# Chatterbox
## Conversational Interfaces for Microtask Crowdsourcing

Mavridis, Panagiotis; Huang, Owen; Qiu, Sihang; Gadiraju, U.K.; Bozzon, Alessandro

**DOI**
10.1145/3320435.3320439

**Publication date**
2019

**Published in**
UMAP'19

**Citation (APA)**
Mavridis, P., Huang, O., Qiu, S., Gadiraju, U. K., & Bozzon, A. (2019). Chatterbox: Conversational Interfaces for Microtask Crowdsourcing. In *UMAP'19: Proceedings of the 27th Conference on User Modeling, Adaptation and Personalization* (pp. 243-251). ACM. https://doi.org/10.1145/3320435.3320439

# Chatterbox: Conversational Interfaces for Microtask Crowdsourcing

Panagiotis Mavridis
Web Information Systems,
Delft University of Technology
Delft, Netherlands
p.mavridis@tudelft.nl

Owen Huang
Web Information Systems,
Delft University of Technology
Delft, Netherlands
o.huang@student.tudelft.nl

Sihang Qiu
Web Information Systems,
Delft University of Technology
Delft, Netherlands
s.qiu-1@tudelft.nl

Ujwal Gadiraju
L3S Research Center,
Leibniz Universität Hannover
Hannover, Germany
gadiraju@l3s.de

Alessandro Bozzon
Web Information Systems,
Delft University of Technology
Delft, Netherlands
A.Bozzon@tudelft.nl

## ABSTRACT

Conversational interfaces can facilitate human-computer interactions. Whether or not conversational interfaces can improve worker experience and work quality in crowdsourcing marketplaces has remained unanswered. We investigate the suitability of text-based conversational interfaces for microtask crowdsourcing. We designed a rigorous experimental campaign aimed at gauging the interest and acceptance by crowdworkers for this type of work interface. We compared Web and conversational interfaces for five common microtask types and measured the execution time, quality of work, and the perceived satisfaction of 316 workers recruited from the FigureEight platform. We show that conversational interfaces can be used effectively for crowdsourcing microtasks, resulting in a high satisfaction from workers, and without having a negative impact on task execution time or work quality.

## CCS CONCEPTS

• **Information systems** → **Chat**; **Crowdsourcing**; • **Human-centered computing** → **Empirical studies in ubiquitous and mobile computing**;

## KEYWORDS

Microtask Crowdsourcing, Chatbots, Conversational Agents

**ACM Reference Format:**
Panagiotis Mavridis, Owen Huang, Sihang Qiu, Ujwal Gadiraju, and Alessandro Bozzon. 2019. Chatterbox: Conversational Interfaces for Microtask Crowdsourcing. In *27th Conference on User Modeling, Adaptation and Personalization (UMAP '19), June 9–12, 2019, Larnaca, Cyprus.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3320435.3320439

## 1 INTRODUCTION

Messaging applications such as Telegram, Facebook Messenger, and Whatsapp, are regularly used by an increasing number of people, mainly for interpersonal communication and coordination purposes [23]. Users across cultures, demographics, and technological platforms are now familiar with their minimalist interfaces and functionality. Such popularity, combined with recent advances in machine learning capabilities, has spurred a renewed interest in conversational interfaces [32], and *chatbots*, i.e. text-based conversational agents that mimic a conversation with a real human to enable conversational, information seeking [1, 2, 30], and transactional tasks [7, 8, 33]. The growing popularity of conversational interfaces has coincided with flourishing crowdsourcing marketplaces.

Microtask crowdsourcing allows the interaction with a large crowd of diverse people for data processing or analysis purposes. Examples of such microtasks include audio/text transcription, image/text classification, and information finding. Microtask crowdsourcing is commonly executed by means of dedicated Web platforms (e.g. Amazon Mechanical Turk, FigureEight), where all the published microtasks are publicly presented to workers. Upon the selection of their preferred microtasks, workers are typically directed to a webpage served by the platform or hosted on an external server by the task requesters. Based on the task design, workers can provide their input by means of standard (e.g. text, dropdown, and multiple choice fields) or custom (e.g. drawing tools) Web UI elements. Recent work has shed light on the importance of task design choices made with respect to user interface elements; and on how such choices can influence the quality of work produced and satisfaction among workers [9].

Although conversational interfaces have been effectively used in numerous applications, the impact of conversational interfaces in microtask crowdsourcing marketplaces has remained unexplored. We aim to address this knowledge gap in this paper. We investigate the suitability of conversational interfaces for microtask crowdsourcing by juxtaposing them with standard Web interfaces in a variety of popularly crowdsourced tasks. Lowering the entry barrier for workers to participate effectively in crowdsourcing tasks is an important step towards securing the future of crowd work [18].

The availability of effective automated text-based conversational interfaces – as an alternative to the traditional Web UI – could broaden the pool of available crowd workers by easing their unfamiliarity with the interface elements. Messaging applications are reported to be more popular than social networks [28], and we argue that such familiarity with conversational interfaces can potentially breed more worker satisfaction.

*UMAP '19, June 9–12, 2019, Larnaca, Cyprus*

ACM ISBN 978-1-4503-6021-0/19/06.
https://doi.org/10.1145/3320435.3320439

**Original Contributions**. Our goal is to further the understanding of how text-based conversational interfaces could serve as an alternative to the standard Web interfaces typically used for microtask crowdsourcing. We seek answer to the following questions:

**RQ1**: To what extent can text-based conversational interfaces support the execution of different types of crowdsourced microtasks?

**RQ2**: How do different types of UI input elements in conversational interfaces affect quality-related outcomes in microtasks?

We carried out experiments to gauge the interest and acceptance of automated, text-based conversational work interfaces by crowd workers, while assessing their performance within different task types. We recruited workers from the FigureEight microwork platform, and implemented a conversational interface based on the popular *Telegram* messaging platform. We addressed five typical microtask types (information finding, human OCR (captcha), speech transcription, sentiment analysis, image annotation) spanning content types (text, image, audio) and UI elements (free text, single and multiple selections, image segmentation). For each task type, we implemented both Web and conversational interfaces.

We addressed **RQ1** by comparing the execution time, quality of results, and satisfaction of workers who used the standard Web interface with those who used the conversational interface. To answer **RQ2**, we compared different implementations of conversational UI elements for single and multiple input selections in microtasks. Results show that the conversational interfaces are positively received by crowd workers, who indicated an overall satisfaction and an intention for future use of similar interfaces. In terms of performance, tasks executed using the conversational interfaces took similar execution times, and yielded comparable output quality.

