to classify each message by reading and annotating each one with one or more category. To solve this, we turned to machine learning for the verification of the annotation results and automated classification of messages. The process of machine learning allows computers to learn and look for patterns in data. Messages of the categories stated above usually follow some pattern according to their textual features. As stated before, messages can belong to multiple categories, which means the machine learning classifier must be able to categorize messages with more than one message type. For this we fed a multi-label random forest classifier with textual features of the annotated messages and trained to predict message's type. Random forests are a learning method for classification that creates a collection of decision trees from the training set for class prediction [15].

The textual features used to train the classifier were bag-of-words and TF-IDF weighted. Bag-of-words is a representation of text where a sentence is seen a collection of words, irrespective of grammar and word order [16].

TF-IDF (term frequency–inverse document frequency) weighting is a statistical measure used to evaluate how important a word is to a document in a collection or corpus [17].

The combination of random forest learning technique and these textual features allows us to train a classifier that can determine a message's types according to the words used in messages.

We assess the performance of the classifier both in terms of accuracy and F-score (harmonic mean of precision and recall) in a 5-fold cross-validation setting. This allows us to measure how the classifier can perform by using the training set for testing.

Table 5.3 reports the classification results. This table shows the classification accuracy and F-score for each message type. It shows that Rating, Social, and Comment messages are those identified more accurately by the classifier, most likely because these messages follow some pattern. For example, messages of the type Rating are more easily identified because workers follow the same structure within messages by linking to Turkopticon ratings. The classification of Experience and Judgement messages is also accurate and with

Type	Accuracy	F-Score	Type	Accuracy	F-Score
Ask or Answer	0.86	0.27	Comment	0.98	0.60
Experience	0.80	0.46	Judgement	0.86	0.46
Rating	0.93	0.85	Social	0.75	0.74

Table 5.3: Classification results for message types.

acceptable F-score (> 0.4).

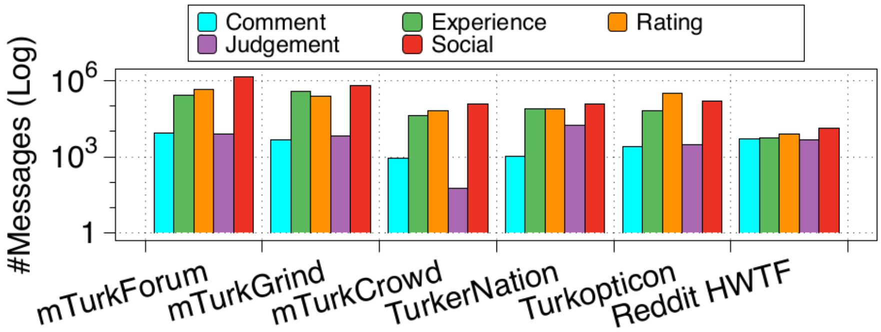


Figure 5.1: Distribution of message types (in #Messages) in fora after the classification of all the messages.

Figure 5.1 shows the distribution of message type in the observed discussions on each forum. Generally across fora we can see from this figure that the most common messages were Social, Experience, and Rating. From this we can observe that users often discuss a social topic, and when discussing tasks, they provide a rating and their experience. Judgment type message were not as popular, especially on mTurkCrowd. This might be due to the popularity of using Rating messages, which suggests workers prefer standard ways to express their opinion about requesters and tasks. From this figure we can see that mTurkForum, mTurkGrind, mTurkCrowd, and TurkerNation follow the same message distribution, because of their structural similarities. They have a of high number of Social, Rating, and Experience messages, while having a relatively low number of Comment and Judgement messages. Turkopticon also follows the general trend among the fora, but has Rating the most popular type of message. Reddit has an almost even distribution of message types, which can be due to its forum purpose (providing recommendations). Which means that discussions rarely occur on the RedditHWTF forum.

5.2.3. WORKER TYPES

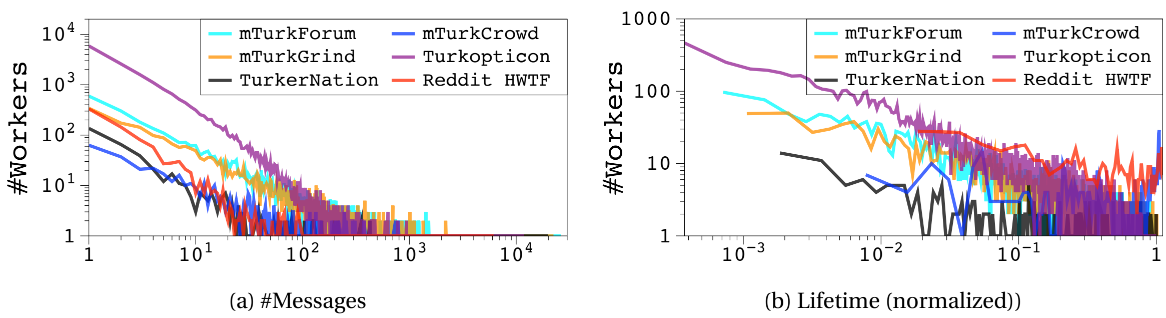
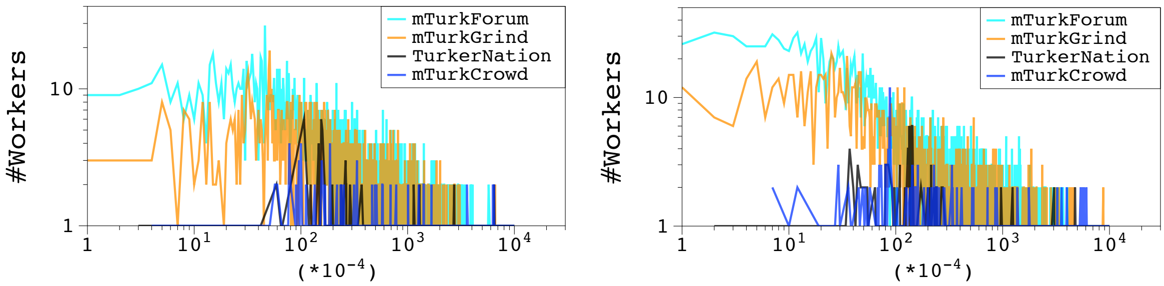


Figure 5.2: Activity distribution and normalized lifetime of community members.

According to Butler et al [11] there are different types of members within an online community. Understanding if worker equally and consistently contribute to discussions in online crowdwork communities provides some insight into the kinds of workers that exist within the community.

Figure 5.2 (a) shows a plot of the number of workers in relation to the number of messages, and Figure 5.2 (b) shows the number of workers in relation to the worker lifetime in the forum. This lifetime is calculated by taking the difference in time of the first and last message post of a worker normalized by the lifetime of the forum. From these figure we observe that there are many workers with a low amount of messages, a low amount of workers with many messages. This means that there are few workers that post most of the messages on the forum, while many rarely post any messages. Similarly, there are few workers with a long lifetime, and many workers with a short lifetime. This indicates that there are few workers that stay and participate within the community while many do not. From these figures we can also see these different communities have similar properties and that community size does not affect their distribution. For example, we can see in Figure 5.2 (a) that for mTurkForum and Turkopticon that the number of workers in relation to the number of messages seem to change in the same trend even though their community sizes differ. From these figures we can observe that there are a relative small amount of "power workers" that dominate discussions. These workers post many messages and have a long lifetime in the community. We refer to these members as doers, while the other users that post infrequently or opportunistically are referred to as observers.



(a) Authority (b) Hub

Figure 5.3: Authority score and hub score distribution crowd workers in different fora.

We also looked at which users fostered discussion in the fora. Understanding which workers initiate and foster discussion also provides some insight to different types of workers in the community. By studying the reply_to property, as described in Chapter 3, we were able to analyse the interaction between community members. Worker communities can be represented as a network of nodes and edges, where community members are nodes and reply messages are edges.

To analyse the worker community we used the HITS method [18] proposed by Keinberg et al. HITS measures the importance of a node in a network with an authority value and a hub value, which represent the value of a node's link to other nodes and centrality of the node in the network respectively. In short, we analysed which workers were more valuable and central in discussions. Figure 5.3 (a) shows the relationship with the number of workers and authority score, and Figure 5.3 (b) shows the relationship between the number of workers and the hub score. We chose the general purpose fora for displaying these scores in these figures because these are the communities where discussions tend to take place. We can observe in these figures that some fora have a small portion of workers that have high authority and hub score. This means that there is small portion of workers that are more valuable and central in discussions within their community. We hypothesized that there is some effect where older users become more central to community activities. To test this we classified member according to the 80-20 rule-of-thumb [19], also known as the Pareto principle. This means that roughly 80% of effect come from 20% of the causes, and for us this means that 20% of the workers in the communities owns roughly 80% of the hub and authority score. For each forum we isolated the top 20% of workers as leaders and followers according to their authority and hub score. From this we also observed that the vast majority of leaders and followers are also doers in fora (on average, 91.83% and 94.33%, respectively). There is also a significant overlap (78.45% on average) between Leaders and Followers. The overlap is higher in general purpose fora like mTurkCrowd (90.26%), mTurkGrind (87.88%), mTurkForum (87.60%), and TurkerNation (85.71%); this hints to the presence of a core set of active members.

To summarize we can combine the observations from Figures 5.2 and 5.3. There is a small portion (core set) of community members that post many messages, have been around for a long time, are more valuable and central to discussions. Also, there is a large portion of the community that consists of transient workers that rarely contribute to discussions.