## 2 BACKGROUND AND RELATED WORK

A conversational agent is a software programmed to automatically interpret and respond to requests expressed in natural language, so to mimic the behavior of a human interlocutor. *Chatbots* are a class of conversational agents that prevalently use text as a interaction medium. While research on chatbot systems dates back to the 1960s, the growing popularity of messaging platforms (especially on mobile devices) is sparking new interest both in industry and academia. In addition to the traditional focus on conversational purposes, recent work in Information Retrieval addressed informational task. For instance, Vtyurina et al. [30] investigate the use of a chatbot system as an alternative for search engines to retrieve information in a conversational manner. Avula et al. [1, 2] explored the adoption of chatbots for collaborative search and content recommendation. Vaccaro et al. [29] investigated the use of chatbot for styling personalization.

### 2.1 Crowd-powered Conversational Systems

Research in (microtask) crowdsourcing addressed the integration of crowd work platforms with text-messaging and chatbots systems, mostly to train the machine learning components of the conversational agent (e.g. intent recognition), or to substitute artificial intelligence for conversation management purposes [20].

An early example of chat-based crowdsourcing system is *Guardian* [15], a semi-automated chat system that helps wrapping up Web-APIs into spoken dialog systems. In a follow up work [14] the same authors proposed *Chorus*, a system that allowed end-users to chat directly with crowd workers. Crowd workers would be able to propose and vote on candidate responses, that would be then aggregated and sent to the user. To facilitate the workers to cast votes on candidate responses, a web-based conversational interface (resembling an online chat room) was used. The interface made use of buttons to *upvote* or *downvote* a candidate response. Evorus [13] is an evolution of Chorus where conversation automation is obtained by adding, learning, and improving automated responses using past information gained from the crowd. *Calender.help* [5] is an email-based personal assistant, with some automation ability to schedule meetings at the time which fits all the participants. The system automatically creates and delivers coordination microtatsks using the Microsoft bot frameworks. Liang et al. [22] propose CI-Bot, an early prototype of a conversational agent as question and answering system. The authors conducted a pilot experiment and reported good performance for image labeling tasks. InstructableCrowd [12] is a conversational agent that can crowdsource "trigger-action" rules for IF-THEN constructs, e.g. to set an alarm or an event in a calendar application. Workers used a web-based interface similar to the chat room proposed in [13, 14].

These systems demonstrated the technical feasibility of application-specific microtask execution through chatbots. Our work has a broader scope, as it addresses the execution of different classes of microtask crowdsourcing, with a principled comparison with traditional Web interfaces aimed at evaluating chatbots as a generic medium for crowd work.

### 2.2 Mobile Interfaces for Crowdsourcing

Previous work addressed the problem of ubiquitous and opportunistic microtask crowdsourcing through user interfaces for mobile devices, either in an humanitarian[1] or academic [19, 21, 26, 31] setting. mCrowd [31] is a platform used to perform crowd sensing tasks with native mobile applications. [27] experiment with different mobile interfaces to perform crowdsourcing on multimedia microtasks. MobileWorks [26] is a mobile crowdsourcing platform designed for the web browser of lower-end phones, to enable the execution of crowdsourcing tasks also by people with limited connectivity. In a similar spirit, Kumar et al. [21] address the dynamics of mobile crowdsourcing for the developing countries. They implement and test both a native application that supports generic crowdsourcing tasks and also a system that can handle tasks with simple sms exchange. To evaluate the system they measure the impact of different screen sizes into the ease of use of their interface as well as the task execution time and quality of different types of tasks. They found correlation between screen size and quality of work, especially for tasks such as video annotation, human OCR and translation. Image annotation tasks were the highest performing. In [6], authors set up an experiment with four different crowdsourcing platforms (FigureEight, formerly known as CrowdFlower, was not included) in order to check the difficulty and execution time of commonly performed tasks and input controls. Authors experienced technical and usability difficulties with straightforward mapping from Web user interfaces to mobile ones, and therefore propose a number of adaptations for their experts when it came to the evaluation (e.g. avoid long descriptions, minimise scrolling).

While our work addresses a different class of interaction systems (chatbots vs. native or web-based mobile interface), the publications and systems mentioned above share our ambition and vision for democratization and scaling up of crowd work. The results obtained

[1] e.g. Ushahidi: https://www.ushahidi.com/

from their analysis of mobile task types and designs [6, 19] features some interesting commonalities and difference with our findings, as discussed in the Evaluation Section.

### 2.3 Lowering Barriers for Participation in Microtask Crowdsourcing

Narula et al. noted that microtask marketplaces were often inaccessible to workers in developing countries, and introduced a mobile-based crowdsourcing platform called *Mobileworks* for OCR tasks, thereby lowering a barrier for participation [26]. Khanna et al. studied usability barriers that were prevalent on AMT, which prevented workers with little digital literacy skills from participating and completing work on AMT [16]. Authors showed that the task instructions, user interface, and the workers' cultural context corresponded to key usability barriers. To overcome such usability obstacles on AMT and better enable access and participation of low-income workers in India, the authors proposed the use of simplified user interfaces, simplified task instructions, and language localization. Several prior works have stressed the positive impact of good task design, clear instructions and descriptions on the quality of work produced to usher effective participation from crowd workers [11, 17, 24]. Complementing these prior works, we propose to use conversational interfaces that people may be generally more familiar with as an alternative to standard web interfaces to lower participation barriers.

## 3 EXPERIMENTAL DESIGN

We considered five types of microtasks that are typically completed by crowd workers in microwork crowdsourcing marketplaces. We selected these tasks both to stress the diversity of evaluated content types (text, images, audio), and the diversity of UI elements used to perform the tasks. For the sake of reproducibility, the complete list of tasks (and related data) is available for download on the companion webpage.[2]

*Information Finding.* Workers are tasked to find specific relevant information from a given data source [10]. We opted for business-related information available on the Web, to facilitate retrieval and minimize task execution delays due to hard-to-find information. We used the first 17 business records listed in the `Yelp dataset`[3]. From these 17 records, we created 50 task objects by randomly removing three of the following fields: *name*, *address*, *city*, *state*, *postal code* and *stars* (i.e. the business rating). To prevent ambiguity, the *name* and *postal code* were never jointly removed from the same business record. The workers' task was to use commercial search engines to retrieve the missing information from the business record, and to provide it as free text in three separate fields.

*Human OCR (CAPTCHA).* This is a media transcription task [10], where workers were required to transcribe the text contained in a CAPTCHA image. We generated[4] 50 distinct CAPTCHAs of four characters, containing only digits and letters (i.e. excluding special characters and symbols such as punctuation marks, currency symbols, etc.).

*Speech Transcription.* In this audio transcription task, workers were asked to transcribe recordings of English speech retrieved from Tatoeba[5]. We selected 50 distinct recordings, with length ranging from 2 to 8 seconds, and asked workers to type the content of the short speech.

*Sentiment Analysis.* In this task, workers were asked to assess the sentiment of user reviews. We relied again on the `Yelp` dataset, and selected 50 reviews. To maintain sufficient diversity on selected businesses, we selected a maximum of three reviews per business. The length of the selected reviews varied, ranging from several sentences to whole paragraphs. Workers were asked to judge the *overall sentiment* of a review as *Positive*, *Negative*, or *Neutral*. An additional *Unsure* option was provided, to address annotation uncertainty and prevent forced choices.

*Image Annotation.* This is another data enhancement task where the goal is to determine the categories of the food items contained in an image. The options included: *Eggs*, *Fish*, *Meat*, *Vegetables*, *Fruits*, *Cheese*, *Mushroom*, *Grain*, and *Sweets*. In case the image did not contain any food category that was applicable, workers were requested to only select a *Non-food* option. We used 50 distinct images from the `Yelp` dataset.

### 3.1 Work Interfaces

We focused on three types of UI elements that are required to perform the task types investigated in our experiments as shown in Table 1; (1) *Free Text*, to input text data retrieved from the Web, annotations about a data object, or transcriptions from images and sound; (2) *Single Selection from List*, for single-class classification (*Sentiment Analysis*); and (3) *Multiple Selection from List*, for multi-class classification (*Image Annotation*).

The following sections describe and justify the interface designs adopted in our work. All the implemented interfaces are available on the companion webpage for reference.

**Table 1: Summary of considered UI elements, and their implementation in web and conversational interfaces.**

| *UI Element* | Web | Conversational |
|---|---|---|
| *Free Text* | Single/Multi line text | Message |
| *Single Selection* | Radio buttons | Single Button |
| *Multiple Selection* | Checkbox(es) | Multiple Buttons |

#### 3.1.1 Standard Web Interface.

The Web interface was developed on the FigureEight platform, which provides a standardized way to specify work interfaces in an HTML-like format. We decided to use only standard interface elements, that are typical of crowdsourcing tasks on FigureEight, to elicit normal interactions of workers with the web interface.

Figure 1 depicts a one-to-one comparison of the Standard Web Interface tasks versus the Conversational Interface tasks.

We can see the screenshots of the developed Web UIs corresponding to each of the 5 task types. FigureEight provides two types of *Free Text* UI elements: `single line` text input and `multi-line` text input. The former type is used in the *Information Finding* and *Human OCR* tasks, as worker were asked to provide short input text (e.g. business name, city, address). The latter type is used in the *Speech Transcription* task, workers had to input short sentences from the processed audio. The *Single Selection* element needed for the *Sentiment Analysis* task has been implemented using `Radio Buttons`, as customary for this type of tasks; while the *Image Annotation* tasks used the `Checkboxes` UI element for *Multiple Selection*. When the task entailed multiple

[2] https://sites.google.com/view/umap2019chatbotmicrowork
[3] Yelp dataset: https://www.yelp.com/dataset
[4] CAPTCHA generator: https://pypi.org/project/captcha/
[5] https://tatoeba.org/eng/audio/index

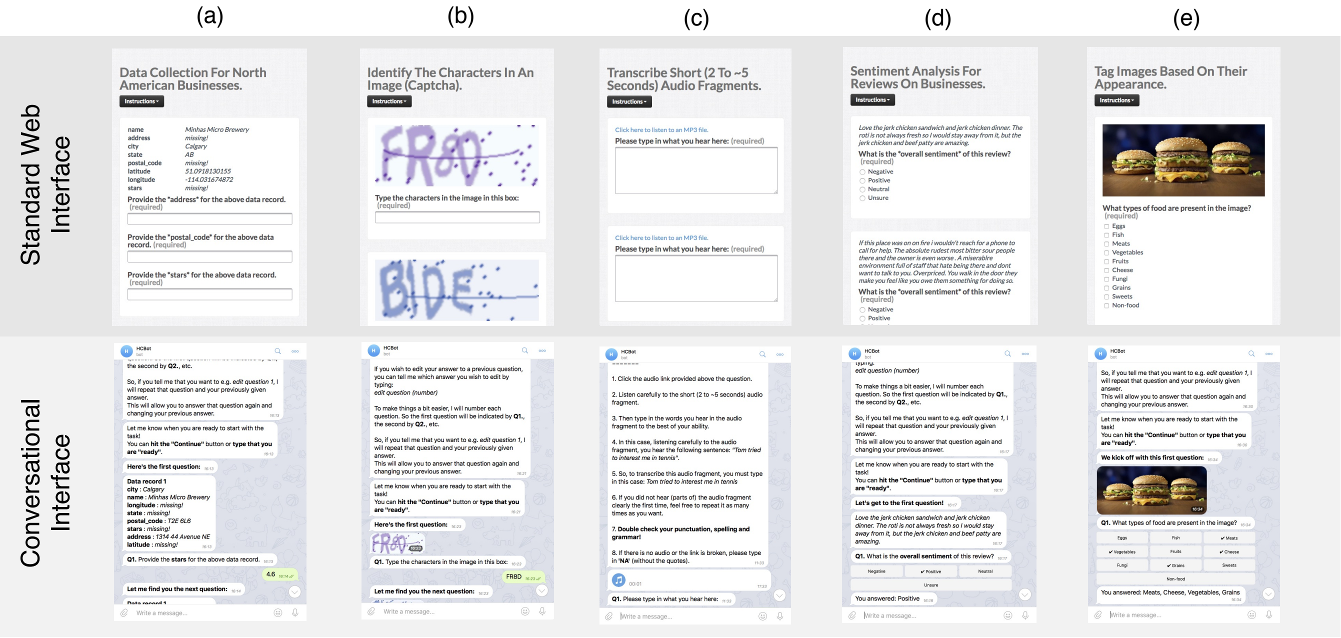

(a)
(b)
(c)
(d)
(e)
Standard Web Interface
Conversational Interface
Data Collection For North American Businesses.
Identify The Characters In An Image (Captcha).
Transcribe Short (2 To ~5 Seconds) Audio Fragments.
Sentiment Analysis For Reviews On Businesses.
Tag Images Based On Their Appearance.
HCBot