5.3. MARKET INFLUENCE ON DISCUSSIONS

Studying the influence of the mTurk market on discussions in the fora allows us to better understand what triggers worker discussions. In this section we seek answer to **RQ2**. To do this we compared the data from the mTurk market with the data from forum discussions.

First, we linked the HIT Groups and requesters that workers discussed with the data gathered from the mTurk market. This data linking procedure is discussed in more detail in Section 3.3.2. We provide an overview of the linked data to show the statistical links between tasks and requesters found on the fora and mTurk.

Then, we investigate the extent by which changes in the mTurk market can trigger discussions among workers. We looked at the availability of tasks on the mTurk market over time and compared it to the number of tasks mentioned on the fora over time to determine if there is some relationship between the market and worker activity on the fora. The Granger causality (explained in Section 2.4.1) test was used to determine if there is a temporal relationship between the mTurk Market and worker community discussion. We investigate if discussion are more likely to occur for tasks having a certain property. For example we examine if high reward equates to more mentions of the task on the fora. We check if the perception of requester attributes by workers spark discussions in fora. For example examine if a requester with a high fairness rating results in more mentions of the requester or their tasks in the fora.

5.3.1. LINK MARKET AND DISCUSSION

To see which tasks on the market is discussed on a forum, we first linked the market data with the community data. In Chapter 2 we showed how messages mention tasks in discussion. Table 5.4 shows some statistics of the market and community data we were able to link to each other. This table consist of two sections, the linking of tasks (Linked HITs) and the linking of requesters (Linked Requesters) to messages.

The Linked HITs section of this table shows the following statistics: the number of messages that mention HITs, these were extracted from the messages as described in Section 3.3.2; The average messages mentioning HITs per user(AvgHMW) is given for each forum to provide some insight into the activity of the users across fora; The percentage of messages that mentioning HITs (%HMF), this indicates how frequent tasks are mentioned on the fora. The number of distinct HIT Groups that are mentioned per forum(#HITs), because HIT Groups can be mentioned multiple times during discussions between workers; The percentage of HIT Groups on the market that were mentioned in messages(%MH), here the market is represented by the tasks that were active in the datasets (comprised of a dataset from previous work and dataset created by our tracker, see Chapter 4) within the time-frame the forum was on-line, the mentioned tasks are then compared to these tasks to determine how much the forum discussion covered the market.

The Linked Requesters section of this table show some similar statistics (from the Linked HITs section of the table)for each forum from the perspective of requesters; The number messages that mention requesters (#RM), extracted from messages as detailed in Section 3.3.2. The average amount of requesters mentioned per user in the forum (AvgRMW). The percentage of messages in the forum that mention requesters(%RMF). The number of distinct requesters mentioned in the forum(#REQ), requesters can be mentioned either as the main topic of conversation or by their tasks. The percentage of requesters in the market mentioned on the fora(%MR).

From this table we can make some observations:

- Turkopticon info is not available in the Linked HITs section. This is because Turkopticon aims at reviewing requesters and tasks are not mentioned specifically.
- mTurkForum and mTurkGrind have the most mentioned tasks and requesters, but are also the oldest among the general purpose fora. From this follows that the younger fora, TurkerNation and mTurkCrowd, have less mentioned tasks.
- Although TurkerNation has a relatively low amount messages that mention tasks(#HM) and requesters(#REQ) compared to the older fora mTurkForum and mTurkGrind, its users more frequently post messages containing HITs or requesters as shown by its high AvgHMW, %HMF, AvgRMW, and %RMF. This indicates that this small and young community is quite active in discussing tasks and requesters.
- mTurkGrind discussions were able to cover almost a quarter of the tasks in the mTurk market. This number is higher that the older mTurkForum (which mentioned more tasks) because the %MH only looks at the time frame the forum was active. This means that in the time mTurkGrind was active,

their worker discussions covered more of the mTurk market. While in the timeframe mTurkForum was active, their worker discussions covered less of the mTurk market.

- We can see that the number of messages that mention tasks(#HM) and the number of messages that contain requesters(#REQ) are close to each other for most of the fora, which shows that workers tend to mention tasks along with the requesters.
- mTurkCrowd is a forum that is still quite young, but their worker discussions has achieved the most coverage of the requesters on the market (%MR). This might be because it is still young and will over time approach the same coverage as the other fora.
- Turkopticon ratings would be expected to cover most of the requesters on the market, because its goal is to review requesters. However, this is not the case. One explanation of this might be Turkopticon's age. Turkopticon started in 2009 when mTurk had a smaller amount of registered workers, and as a result an even smaller active worker community that reported requesters on Turkopticon. Over time both mTurk and Turkopticon have grown in both popularity and the amount of workers.

Forum	Linked HITs					Linked Requesters				
	#HM	AvgHMW	% HMF	#HITs	% MH	#RM	AvgRMW	% RMF	#REQ	% MR
`mTurkForum`	233,294	47.36	16.30%	104,893	15.81%	217,489	44.15	15.19%	22,737	43.08%
`mTurkGrind`	229,504	71.34	24.08%	100,310	23.07%	233,063	72.45	24.46%	22,078	60.44%
`TurkerNation`	73,637	128.74	41.87%	41,659	5.44%	73,767	128.96	41.95%	11,610	45.69%
`mTurkCrowd`	40,771	66.19	21.88%	19,278	11.76%	41,522	67.41	22.29%	6,896	85.66%
`Turkopticon`	NA	NA	NA	NA	NA	371,797	19.95	94.50%	45,701	54.10%
`Reddit HWTF`	1,937	2.08	10.01%	1,649	2.58%	635	0.68	3.28%	415	9.21%
Overall	579,143			184,390		938,273			50,912	

Table 5.4: Statistics of links found between tasks and requesters in forum and mTurk. Legend: #HM – number of messages with Links to HIT groups; AvgHMW – average number of HM per user in forum; % HMF – percentage of HM in Forum messages; #HITs – number of unique HIT groups mentioned in forum messages; %MH percentage HIT groups in the market mentioned in messages; #RM – number of messages with Links to Requesters; AvgRMW – average number of RM per user in forum; % RMF – percentage of RM in Forum messages; #REQ – number of unique Requesters mentioned in forum messages; %MR percentage of requesters in the market mentioned in messages;

5.3.2. THE TEMPORAL DYNAMICS OF MARKET DISCUSSIONS

To understand the relationship between the dynamics of the mTurk market and the discussion by worker communities, we compared the temporal distribution of the amount of HIT Groups available in the market, with the temporal distribution of the amount of HIT Group mentions in forum messages. Figure 3.5 shows the time series of both these distribution of the time period Apr 11 until May 20 as recorded by our *mTurk Market and Community Tracker* (Chapter **??**). The blue curve shows the available HIT Groups (which mTurk reported and the tracker recorded) and the red curve shows the number of mentions of tasks in the fora.

When studying these time series, we found that there is a weekly periodicity. Difallah, Djellel Eddine, et al. [4] also found such a weekly periodicity of the mTurk market. Also, the mentioning of tasks in the worker communities also display a weekly periodicity with more mentions during the weekdays.

Next, we investigated if the the market influences discussion on the fora. To test this we used Granger causality [10] explained in Chapter 2. The Granger causality test determines if one time series is useful in forecasting another. In our case we test to see if the available HIT Groups time series can predict the mentions on the fora time series.

According Granger causality the available HIT Groups signal (blue) contains information that helps predict the discussions signal (red) beyond just the past information of the discussions. When testing for causality, lag is also considered, where values of available HIT Groups of the past x time are considered for testing for causality. Optimal lag is usually selected by searching for the one with the lowest AIC/BIC (Akaike or Bayesian information criteria [20]) within a predefined range.

In our case we tested for a lag between 1 and 6 hours, and found the high significance was achieved with a lag of 4 hours. This means that that the effect of the market dynamics on crowd communities is visible (on average) after 4 hours. This 4 hour gap between market activity and crowd communities could indicate that workers tend to take their time to find and assess tasks before mentioning them to other workers. Another possibility is that workers might tend to first perform crowdwork when it is available then when their "shift" of 4 hours ends, then go to the fora to discuss these tasks with other workers or find tasks for their next shift.

5.3.3. HIT GROUP PROPERTIES

Comparing HIT Group properties of tasks that were mentioned and tasks that were not mentioned yields some interesting results. Table 5.5 shows the mean, median and standard deviation of the mentioned and non-mentioned tasks. From this table we can observe significant differences in all properties when comparing mentioned and non-mentioned task properties, except for TimeAllotted.

Mentioned tasks tend to have bigger batch size, this might be because bigger batch size also indicated availability of the task. For example, A worker would like to mention a task that others can also perform, so it follows that a large batch size allows more of their fellow community members to perform the task. Tasks with low reward are mentioned more often, which means a high reward does not always attract more workers. This is because work with high reward might indicate a difficult or time consuming task, and workers might want to perform many short and easy tasks instead. The tasks that workers mention tend to have some qualification or requirement, this might be because workers in the fora might already be well established qualified workers on mTurk and these requirements provide them more of their preferred task types.

We calculated the Spearman correlation for each property of HIT Groups and the number of mentions in fora. This shows us which task properties result in more mentions by workers. Figure 5.4 shows some these results in a bar chart. From this figure we can observe that overall workers frequently discuss tasks with approval requirement, large batch size, low reward, and little TimeAllotted. Looking at the the results across fora we can see that discussions generally follow the overall trend. This means that, across the fora, workers mention for the same types of tasks from the mTurk market.

	Batch Size	**Reward** (cents)	**TimeAllotted** (seconds)	**Requirement**
Mentioned HIT Groups	$313.29, 1 \pm 1755.97$	$67.07, 25 \pm 369.85$	$1.82e04, 3.6e03 \pm 2.56e05$	$0.62, 1 \pm 0.49$
Non-Mentioned HIT Groups	$25.36, 1 \pm 409.64$	$115.67, 45 \pm 448.41$	$2.01e04, 7.2e03 \pm 3.31e05$	$0.18, 0 \pm 0.38$

Table 5.5: Mean (μ), median (m) and standard deviation (σ) – $\mu, m \pm \sigma$ – value of HIT groups' properties. *Requirement* is a boolean variable , with a value of 1 encoding "Qualification or approval rate greater than x needed".

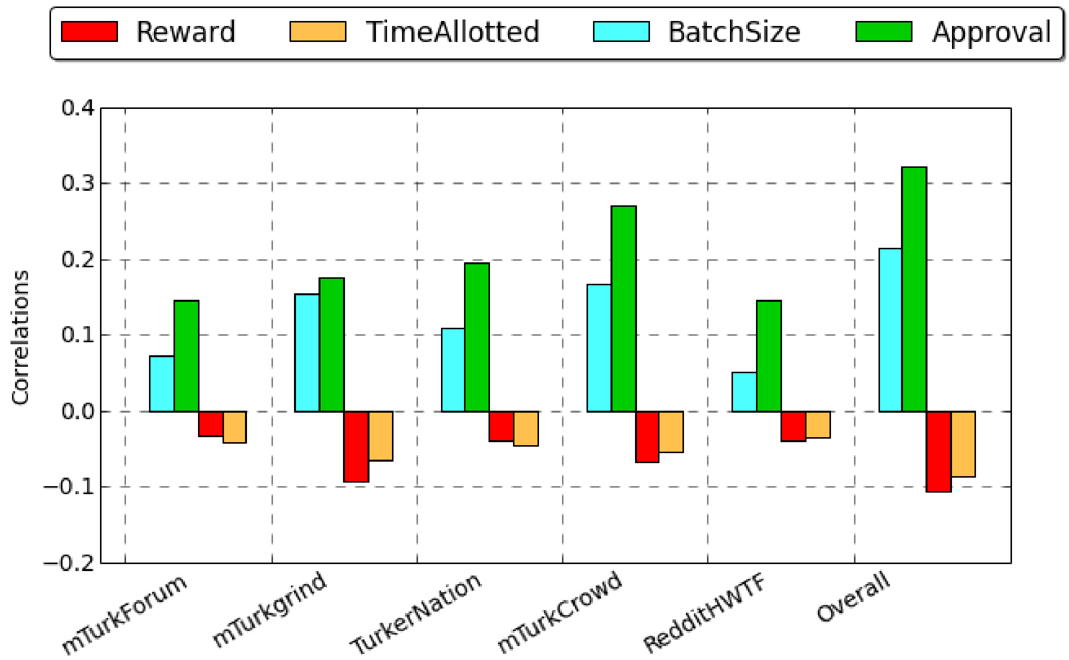


Figure 5.4: Spearman correlation between the properties of HIT group and the number of its mentions in fora.

5.3.4. PROPERTIES OF REQUESTERS

To study the effect of requester properties on discussion we chose to use their Turkopticon ratings. Requester properties are assessed by workers on Turkopticon by their Communicativity, Generosity, Fairness, and Promptness. Using these properties we calculated the Spearman correlation between these properties

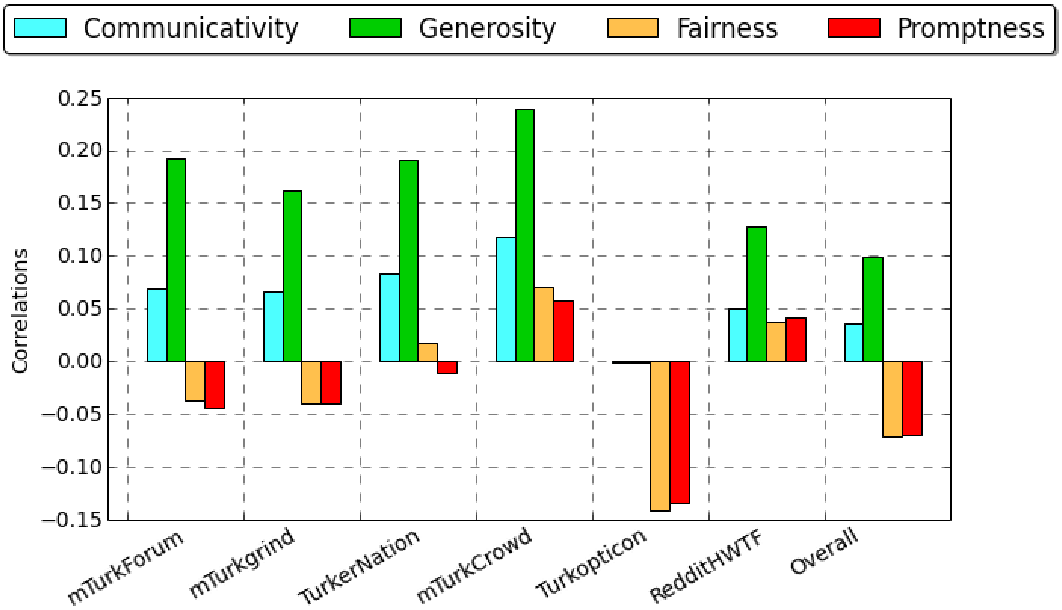


Figure 5.5: Spearman correlation between the properties of requesters and number of requesters' tasks mentioned in fora.

and the amount of times these requesters are mentioned. Figure 5.5 shows the results in a bar chart. From this we can see that most fora tend to mention requesters with high generosity and communicativity. Except for Turkopticon, which tends to mention requester frequently with low fairness and promptness. This is because the goal of Turkopticon is to warn workers of bad requesters. mTurkForum and mTurkGrind tend to follow the overall trend of mentioning requesters with high Communicativity and Generosity ratings, and low Fairness and Promptness ratings. This shows that, on these fora, requesters scoring high in Communicativity and Generosity are mentioned in a positive light, while requesters with low Fairness and Promptness ratings are mentioned as warnings to other workers about the requester or their task. mTurkCrowd and Reddit seem to only have positive correlation between requester attributes and the amount of mentions, which might indicate that workers in these communities are mainly interested in providing recommendations to each other.

5.4. Impact of Communities on the Market

Determining and measuring the impact of worker discussions on the mTurk market allows us to inspect the advantages and disadvantages of having crowdwork discussions. In this section we investigate the relationship between discussions in the worker communities and changes on HITs and requesters. This investigation intends to answer **RQ3** and **RQ4**. We checked to see if discussions on the fora changed the behaviour on the market in some way. First, we determined if executions of tasks that were mentioned are affected in some way. For example, determine if the number of executions of a task increased after it was mentioned on a forum. Then, we determined if requesters responded to the feedback provided by workers. For example, determine if requesters change their task attributes after having received negative feedback.

5.4.1. Impact on HIT Groups

We tested for the presence of a temporal relationship between the mentions in the forum and the change in HITs available of a HIT Group over time using Granger Causality. HIT Groups were selected for testing Granger Causality depending on the availability of consumption data in the dataset. Figure 5.6 shows a plot of the consumption data and the number of mentions of a HIT Group over time. Such time series are tested for Granger Causality.

Table 5.6 shows some statistics of the selection of HIT groups that could be examined. Of the considered HIT Groups we determined that for 9.61% it was possible to establish significant Granger Causality (p-value < 0.05) between the mentions and the task executions over time. As a result of the Granger Causality we obtained two groups of tasks, tasks with Granger causality and those without. This table shows for both groups

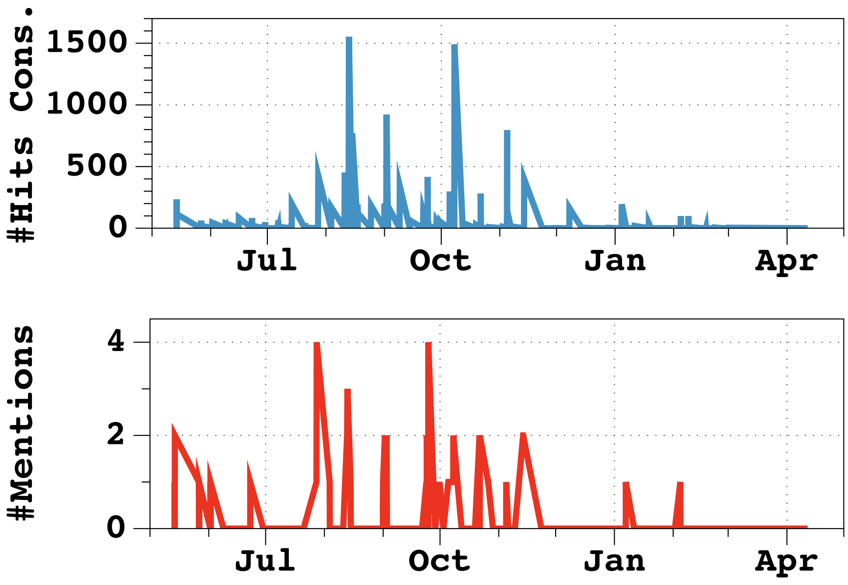


Figure 5.6: An example of HITs consumption in the market vs. #mention by crowd communities.

the mean, median, and standard deviation of the Initial Batch Size, Reward, TimeAllotted, Requirement, and Throughput (the number of HITs in the group completed in a given time interval). From this table we can observe that the tasks with Granger causality tend to have bigger Initial Batch Size, lower Reward, and a slight increase in Throughput. These differences might point to which types of tasks workers actually chose to perform, because mentioning a task on a forum does not necessarily mean that workers will choose to perform the task. This indicates that workers actually perform tasks with a large batch size and low reward.