**Figure 1: In this Figure we depict different tasks (a, b, c, d, e) and how they look from a Standard web (top) versus a conversational (bottom) interface perspective. The different types of tasks depicted: a) Information Finding, b) Human OCR, c) Speech Transcription, d) Sentiment Analysis, e) Image Annotation. Best viewed digitally.**

annotations (e.g. sentiment analysis, image labeling), content items and their respective input elements were presented in a sequence, to be navigated top-to-bottom within the same page.

*3.1.2 Conversational Interface.* To resonate with popular conversational interfaces, we designed and implemented our conversational interface in the `Telegram`[6] messaging platform.

The interface comprises two main modules: 1) a *conversation management* module, responsible for aligning the status of the task execution with the status of a conversation, and for supporting navigation within the conversation ; and 2) an *input management* module, responsible for rendering the content associated to a task, and the UI elements required to allow and control user input.

Microtask crowdsourcing user interfaces are typically designed to be minimalistic and easy to use, to enable fast and effective work execution [18]. We shared the same design principle in the creation of the *conversation management* module, which consists of five simple states as illustrated in Figure 2. Figure 3 shows a brief example of the conversational flow in the chat interface.

***1)*** At the beginning of the task execution stage, a *chatbot* that drives the conversation, prompts the worker with messages containing task instructions, including an explanation of the task at hand, and examples of how input could be provided. ***2)*** Once no more annotations are pending in the task, the chatbot prompts the next question to the worker (content plus UI elements), and waits for the worker's response. ***3)*** Next, the answer provided by the worker is validated, with positive feedback if the answer is acceptable, or a re-submission sequence if the answer not valid. ***4)*** When no more annotations are pending, workers are shown their answers for review; and can ***5)*** re-process a previously submitted answer.

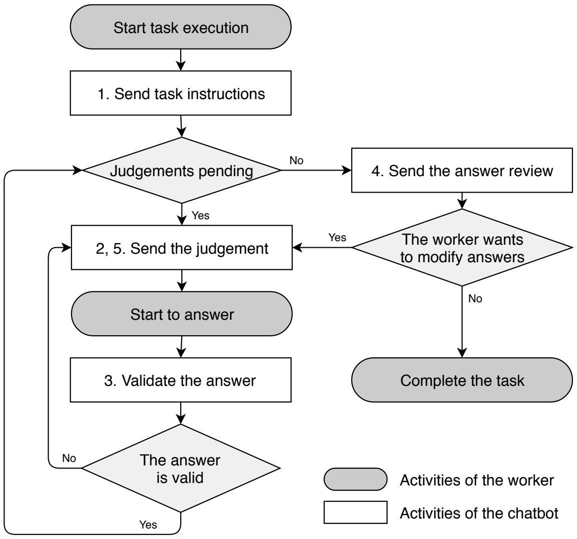


**Figure 2: Conversation management logic.**

The *input management* component is built upon the standard *message* UI element, used by the workers and the chatbot to exchange information. Traditional text messaging systems only allow for alphanumeric content to be exchanged and rendered.

[6] https://core.telegram.org/bots

Systems like *Telegram* allow for richer content, which include: ***1)*** *multimedia* content (images, videos, sound). ***2)*** *Interactive applications* (e.g. games), hosted on third party servers but rendered and accessible within the messaging application. ***3)*** *Custom keyboards*, which show predefined inputs, rendered textually or visually; notice that keyboards are complementary to the standard *message* element: the user can also simply type an abbreviated input (a single alphabet letter) used as a code associated with a pre-defined key option. And ***4)*** *commands*, i.e. instructions sent by the user to change the state of the chatbot (e.g. to start a new working session, or end an existing one).

Figure 1 depicts screenshots of the developed conversational interfaces. The design of both the interfaces and the interaction flows for each task type has been iterated and validated several times by the authors, through experiments with researchers and students from the research group. The *information finding* (a), *human OCR* (b), and *speech transcription* tasks (c) use a simple *message* element, where validation is performed by simply rejecting empty answers. The *sentiment analysis* (d) and *image annotation* (e) tasks were implemented with custom keyboards, allowing for (respectively) the single or multiple selection of predefined answers rendered as buttons associated with some option codes. Here, validation is performed by ensuring that only one button, option code, or content corresponding to an option is given. With custom keywords, workers could express their preference textually (with answers separated by whitespace or commas), using the option codes associated with the button, or by pressing the buttons. We use 4 custom keyboards configurations: 1) *Button-only Custom Keyboard*: Worker can select any button provided; 2) *Text-Only Custom Keyboard*: Worker can only type to provide its answer; 3) *Code-Only Custom Keyboard*: Worker can only type a letter to provide the answer from a predefined list; and 4) *Mixed Custom Keyboard*: a worker can either select a button, type the full answer or the abbreviated code that corresponds to the answer (a single letter).

In all the tasks types that we considered, the chatbot prompts the worker with the item to evaluate by rendering text (the business record to complete), images (the CAPTCHA and the food image), or speech (the audio to transcribe).