To show that HIT consumption is indeed affected by mentions in fora, we plotted the HIT consumption data within and hour before and withing an hour after discussions. Figure 5.7 shows this plot. We can see in this plot that the majority of tasks is boosted after discussion. The results show that consumption increased by an average 9.38% after discussions. These analyses show that tasks can positively affected, in terms of throughput, by being mentioned in discussions between workers on fora.

When testing for Granger Causality on the HIT Groups the optimal lag was recorded. Figure 5.8 shows the log-log distribution of the optimal lag across HIT groups: the majority of groups achieve higher correlations for lags lower than 15 minutes. This shows that the discussions in the community can have a quick effect and impact on HIT consumption.

From these result we can see that mentioned tasks can be positively affected by the worker community shortly after being mentioned. This shows the influence that the worker community can have on tasks and the crowdwork market. Having workers discuss tasks is not only beneficial to requesters but also the crowdsourcing platform. Requesters benefit by having their tasks completed more quickly and effectively. The crowdsourcing platform as a consequence has more business, satisfied requesters, and qualified workers.

	Initial Batch Size	**Reward** (cents)	**TimeAllotted** (seconds)	**Requirement**	**Throughput**
With Granger Causality	$424.08, 2 \pm 2556.37$	$45.01, 15 \pm 126.95$	$1.31e04, 2.70e03, \pm 2.62e05$	$0.65, 1 \pm 0.48$	$44.72, 3 \pm 209.25$
Without Granger Causality	$299.88, 1 \pm 1757.33$	$77.21, 30 \pm 338.19$	$1.59e04, 3.60e03 \pm 1.92e05$	$0.62, 1 \pm 0.48$	$43.26, 1 \pm 414.59$

Table 5.6: Descriptive statistics of HIT groups properties. We distinguish between HIT groups for which Granger Causality could have been tested. For each property, we report mean (μ), median (m) and standard deviation (σ): $\mu, m \pm \sigma$. *Requirement* is a Boolean variable.

5.4.2. IMPACT ON REQUESTER BEHAVIOUR

In the previous sections we have seen than worker communities have some influence on the crowdwork market. In this section we look at the change in requester behaviour after ratings were provided by workers (**RQ4**). To do this we adopted the method used by Cheng et al.[12], which measure the effects of social feedback on user posts, to measure what the effect of positive and negative worker evaluations are on requesters.

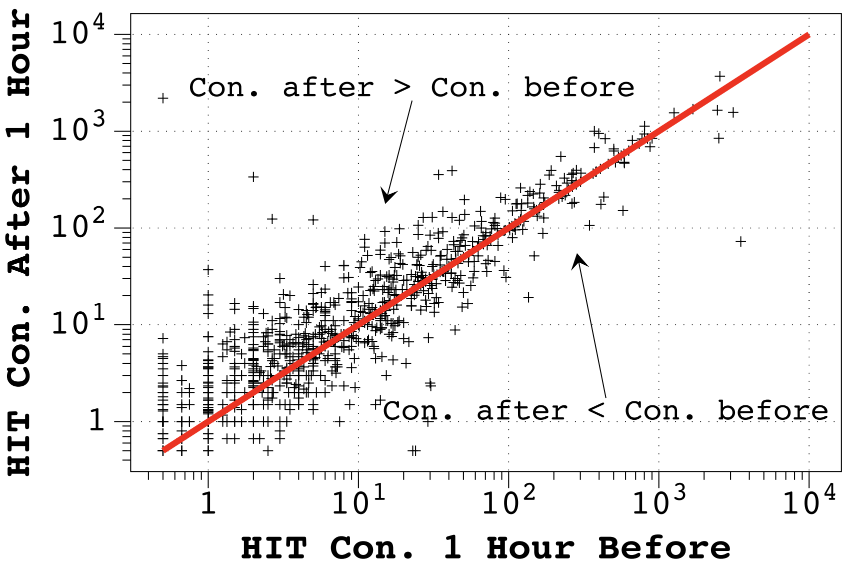


Figure 5.7: #HITs consumption within 1 hour before the discussion in fora and within 1 hour after the discussion.

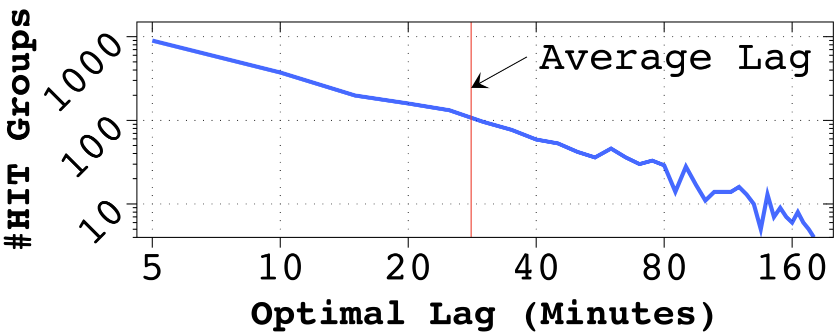


Figure 5.8: Distribution of optimal lags in Granger-Causal HIT groups.

The basic idea is to create two groups of requesters, one containing requesters that were positively evaluated and the other where requesters were negatively evaluated by workers, and then measure how much their tasks and ratings change over time. To create these groups, we first get two requesters with a similar task, where one was positively and the other negatively evaluated, and these requesters were then added to the appropriate group. Tasks were considered similar if they have the same Reward and TimeAllotted attributes. After creating these groups, for each requester the change in rating is calculated by first taking the mean of the future ratings and task attributes, and subtracting that by the initial rating and task attributes. Then for each group, the mean, median and standard deviation of the change in task attributes and rating are calculated.

On Turkopticon there are four attributes workers evaluate (a score from 1 to 5) of the requester: *Fairness*, *Promptness*, *Generosity*, and *Communicativity*. We considered a rating positive if it was higher than 3, and negative if it was lower than 3. Ratings equal to 3 are considered neutral and are ignored because the purpose of this study is to determine the effect of positive and negative evaluations. For each of these ratings we repeated the study of the impact of worker evaluations on requesters, because each of these attributes reflect a different aspect of requester behaviour.

In table 5.7 we show some statistics we obtained from this analysis. This table show the mean, median, and standard deviation for each group that were calculated in the study of the effects of worker evaluations on requesters. This table shows the change in the rating and task attributes (Reward, Batch Size, TimeAllotted, Title Length in number of characters). Table 5.8 shows the Pearson correlation coefficient between task attribute and rating along with the 2-tailed p-value. The Pearson correlation allows us to determine and measure the relationship between task attributes and ratings. We performed this analysis for each requester attribute rating, and results in 3 more tables just like table 5.7. These can be found in the appendix.

From these tables (in this section and the appendix) we can make some observations:

- *Positive evaluated requesters tend to settle to a lower rating*, while *negative rated requesters improve their rating*. This means requester were either overrated or underrated by workers but in time settle on another rating.
- *Positively evaluated requesters increase the batch size in later tasks*, while *negatively evaluated requesters decreased the batch size of later tasks*. We can also see that positively evaluated requesters tend to adjust batch size of future tasks by a greater amount compared to negatively evaluated requesters. This shows that requesters with a high rating are in some way encouraged and then create tasks with larger batch sizes, while requesters with the low rating are in some way discouraged and create tasks with smaller batch sizes. However it is not clear what causes this encouragement and discouragement of requesters, which could be from worker performance of tasks or by worker feedback provided in the fora.
- Both positive and negative evaluated requesters tend to increase the length of the task titles. However with most ratings there is a greater increase in task title length with positively evaluated requesters than negatively evaluated requesters, except in the case of generosity ratings where negatively evaluated requesters increased title length slightly more than positively rated requesters. The increase in title length might indicate that requesters make slight changes in their task such as adding more description using the task title.
- With the fairness and promptness ratings, both positive and negative evaluated requesters increase the *Time Allotted* for workers to complete the task. However, with the communicativity and generosity ratings positively evaluated requesters increase the *Time Allotted* while the negatively evaluated requester decrease the *Time Allotted*. This might indicate that requesters change some aspects of their task to better suit workers.
- From Table 5.7 we observe that requesters starting with a low fairness evaluation tend to improve their rating and increase the *Time Allotted* for their tasks according to mean and Pearson correlation coefficient of the Time attribute. This might hint that the initial rating was low because there was not enough time allotted for the task, and thus was considered unfair by the workers.

		mean	**median**	**std**
Rating	pos	-0.3731	0	0.9810
	neg	1.5299	1.3333	1.6450
Reward	pos	0.0994	0	1.5466
	neg	0.0445	0	0.6192
Batch	pos	268.4750	0	3901.7831
	neg	-175.7569	0	5509.4241
Time	pos	11.2059	0	187.6549
	neg	29.3048	0	555.7716
Title	pos	2.1835	0	10.3412
	neg	1.1089	0	6.3241

Table 5.7: Statistics of the change task attributes and fairness ratings relating to requesters. *pos* refers to the group of requesters that were first positively evaluated and *neg* refers to the group of requesters that were first negatively evaluated. Also provided is a correlation column that shows the correlation between the task attribute and rating from turkopticon.