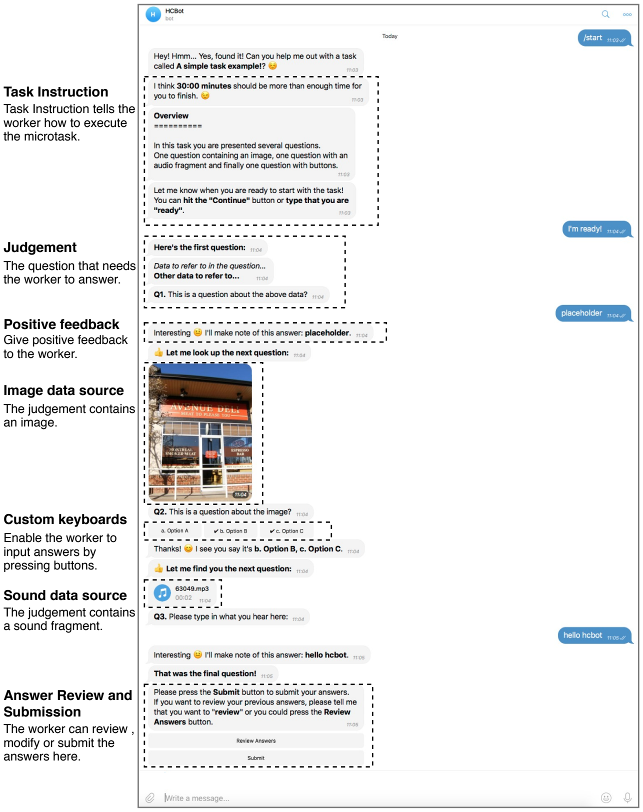
## 3.2 Experimental Conditions

To answer **RQ1**, we designed 12 experimental conditions, with working interface type (Web, Conversational) and task type as independent variables, and the *Mixed Custom Keyboard* configuration for the *Sentiment Analysis* and *Image Labeling* conversational interfaces. As observable from Figure 1, the instructions at the beginning of the conversational task are relatively long, thus possibly affecting the task execution time.

To account for this, we include 6 additional experimental conditions where the conversational interface has task instructions partially hidden (workers are only presented with a brief overview of the task), and workers could instruct the chatbot through specific commands to display more detailed instructions (i.e. an example and its steps, and also inquire about how to edit a previously given answer). With **RQ2**, we tested the 3 *Custom Keyboard* configurations with the *Sentiment Analysis* and *Image Annotation* tasks, thus adding 6 additional experimental conditions.

## 3.3 Task Assignment and Execution

On FigureEight (F8), we set up two types of jobs: *Web* jobs and *Conversational* jobs, where the latter included the string `*|*Requires Telegram*|*` in their title, to suggest the presence of a technical requirement for their execution.



HCBot
bot

Today

/start 11:03

Hey! Hmm... Yes, found it! Can you help me out with a task called **A simple task example!?** ☺ 11:03

**Task Instruction**
Task Instruction tells the worker how to execute the microtask.

I think **30:00 minutes** should be more than enough time for you to finish. 😉 11:03

**Overview**
==========

In this task you are presented several questions.
One question containing an image, one question with an audio fragment and finally one question with buttons. 11:03

Let me know when you are ready to start with the task!
You can **hit the "Continue"** button or **type that you are "ready"**. 11:03

I'm ready! 11:04

**Judgement**
The question that needs the worker to answer.

**Here's the first question:** 11:04

*Data to refer to in the question...*
**Other data to refer to...** 11:04

**Q1.** This is a question about the above data? 11:04

placeholder 11:04

**Positive feedback**
Give positive feedback to the worker.

Interesting 😐 I'll make note of this answer: **placeholder.** 11:04

👍 **Let me look up the next question:** 11:04

**Image data source**
The judgement contains an image.

**Q2.** This is a question about the image? 11:04

**Custom keyboards**
Enable the worker to input answers by pressing buttons.

Thanks! 😊 I see you say it's **b. Option B, c. Option C.** 11:04

👍 **Let me find you the next question:** 11:04

**Sound data source**
The judgement contains a sound fragment.

63049.mp3
00:02 11:04

**Q3.** Please type in what you hear here: 11:04

hello hcbot 11:05

Interesting 😐 I'll make note of this answer: **hello hcbot.** 11:05

**That was the final question!** 11:05

**Answer Review and Submission**
The worker can review , modify or submit the answers here.

Please press the **Submit** button to submit your answers.
If you want to review your previous answers, please tell me that you want to "**review**" or you could press the **Review Answers** button. 11:05

**Figure 3: An example showing the conversational interface developed for our experimental study.**

Web jobs were completely performed within the F8 platform, with the standard F8 workflow and task assignment strategy.

Conversational jobs had a different flow: upon job selection, workers were informed that logging into Telegram was a requirement for participation. Additional instructions on how to register a Telegram account (if necessary) were also provided on an external web-page through a link. Several preview images were provided to inform workers about the nature of the task, and a short survey inquired about their working platform. We did not employ fingerprinting techniques to detect the digital work environment of workers to preserve worker privacy. Workers were informed that no personal information (e.g. names or phone numbers) would be stored, and that they would be allowed to withdraw from the experiment at any point in time.

To facilitate the assignment of tasks in Telegram, we redirect users via a URL to Telegram. According to their working environment, the worker could 1) have been redirect to a Web client version of Telegram; or, if the worker had a native Telegram client installed, 2) to the native Telegram application. Task assignment was performed dynamically, with a round robin policy on the content to be processed. A click of the *Submit* button commanded the finalization of the task, which resulted in a randomly generated validation token to be used in F8 to fully complete the task and receive payment. Workers were also asked to indicate their intention to perform a similar task again in Telegram (yes/no)[7], and to optionally provide a comment about their working experience.

### 3.4 Evaluation Metrics

The dependent variables in our experiment are *Execution Time*, *Answer Quality*, and *Workers Satisfaction*. Ground truth and evaluation data is available on the companion Web page.

*Execution Time* is measured as the time (in seconds) between the start and the submission of a task. In the web interface, this is calculated as the time from when the F8 task is initiated, up to the moment the *Submit* button is clicked. In the conversational interface, this is calculated as the time difference between a click event on the *Start* button, and a click event on the *Submit* button.

*Answer quality* is measured by comparing the worker answers with ground truth *Sentiment Analysis* and *Image Annotation*. For the *Information Finding* and *Speech Transcription* task, workers results were manually inspected by the authors; simple syntactical and grammatical errors were tolerated. For the *human OCR* task, we compared the entire answer to the label of the CAPTCHA, disregarding errors with capitalization. To judge whether a worker had answered correctly for the *Image Annotation* task, we marked an answer as correct, as long as it contained at least one correct annotation, and no more than two wrong annotations.

*Workers Satisfaction* of both web and chatbot tasks is measured by default task ratings on F8 (workers will be re-directed back to F8 when they submit the answer on Telegram) after workers finish the task. Furthermore, for the chatbot tasks, the optional comments are left at the end of the chatbot task to let workers give their personal opinions.

## 4 EVALUATION

The experiments were performed recruiting workers from the F8 microtask crowdsourcing platform. As the main objective of this work is to understand if text-based conversational agents can enable microtask crowdsourcing, we did not condition the participation of workers to pre-existing quality levels, nor did we run qualification tests. Each experimental condition has been deployed as a separate job in F8 (FigureEight). Each job contained 50 task instances, totaling 1200 executions for the whole experiments. Each instance has been compensated 0.15¢. *Information Findings* tasks contained 1 business record; *Human OCR* tasks contained 5 distinct CAPTCHAs, *Speech Transcription* tasks contained 3 audio samples; *Sentiment Analysis* tasks contained 3 reviews; *Image Annotation* tasks contained 3 images each. The distribution and frequency of objects in Web and Chatbot tasks were identical. Workers could only execute one task instance per available job. Web and Chatbot jobs were deployed on different dates, to maximize the chance of obtaining disjoint worker populations. The statistical tests that we performed to test the significance are always Mann-Whitney-Wilcoxon pair-wise significance test.

316 distinct workers executed at least one task ($\mu$ = 3.886, $\sigma = 2.4941$, $median = 2$). 31 workers executed both web and chatbot jobs. 12.2% of the workers self-reported that they performed chatbot jobs with a mobile device. To eliminate the influence of malicious behavior, a manual inspection of workers' submissions was conducted. Consequently, 19 workers are excluded in web tasks, and 33 workers are excluded in chatbot tasks.

### 4.1 RQ1: Standard Web versus Conversational Interfaces

**Execution Time**. Table 2 and Figure 4 depict basic statistics and the distribution of execution times for the considered experimental conditions. With the exception of the *Human OCR* task and the *Sentiment Analysis* task, the execution time distributions for the specific task types have no statistically significant difference (Mann-Whitney-Wilcoxon pair-wise significance test, region of rejection $p > 0.05$). *Speech Transcription* tasks show a slightly longer execution time, a result that we account to the UI design of the Web task, which, by forcing workers to open another browser tab to play the audio sample, might have caused delays.

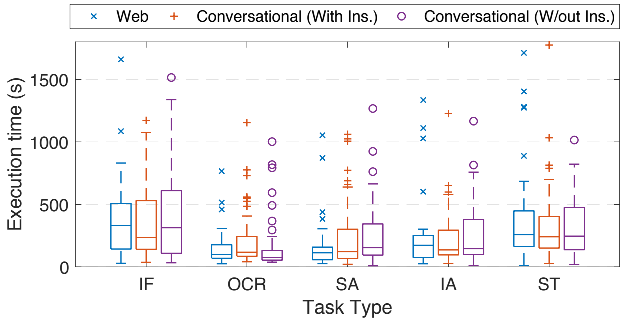


**Figure 4: Tasks execution time (sec): Web vs. Conversational with instructions vs. Conversational without instructions.**

**Table 2: Execution time ($\mu \pm \sigma$: average and standard deviation, unit: seconds) in each work interface. With Ins.: with instructions; W/out Ins.: without instructions.**

| *Task type* | Web | Conversational | |
|---|---|---|---|
| | | With Ins. | W/out Ins. |
| *Information Finding* | 364±301 | 362±295 | 393±328 |
| *Human OCR* | 150±135 | 219±227 | 160±209 |
| *Speech Transcription* | 384±381 | 333±306 | 311±223 |
| *Sentiment Analysis* | 158±187 | 243±276 | 244±247 |
| *Image Annotation* | 223±264 | 222±212 | 261±249 |

The statistically significant difference between the *Sentiment Analysis* tasks (web vs. chatbot without instructions, $p = 0.03$) and the *Human OCR* tasks (chatbot with instruction vs. chatbot without instructions, $p = 0.01$) could be explained by the presence of long textual instructions at the beginning of the conversational interface which, differently from the Web interface, could not be hidden. This hypothesis is supported by the results obtained with the chatbot configurations where instructions were not initially visible: for all task types, execution time are lower, and with no statistical difference from their Web counterpart. Interestingly, only within very few tasks (10) workers executed the chatbot command to fully display task instructions, but in 150 occasions they asked to instructions steps or instructions examples at the beginning of the task. Finally, it is worth mentioning that in 84 occasions workers used the task reviewing and editing functionality, to correct their answers before submitting the results.

[7] *Would you be interested in doing a similar task again in Telegram?*

**Work Quality**. Table 3 summarizes the work performance evaluation for the considered task types. We observe comparable performance across tasks, with precision that is slightly lower (on average) with Chatbot tasks. A manual analysis of the results highlights and interesting difference with the *Human OCR* tasks, where errors were mostly due to ambiguous characters in the CAPTCHA (e.g. "D" looking like either a capital "O" or a "0" (zero), rotated "L" looking like a "V"), but less present with chatbot workers. An analysis of the reasons beyond this result is left to future work.

**Table 3: Quality of crowdwork produced across different task and interface types.**

| *Task type* | Web | Conversational |
|---|---|---|
| *Information Finding* | 0.95 | 0.92 |
| *Human OCR* | 0.75 | 0.82 |
| *Speech transcription* | 0.85 | 0.75 |
| *Sentiment analysis* | 0.93 | 0.88 |
| *Image annotation* | 0.90 | 0.81 |

**Workers Satisfaction**. Workers participating in Chatbot tasks were also asked to provide feedback on their experience with microwork executed through conversational interfaces. 349 out of 600 executions received comment. Workers reported a positive opinion in 81.9% of comments. 44 workers gave a neutral comment. 19 workers indicated the issue about the slow response of the chatbot.

The majority of the comments highlighted the intuitive user experience (e.g. *"Very easy to understand , and easy and fastest now we have buttons"*, *"very pleasant experience, i like the replays from the BOT, very interactive! Thx!"*, *"i loved this task, is so much different to the others, and i think is a excellent work it with telegram. nice"*, *"It was different, but i like it.."*, *"Yeah, i like this type of Task, is cool, a new feature is coming to us"*). Others remarked the enjoyable experience (*"This is fun and easy task I may try another task like this! Great!"*, *"Its fun!! best experience for first time using telegram haha"*). Some workers reported issues with the *"complicated"* set up, or with instructions that could be improved (*"MEJORAR LAS INSTRUCCIONES"* – *"Improve the instructions"*).