To better visualize the relationship between change of task attributes and ratings we the following figures. Figure 5.9 shows a scatter-plot along with histograms corresponding to the axes that show the change in rating and task attribute. There is such a plot for each rating, and results in many more plots which are found in the appendix. Most of these figures illustrate that most tasks there is for most task attributes and ratings there is no correlation between them. As shown in Table 5.7 for most attributes there is no correlation between most task attributes and ratings. This means that workers can change task attributes but have no affect on ratings, or that sometimes their rating changes without having changed their task attributes.

		Fairness		Communicativity		Generosity		Promptness	
		corr	p	corr	p	corr	p	corr	p
Reward	pos	0.0592	0.3106	0.1073	0.1133	0.0672	0.1107	0.0994	0.1053
	neg	0.0167	0.7758	0.0166	0.8069	0.0285	0.4988	-0.0463	0.4513
Batch	pos	0.0754	0.1963	0.0544	0.4228	0.0440	0.2974	0.0167	0.7854
	neg	0.0675	0.2481	0.0429	0.5275	0.0372	0.3784	0.0623	0.3109
Time	pos	-0.0203	0.7290	-0.0594	0.3815	-0.0577	0.1709	-0.0297	0.6286
	neg	0.1050	0.0716	0.0342	0.6149	-0.0105	0.8039	-0.0143	0.8157
Title	pos	0.0415	0.4771	0.0749	0.2696	-0.0018	0.9669	0.0637	0.2997
	neg	-0.0246	0.6735	0.0185	0.7859	-0.0266	0.5279	-0.0178	0.7728

Table 5.8: The Pearson Correlation between task attributes and ratings (corr), along with the two tailed p-value. *pos* refers to the group of requesters that were first positively evaluated and *neg* refers to the group of requesters that were first negatively evaluated.

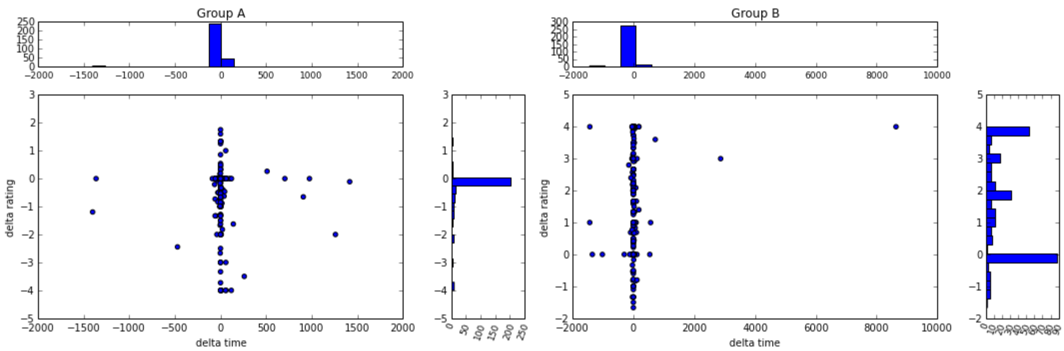


Figure 5.9: Scatterplot between the delta fairness and delta time for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

To see more clearly what these figures mean we have Table 5.9 and 5.10, which shows the percentage of the requesters that experienced changes relating to the ratings and task attributes. This table show the percentage of requesters for each combination of change between rating and task attribute. For example. In a group of requesters, 1.02% of requesters who raised the reward of their tasks experienced a increase in their fairness rating, while 10.51% of requester who raised their reward experienced a decrease in fairness rating. It shows that the majority requesters experience no changes in ratings when modifying task attributes.

From these two tables we can make some interesting observations:

- Requesters that were first positively evaluated (Table 5.9), 10.51% raised their reward but had their fairness rating go down. This also seems the case with other ratings. Of the requesters that change the task reward and experience change in ratings, most raise reward but experience a decrease in ratings. The general trend for this group of requesters is that their ratings go down, however interestingly in these cases it also coincides with the raise in rewards. This shows that worker evaluations for these requesters follow the trend and a high reward does not keep ratings up.

- Requesters that were first negatively evaluated (Table 5.10) and change task attributes experience an increase in rating. However in some cases ratings increase even when task attributes decrease. For example, 11.78% of requesters decreased the amount of time allotted for tasks experiencing increase in communicativity rating, while 8.68% increase the amount of time allotted for tasks also experience an increase in communicativity rating. Again, this shows that ratings follow the general trend of ratings increasing after first being negatively evaluated.

These results further illustrate that correlation could not be established between the changes tasks attributes and the change in ratings of the requesters. This could be due to the fact that these rating are meant for workers (as warnings or recommendations), and not for the requester directly. Also, requesters might not be exposed feedback directly from workers on Turkopticon which shows that the relationship between the requester and worker is through the crowdwork market and not through a third party, such as worker communities platforms.

		Fair up	Fair down	Comm up	Comm down	Gene up	Gene down	Fast up	Fast down
Reward	up	1.02%	10.51%	2.28%	19.18%	5.67%	21.63%	2.25%	12.36%
	down	2.71%	7.46%	0.91%	10.05%	3.19%	16.13%	2.62%	9.36%
Time	up	1.69%	7.80%	0.46%	9.13%	2.66%	11.52%	2.62%	9.36%
	down	0.34%	5.42%	1.37%	7.31%	3.90%	9.40%	0.37%	5.62%
Batch	up	1.69%	5.76%	0.46%	10.05%	2.84%	9.22%	1.12%	8.24%
	down	0.68%	2.03%	1.37%	3.20%	1.24%	5.14%	0.75%	2.25%
Title	up	1.69%	5.08%	0.91%	5.48%	2.13%	8.33%	1.87%	5.99%
	down	0.34%	1.69%	0.0%	5.48%	0.89%	3.55%	0.0%	2.62%

Table 5.9: The percentage of requesters that experience changes in relation between the rating and task attribute. This table is for the group that was first positively rated.

		Fair up	Fair down	Comm up	Comm down	Gene up	Gene down	Fast up	Fast down
Reward	up	29.49%	3.05%	18.26%	3.20%	30.85%	4.61%	27.34%	3.37%
	down	14.24%	1.69%	12.33%	1.83%	12.41%	3.37%	17.23%	1.87%
Time	up	14.92%	1.02%	8.68%	0.91%	12.77%	3.19%	14.61%	2.25%
	down	10.51%	1.69%	11.87%	0.46%	10.28%	1.60%	12.73%	1.87%
Batch	up	16.95%	1.69%	13.24%	2.28%	13.48%	1.77%	19.85%	0.75%
	down	10.17%	1.36%	8.22%	1.37%	8.16%	2.66%	10.49%	1.12%
Title	up	5.42%	1.02%	5.02%	1.37%	7.62%	2.30%	5.99%	0.75%
	down	1.36%	3.68%	1.83%	0.46%	1.95%	0.53%	3.00%	0.37%

Table 5.10: The percentage of requesters that experience changes in relation between the rating and task attribute. This table is for the group that was first negatively rated.

5.5. DISCUSSION

In this section we summarize how we performed our study of crowdwork communities and discuss some of the results from the performed analyses.

First, we analysed the behaviour of worker communities in isolation to see which aspects of crowdwork are discussed in online fora (**RQ1**). By understanding what types of discussions occur among workers, and determining workers involvement in discussions, we can better understand the worker community and their relationship with crowdwork. To do this we annotated a sample of messages posted by workers to create an automated process for classifying all the messages found on the fora within an acceptable range of certainty. For example using the classifier we were able to classify messages that contained a "task comment" with an accuracy of 0.98. We looked at worker community involvement by the number of messages workers post, the lifetime workers have on fora, and relationships between workers indicated by replies to each other. We observed that there is a small group of workers, so called "power workers", post hundreds of messages and have been part of the community for a long time. Using the message replies between workers as an indication of a relationship between workers we were able to analyse this interaction between community members and determine which workers were more central in discussions the community. Interestingly, we observed that there is an overlap between "power workers" and the more central workers that start and/or participate in the community discussions.

Having examined worker discussions, we next look for the influence of the mTurk market on these discussion (**RQ2**). Determining which aspects of the mTurk market triggers discussion by workers helps us understand more about worker activity and behaviour within their community. For example, knowing if tasks with a low reward are more likely to be discussed can point to which types of tasks workers look for on the market. In order to examine the relationship between the mTurk market and community we first linked the tasks and requesters that workers mention on the fora with data gathered from the mTurk market. From this we were able to create an overview for each forum to test the statistical differences across fora. Then, we studied the relationship between the availability of tasks and the discussions on the fora. We observed that there is a temporal synchronicity between available tasks on the market and number of tasks mentioned on the fora. Then, we analysed the properties of the HIT Groups mentioned in the fora and compared them with properties of the non-mentioned ones. We observed statistical differences between most attributes of mentioned and non-mentioned HIT Groups. These statistics showed, for example, that workers preferred to mention tasks with large batch size. Also, of the mentioned HIT Groups and requesters we calculated the correlation between their attributes and the amount of times they are mentioned. Interestingly, the for the mentioned requesters on Turkopticon there is a negative correlation between their attributes and the amount of times they are mentioned, meaning the low scoring requesters are mentioned more often. This points to a very specific behaviour of this particular worker community about what they tend to mention and discuss.

To study the relationship between worker community and mTurk market in more detail, we look at how the worker discussions influenced the mTurk market (**RQ3**). It would follow that workers discussing tasks should influence them in some way, and testing this hypothesis could point to the advantage or disadvantage of workers discussing tasks. For example, mentioned tasks could have more executions compared to non-mentioned tasks. To determine if tasks are influenced by being mentioned on fora, we tested for some temporal relationship between the amount of execution of a task and the number of mentions on the fora over time. This yielded two groups; the influenced tasks, and the non-influenced tasks by being mentioned on the fora. Results show that 9.61% of the considered tasks were influenced by being mentioned on the fora. For both influenced and non-influenced tasks we calculated the mean, median, and standard deviation of their attributes to show the statistical differences. Interestingly, this showed that the influenced tasks have a larger batch size and lower reward. To understand how tasks were influenced we look at the executions 1 hour before and 1 hour after workers discussed the task. Results showed that amount of executions of tasks was boosted after worker discussion. This hints that worker discussion can be beneficial (as far as the requester is concerned).

Another aspect we studied is the relationship between worker discussion and requesters in the mTurk market, specifically the influence of worker duscussions on requester behaviour(**RQ4**). Determining if workers can influence the behaviour of requesters would expand our understanding of crowdsourcing, showing the impact crowdworker communities have in the conceptual crowdwork model shown in figure 3.3. We looked at the change of requester behavior after evaluations provided by workers on Turkopticon. We did this by pairing two similar requesters, where one is initially evaluated positively and the other negatively. This pairing is done so that we obtain two groups that are comparable and can show the general indication of the influence positive and negative evaluation can have on requesters. After pairing of the requesters we were left with two groups, the ones with positive initial evaluation and the other with negative initial evaluation. By calculating for the two groups the mean, median, and standard deviation of their evaluations and task attributes over time we can show the statistical differences between these groups. Results show that initially negative evaluated requesters tend to improve their evaluation indicating that requesters are in some way affected by feedback.

5.6. DESIGN GUIDELINES

In this section we use the results of the analyses in this chapter to construct guidelines relating to task design and crowdsourcing platform design. These guidelines exploit the fact that workers communities exist alongside crowdsourcing platforms, like Figure 3.3, to allow for improvement of crowdwork in general for the workers, requesters, and the crowdsourcing platform. First we look at task design guidelines which allow for construction of tasks that result in an increased chance of community discussions. Then, we look at platform design guidelines. In the new conceptual model provided in Figure 3.3, we see that worker communities are an actor in crowdsourcing alongside the crowdsourcing platform. Having a crowdsourcing platform that takes this model into consideration can improve many aspects of crowdsourcing performance for all actors involved.

5.6.1. TASK DESIGN

From the analysis we performed it can be argued that requesters must try to have their tasks mentioned and positively evaluated by workers. Requesters should have the following aspects in mind when designing tasks.

- Large batch size
  Having tasks with large batch sizes increases the likelihood of workers discussing the task (Section **??**), which should follow from the fact that a large batch size signals task availability to the workers.

- Combination of low reward and time allotted
  Having tasks that workers can perform quickly and earn a fast reward seems to be appealing to workers (Section ??). If the task receives negative feedback, increase the time allotted for the task (Section 5.4.2).

5.6.2. PLATFORM DESIGN

As an important actor in the crowdsourcing model, it is important to optimize the facilitation of providing work from requesters to workers. Our analysis highlights the importance of worker communities, which are an integral actor for this facilitation process. The following points should be considered by crowdsourcing platforms:

- Provide a platform for discussion
  Worker communities are beneficial for the worker, requester, and crowdwork platform (Section 5.3.3). This could be done internally, or there should be a reference towards some external community.

- Advise requesters on task design
  Advising on task design would allow for greater likelihood of tasks being mentioned and result in more business, more qualified workers (Section 5.3.3), etc. Much like the guidelines above, it is also in the best interest of the crowdsourcing platform to improve the quality of the tasks as well as the requesters (Section 5.4.2).

5.7. THREATS TO VALIDITY

The results we produced in this chapter do have some threat to their validity.

The study of worker community might suffer from some selection bias, where the selection of crowdworkers might not be representative of mTurk workers as a whole:

- Previous work as stated that a significant portion (up to %60) of the mTurk workers are part of a worker community [3]. Even though crowdwork communities are popular, it does not guarantee that the majority of users are active members of these communities.

- In Section 5.2.3 we stated that it was a small group of workers that are central in and responsible for most of the community activity, which might also not be representative of the mTurk worker at large.

- In our study we only looked at six popular crowdwork communities, and there are other communities which would provide supplementary data to more accurately represent mTurk workers and overcome selection bias.

- There could also be a selection bias because the workers in the crowdwork community fora that we studied are mainly from the US.

Although we cannot rule out that the presence of these biases, we still consider crowdwork communities an important aspect of crowdsourcing because of the tendency of humans to create communities and the size these communities have achieved. Even if some of the biases would be present, it is still worth considering crowdwork communities for further study.

Another threat to the validity of our work is the linking procedure described in Section 3.3.2. In our study we linked messages to tasks or requester when they were explicitly mentioned by workers, but there are other ways workers can mention them in messages (e.g. mention requesters by name). However, from our own observations when annotating forum messages (13K messages) we observed that such messages have a small (around 5%) and negligible impact on the validity of our results.

6
Conclusions and Future Work

6.1. Conclusion

The main goal of this work is to study the connection between crowdsourcing and worker communities. We aimed at a better understanding of the structure and roles of worker communities and their relationship with the crowdsourcing market and its actors.

The *mTurk Market and Community Tracker* was constructed to gather data from the mTurk market and worker community fora. This allowed us to create a new dataset to supplement data from previous work and study new aspect of crowdwork.

We were able to study six crowdwork community fora: mTurkForum, mTurkGrind, mTurkCrowd, TurkerNation, Turkopticon, and RedditHWTF. Our analysis provides a detailed characterisation of the content of discussion within a forum, and the different role of community members. We found that workers discuss crowdwork by recommending tasks and requesters to each other and also use these fora to socialise. We also identified different types of workers present on the fora. A small core set of users are central to discussion and are responsible of the most activity on the fora, while the rest follow their lead or contribute little to the community.

The relationship between the crowdwork market and discussion also examined. We investigated if the crowdwork market triggers discussion among workers and to which extent. The availability of tasks on the mTurk market were found to influence discussions on the fora. Tasks that were mentioned on the fora had certain properties (large batch size, low reward), indicating that workers tend to discuss certain types of tasks. Requesters were discussed on fora as part of task recommendations for workers, but also could be mentioned as a warning to other workers.

We also analysed the impact of communities' activities in online fora on the crowdwork market. We found that worker communities can indeed influence the market when mentioning tasks to other workers resulting in an increase of task execution. However, workers have a limited influence on requesters' behaviour, at least considering feedback provided through worker community fora.

Finally, our work shows that crowdwork communities can have a tangible benefit for crowdsourcing platforms. Inspired by our analysis we were able to create guidelines for the creation of crowdwork platforms and tasks.

6.2. Future Work

There are many possibilities for future work, to be performed on the collected datasets, or with a different investigation approach.

For instance, more data could be retrieved. One such example is the community data of the Reddit community, which we chose not to obtain because of time and monetary constraints. However, for those interested, the data could be extracted of a Reddit dataset provided here [1]. Other communities could also be included in the analysis, for instance CloudMeBaby.

In our study of crowdworkers we looked at many cross forum differences and observed that some worker communities share some commonalities in behaviour. It would also be interesting to study the overlap be-

[1] https://www.reddit.com/r/datasets/comments/3bxlg7/i_have_every_publicly_available_reddit_comment/

tween these communities, such as workers that are part of multiple communities, in future work to improve our understanding of workers and their behaviour.

The analysis done in this work could be performed for other crowdsourcing platforms and worker communities to provide additional insight into the relationship between crowdsourcing, other crowdwork actors, and their worker communities.

There could be a follow up study on our conclusions that empirically test the community influence, by applying our guidelines when creating tasks as a requester and measuring worker community influences. The goal would be to design tasks so that they are discussed by workers.

Finally, the relationship between the crowdwork market and the worker community requires more study. There are many opportunities for more research to improve out understanding of the relationship.