Table 4 reports the average Overall (**OV**), Instruction (**IN**), Ease of Job (**EA**), and Pay (**PA**) ratings given by workers after finishing the tasks. These ratings, expressed in a range between 1 and 5, are requested by the Figure Eight platform, and are optionally provided by workers. Ratings for Standard Web interfaces are to be considered as references for the deployed task types and object instances. Conversational interfaces received on average high, although slightly lower ratings than the ones received by Web interfaces. The difference is evident especially with the *Information Finding* task, where workers reported significantly lower ratings for all considered dimensions. With *Sentiment Analysis* tasks, ratings highlight differences in instructions and ease of use. With *Human OCR*, *Image Annotation*, and *Speech Transcription* ratings are comparable.

**Table 4: Ratings of workers satisfaction. OV: Overall; IN: Instruction; EA: Ease of Job; PA: Pay**

| *Task type* | Platform | OV | IN | EA | PA |
|---|---|---|---|---|---|
| *Information Finding* | Web | 4.5 | 4.3 | 4.5 | 4.5 |
| | Conversational | 3.0 | 3.4 | 2.9 | 3.3 |
| *Human OCR* | Web | 4.3 | 4.2 | 4.0 | 4.5 |
| | Conversational | 3.4 | 4.0 | 3.8 | 4.3 |
| *Speech Transcription* | Web | 4.7 | 4.7 | 3.9 | 4.1 |
| | Conversational | 4.1 | 4.5 | 4.0 | 4.3 |
| *Sentiment Analysis* | Web | 4.3 | 4.3 | 4.1 | 4.1 |
| | Conversational | 3.7 | 3.4 | 3.2 | 3.8 |
| *Image Annotation* | Web | 3.8 | 3.8 | 3.3 | 3.8 |
| | Conversational | 3.7 | 3.9 | 3.1 | 3.9 |

## 4.2 RQ2: Conversational Interfaces — UI Elements

Figure 5 and Table 5 depict basic statistics and the distribution of execution times for the considered experimental conditions. The use of different custom keyboards have an impact on the task execution times, both for single- and multiple-selection tasks, with statistically significant differences (Mann-Whitney-Wilcoxon pair-wise significance test, $p < 0.05$) with the text configuration ($p = 0.0011$ for *Sentiment Analysis* and $p = 0.0036$ for *Image Annotation*) and the code configuration ($p = 0.0003$ for *Sentiment Analysis*).

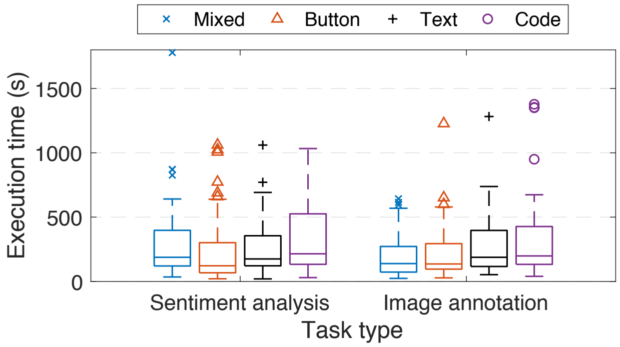


**Figure 5: Task execution time (in seconds) with different *custom keyboard* configurations.**

**Table 5: Execution time ($\mu \pm \sigma$: average and standard deviation, unit: seconds) in each chatbot Interface. The *Mixed* configuration is the one adopted in RQ1 experiments.**

| *Task type* | Mixed | Button | Text | Code |
|---|---|---|---|---|
| *Sentiment Analysis* | 301±306 | 243±276 | 325±257 | 267±219 |
| *Image Annotation* | 211±178 | 222±212 | 339±342 | 284±233 |

For the multiple-selection tasks, the availability of multiple input alternatives (*Mixed Custom Keyboard*) yields faster execution times; however, no clear total order of performance emerge across the two tasks. The removal of button shortcuts has a detrimental effect on workers execution time, while output quality is not affected. This is due to the input validation mechanism implemented in the chatbot, that prevents wrong results from being submitted.

## 4.3 Discussion and Implications

Results show that chatbots could be a suitable alternative to Web-based microwork platforms, at least for the considered task types, both in terms of execution time and quality. Although a direct comparison is not possible due to unavailable datasets and code, our results matches the outcome of previous studies with mobile UIs [6, 21]. Differently from [21], in our experiment the performance in *Human OCR* and *Image Labeling* tasks were of comparable quality. As also highlighted by previous work in mobile crowdsourcing [6, 19, 21], task and interaction design matter. Results suggest that for common tasks like *Sentiment Analysis* and *Image Labeling*, custom keyboard can enable execution times comparable to Web interfaces. Instructions and chatbot commands also have an impact, especially for domain specific tasks (e.g. food labeling).

Workers expressed positive opinions about this work interface modality. The analysis of workers' satisfaction highlight some differences across task types. While execution time and quality of output are comparable, workers were less satisfied with the quality of the instructions and ease of job (*Information Finding*, *Sentiment Analysis*) and with payment (*Information Finding*). This is an interesting outcome, that we hypothesise to be due to the novel work interface, and its relationship with the usual workflow of workers (e.g. in terms of keyboard usage, and cut&paste actions for information finding). This hypothesis will be tested in future work.

Overall, the obtained results are promising. Our takeaway from the whole experimental procedure and our results is that the flexibility (mixed-keyboard input and selection between Web and mobile client) for the interface to be used, the design of the interface, and the task itself are all important factors to consider when building crowdsourcing tasks for conversational interfaces. We believe that the experience with conversational crowd work interfaces could also play a role, but more experiments are needed to understand its relationship with execution time and quality.

We argue that the use of conversational interfaces for crowd work can provide a number of potential benefits, for instance: further democratization of crowd work, as people with limited digital skills or connectivity could then perform retributed digital work [26]; increased workers diversity (in terms of demographics, knowledge, and skills), thus providing better digital experimental environment, e.g. for psychological research [3]; increased workers capacity for low-latency and/or situational microtask crowdsourcing [13–15, 19]; and push microtask crowdsourcing [4, 25].

**Threats to Validity**. The recruited workers might not be representative of the whole population of crowd workers. While this risk is mitigated by the popularity of the F8 platform, experiments on other crowdsourcing and messaging platforms are needed for further generalization. To minimize the effect of user interface usability issues, we designed task interfaces that were either standard (Web tasks) or simplified (Chatbot). Not all workers were familiar with the Telegram messaging system, but we believe the presence of a web client (identical in functionality and look and feel to the native clients) to have minimized the risk of poor performance due to lack of experience with messaging systems. Issues of task complexity, clarity, and difficulty (tackled, for instance, in [6, 21]) will be addressed in future work. Finally, the experiment included a limited amount of task types and UI elements variations. While we acknowledge such limitation, we believe that our experimental design and results evaluation provide solid answer to the targeted research questions.