A

Appendix

A.1. Figures and Tables

		mean	median	std
Rating	pos	-0.8455	0	1.3675
	neg	0.8485	0	1.3905
Reward	pos	0.2685	0	1.8796
	neg	0.0746	0	0.6675
Batch	pos	437.1513	0	4058.1929
	neg	-25.8612	0	6562.2125
Time	pos	11.4807	0	154.1083
	neg	-14.9821	0	156.8657
Title	pos	2.0527	0	12.2798
	neg	1.0241	0	7.7444

Table A.1: Statistics of the change task attributes and communicativity ratings relating to requesters. *pos* refers to the group of requesters that were first positively evaluated and *neg* refers to the group of requesters that were first negatively evaluated. Also provided is a correlation column that shows the correlation between the task attribute and rating from turkopticon.

		mean	median	std
Rating	pos	-0.6770	-0.2583	1.1544
	neg	1.0797	0.7889	1.3542
Reward	pos	0.0894	0	1.1336
	neg	0.1125	0	0.7683
Batch	pos	279.1303	0	3525.5996
	neg	-92.0551	0	3958.2599
Time	pos	6.5691	0	190.7469
	neg	-7.9058	0	145.0551
Title	pos	1.3487	0	8.3632
	neg	1.5680	0	9.1654

Table A.2: Statistics of the change task attributes and generosity ratings relating to requesters. *pos* refers to the group of requesters that were first positively evaluated and *neg* refers to the group of requesters that were first negatively evaluated. Also provided is a correlation column that shows the correlation between the task attribute and rating from turkopticon.

		mean	median	std
Rating	pos	-0.3229	0	0.8253
	neg	1.4903	1.3333	1.5561
Reward	pos	0.1271	0	1.6322
	neg	0.0217	0	0.6298
Batch	pos	333.0551	0	4137.7627
	neg	-119.4386	0	5647.2354
Time	pos	5.9898	0	216.0963
	neg	18.6899	0	556.0728
Title	pos	2.2356	0	9.5368
	neg	0.5937	0	6.8053

Table A.3: Statistics of the change task attributes and promptness ratings relating to requesters. *pos* refers to the group of requesters that were first positively evaluated and *neg* refers to the group of requesters that were first negatively evaluated. Also provided is a correlation column that shows the correlation between the task attribute and rating from turkopticon.

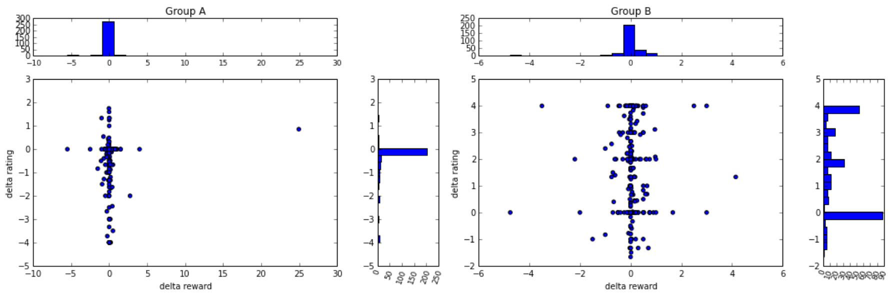


Figure A.1: Scatterplot between the delta fairness and delta reward for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

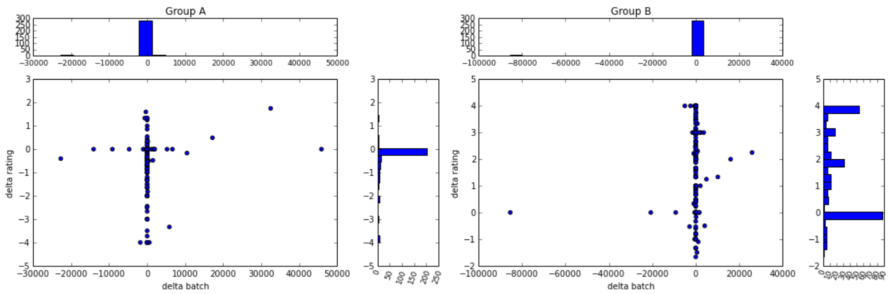


Figure A.2: Scatterplot between the delta fairness and delta batch for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

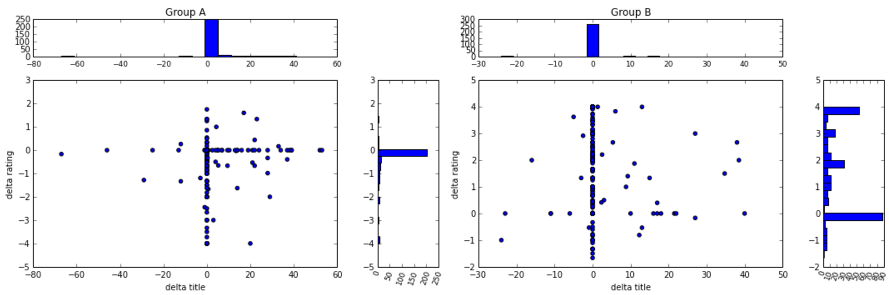


Figure A.3: Scatterplot between the delta fairness and delta title for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

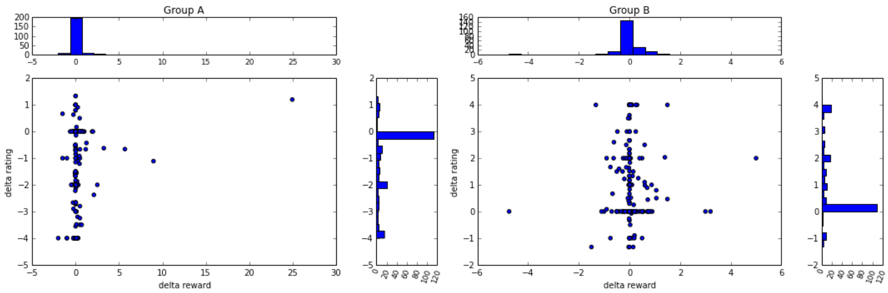


Figure A.4: Scatterplot between the delta communicativity and delta reward for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

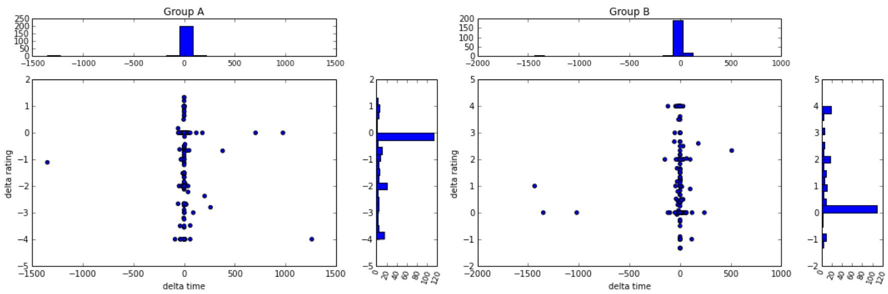


Figure A.5: Scatterplot between the delta communicativity and delta time for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

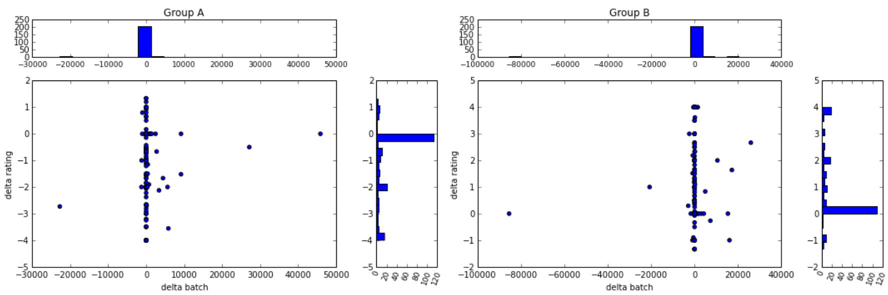


Figure A.6: Scatterplot between the delta communicativity and delta batch for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

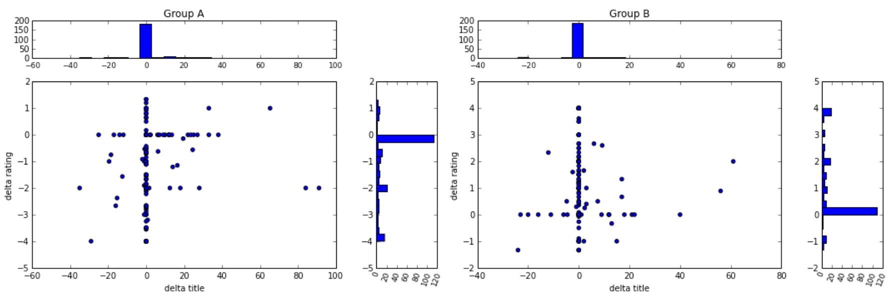


Figure A.7: Scatterplot between the delta communicativity and delta title for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

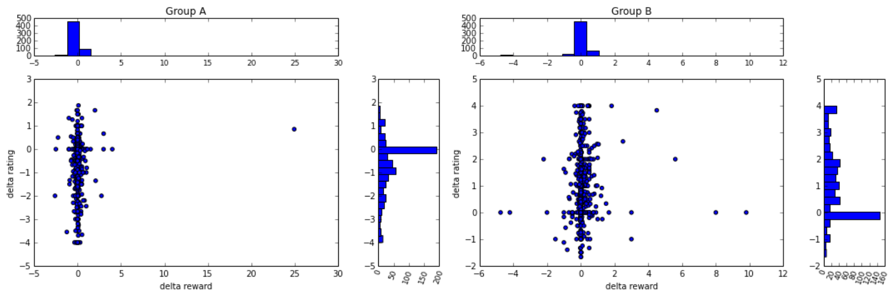


Figure A.8: Scatterplot between the delta generosity and delta reward for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

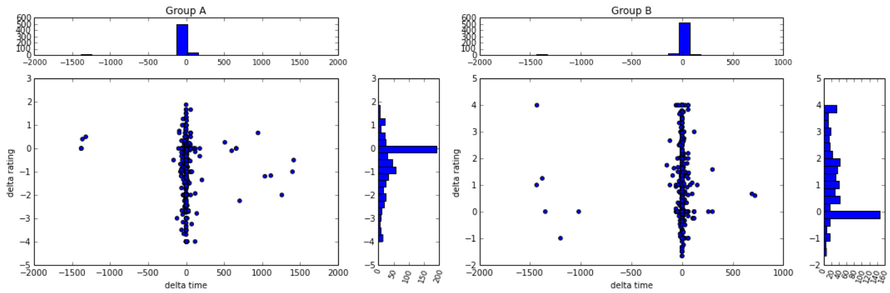


Figure A.9: Scatterplot between the delta generosity and delta time for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

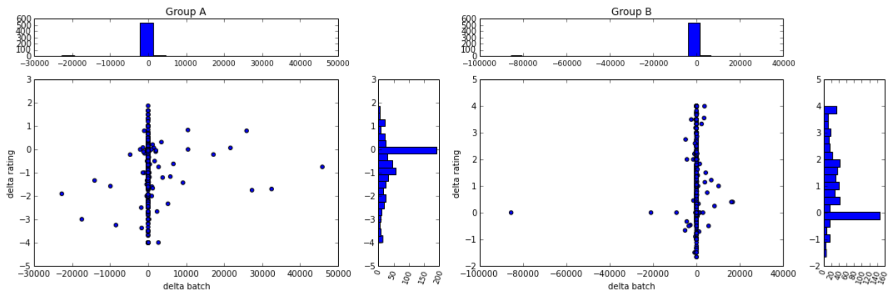


Figure A.10: Scatterplot between the delta generosity and delta batch for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

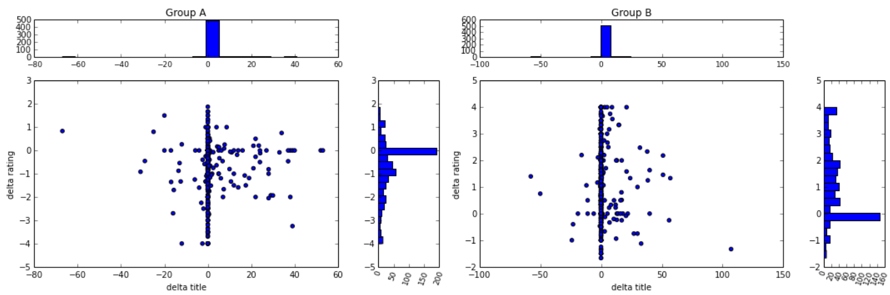


Figure A.11: Scatterplot between the delta generosity and delta title for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

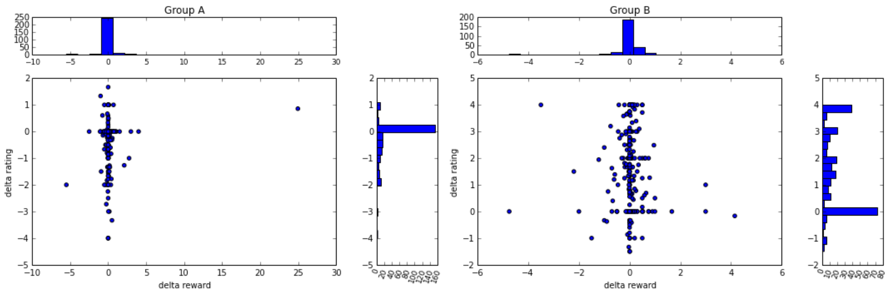


Figure A.12: Scatterplot between the delta promptness and delta reward for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

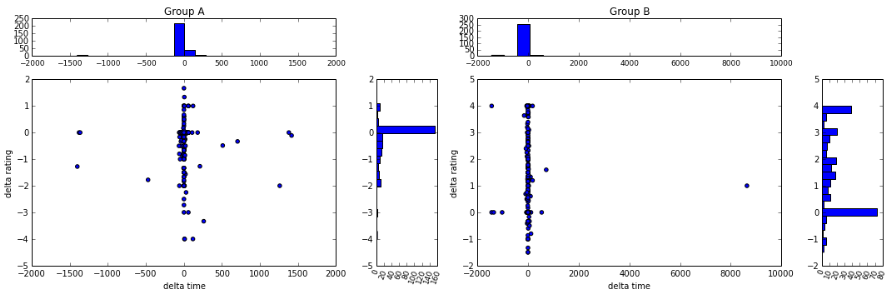


Figure A.13: Scatterplot between the delta promptness and delta time for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

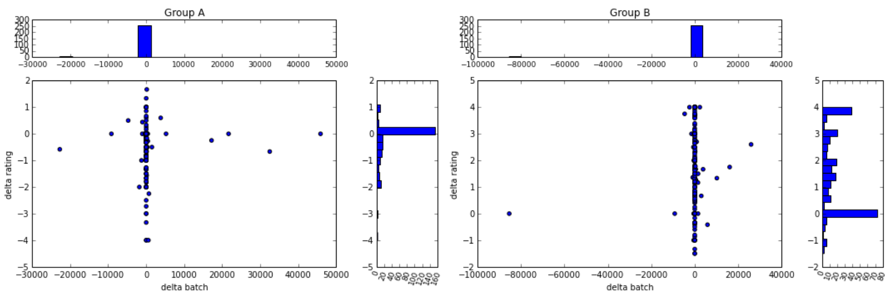


Figure A.14: Scatterplot between the delta promptness and delta batch for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

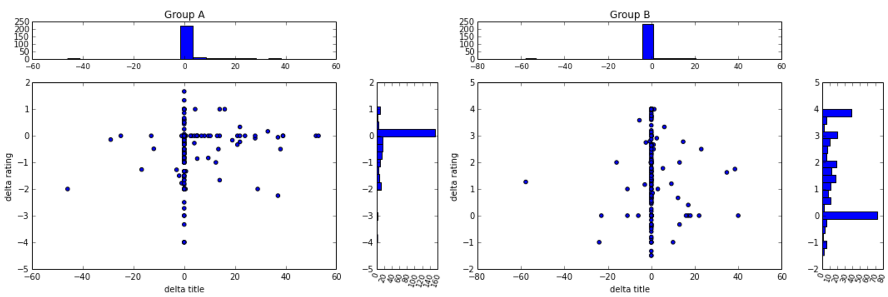


Figure A.15: Scatterplot between the delta promptness and delta title for two groups. Group A are requesters that were first positively rated and Group B are requesters that were first negatively rated.

BIBLIOGRAPHY

[1] M. L. Gray, S. Suri, S. S. Ali, and D. Kulkarni, *The crowd is a collaborative network,* in *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work* (ACM, 2016) pp. 34–147.

[2] D. Martin, B. V. Hanrahan, J. O'Neill, and N. Gupta, *Being a turker,* in *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing* (ACM, 2014) pp. 224–235.

[3] M. Yin, M. L. Gray, S. Suri, and J. W. Vaughan, *The communication network within the crowd,* in *Proceedings of the 25th International Conference on World Wide Web* (International World Wide Web Conferences Steering Committee, 2016) pp. 1293–1303.

[4] D. E. Difallah, M. Catasta, G. Demartini, P. G. Ipeirotis, and P. Cudré-Mauroux, *The dynamics of micro-task crowdsourcing: The case of amazon mturk,* in *Proceedings of the 24th International Conference on World Wide Web* (International World Wide Web Conferences Steering Committee, 2015) pp. 238–247.

[5] J. Howe, *The rise of crowdsourcing,* Wired magazine **14**, 1 (2006).

[6] M.-C. Yuen, I. King, and K.-S. Leung, *A survey of crowdsourcing systems,* in *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on* (IEEE, 2011) pp. 766–773.

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

[8] L. C. Irani and M. Silberman, *Turkopticon: Interrupting worker invisibility in amazon mechanical turk,* in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM, 2013) pp. 611–620.

[9] R. Laplante and M. S. Silberman, *Building trust in crowd worker forums: Worker ownership, governance, and work outcomes,* in *Weaving Relations of Trust in Crowd Work: Transparency and Reputation across Platforms. Workshop at WebScience 2016* (2016).

[10] C. W. Granger, *Investigating causal relations by econometric models and cross-spectral methods,* Econometrica: Journal of the Econometric Society , 424 (1969).

[11] B. Butler, L. Sproull, S. Kiesler, and R. Kraut, *Community effort in online groups: Who does the work and why,* Leadership at a distance: Research in technologically supported work , 171 (2002).

[12] J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec, *How community feedback shapes user behavior,* arXiv preprint arXiv:1405.1429 (2014).

[13] M. Silberman, L. Irani, and J. Ross, *Ethics and tactics of professional crowdwork,* XRDS: Crossroads, The ACM Magazine for Students **17**, 39 (2010).

[14] D. Spencer, *Card sorting: Designing usable categories* (Rosenfeld Media, 2009).

[15] T. K. Ho, *Random decision forests,* in *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on,* Vol. 1 (IEEE, 1995) pp. 278–282.

[16] Y. Zhang, R. Jin, and Z.-H. Zhou, *Understanding bag-of-words model: a statistical framework,* International Journal of Machine Learning and Cybernetics **1**, 43 (2010).

[17] J. Ramos, *Using tf-idf to determine word relevance in document queries,* in *Proceedings of the first instructional conference on machine learning* (2003).

[18] J. M. Kleinberg, *Hubs, authorities, and communities,* ACM Computing Surveys (CSUR) **31**, 5 (1999).

[19] M. E. Newman, *Power laws, pareto distributions and zipf's law,* Contemporary physics **46**, 323 (2005).

[20] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control* (John Wiley & Sons, 2015).