## 5 CONCLUSIONS AND FUTURE WORK

Text-based conversational agents are witnessing widespread adoption as effective tools to automate repetitive tasks, or as an alternative to traditional information seeking interfaces.

In this paper, we provide evidences of their suitability as microtask crowdsourcing platform (**RQ1**). Through a systematic analysis of five task types, we show that task execution times and output qualities are comparable to the ones achievable through Web based interfaces. The workers recruited in our experiments expressed positive opinions towards this work execution medium.

We highlighted the importance of task-specific interaction design, but also the convenience of advanced text input interfaces currently available in messaging platforms like Telegram (**RQ2**). The continuous evolution of the functionalities available in such platforms (e.g. novel content types, micropayment, etc.) could allow a broader, more democratic, and potentially decentralised adoption of crowd work (both for offer and demand).

This work provides plenty of inspirations for future research directions. Clearly, more research is needed to better understand the peak performance (speed and quality) achievable with different task and content types. Our work did not specifically study differences due to the devices used for work execution (desktop vs. mobile), both as a challenge (e.g. attention span, smaller keyboards, etc.) and as an opportunity for situational and location-based crowd sourcing. Further experiment could focus on push-based strategies initiated by the chatbot, as a method to perform and sustain near-real time crowdsourcing. Finally, we are interested in investigating the utility and performance conversational interfaces addressed to requester, both for task creation and monitoring.

## ACKNOWLEDGEMENTS

This research has been supported in part by the Amsterdam Institute for Advanced Metropolitan Solutions with the AMS Social Bot grant, by the Dutch national e-infrastructure with the support of SURF Cooperative (grant e-infra170237), and by the Erasmus+ project DISKOW (grant no. 60171990). We also thank all the anonymous crowdworkers from F8 who participated in our experiments.

## REFERENCES

[1] Sandeep Avula. 2017. Searchbots: Using Chatbots in Collaborative Information-seeking Tasks. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17).* ACM, New York, NY, USA, 1375–1375.

[2] Sandeep Avula, Gordon Chadwick, Jaime Arguello, and Robert Capra. 2018. SearchBots: User Engagement with ChatBots During Collaborative Search. In *Proceedings of the 2018 Conference on Human Information Interaction & Retrieval (CHIIR '18).* ACM, New York, NY, USA, 52–61.

[3] Tara S Behrend, David J Sharek, Adam W Meade, and Eric N Wiebe. 2011. The viability of crowdsourcing for survey research. *Behavior research methods* 43, 3 (2011), 800.

[4] Alessandro Bozzon, Marco Brambilla, Stefano Ceri, and Andrea Mauri. 2013. Reactive Crowdsourcing. In *Proceedings of the 22Nd International Conference on World Wide Web (WWW '13).* ACM, New York, NY, USA, 153–164.

[5] Justin Cranshaw, Emad Elwany, Todd Newman, Rafal Kocielnik, Bowen Yu, Sandeep Soni, Jaime Teevan, and Andrés Monroy-Hernández. 2017. Calendar. help: Designing a workflow-based scheduling agent with humans in the loop. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems.* ACM, 2382–2393.

[6] Vincenzo Della Mea, Eddy Maddalena, and Stefano Mizzaro. 2015. Mobile Crowdsourcing: Four Experiments on Platforms and Tasks. *Distrib. Parallel Databases* 33, 1 (March 2015), 123–141.

[7] M v Eeuwen. 2017. *Mobile conversational commerce: messenger chatbots as the next interface between businesses and consumers.* Master's thesis. TU Twente.

[8] Asbjørn Følstad, Petter Bae Brandtzaeg, Tom Feltwell, Effie L-C. Law, Manfred Tscheligi, and Ewa A. Luger. 2018. SIG: Chatbots for Social Good. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems (CHI EA '18).* ACM, New York, NY, USA, Article SIG06, 4 pages.

[9] Ujwal Gadiraju, Alessandro Checco, Neha Gupta, and Gianluca Demartini. 2017. Modus operandi of crowd workers: The invisible role of microtask work environments. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 49.

[10] Ujwal Gadiraju, Ricardo Kawase, and Stefan Dietze. 2014. A Taxonomy of Microtasks on the Web. In *Proceedings of the 25th ACM Conference on Hypertext and Social Media (HT '14)*. ACM, New York, NY, USA, 218–223.

[11] Ujwal Gadiraju, Jie Yang, and Alessandro Bozzon. 2017. Clarity is a worthwhile quality: On the role of task clarity in microtask crowdsourcing. In *Proceedings of the 28th ACM Conference on Hypertext and Social Media*. ACM, 5–14.

[12] Ting-Hao Kenneth Huang, Amos Azaria, and Jeffrey P Bigham. 2016. Instructable-crowd: Creating if-then rules via conversations with the crowd. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, 1555–1562.

[13] Ting-Hao (Kenneth) Huang, Joseph Chee Chang, and Jeffrey P. Bigham. 2018. Evorus: A Crowd-powered Conversational Assistant Built to Automate Itself Over Time. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, 295:1–295:13.

[14] Ting-Hao Kenneth Huang, Walter S Lasecki, Amos Azaria, and Jeffrey P Bigham. 2016. "Is There Anything Else I Can Help You With?" Challenges in Deploying an On-Demand Crowd-Powered Conversational Agent. In *Fourth AAAI Conference on Human Computation and Crowdsourcing*.

[15] Ting-Hao Kenneth Huang, Walter S Lasecki, and Jeffrey P Bigham. 2015. Guardian: A crowd-powered spoken dialog system for web apis. In *Third AAAI conference on human computation and crowdsourcing*.

[16] Shashank Khanna, Aishwarya Ratan, James Davis, and William Thies. 2010. Evaluating and improving the usability of Mechanical Turk for low-income workers in India. In *Proceedings of the first ACM symposium on computing for development*. ACM, 12.

[17] Aniket Kittur, Ed H Chi, and Bongwon Suh. 2008. Crowdsourcing user studies with Mechanical Turk. In *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 453–456.

[18] Aniket Kittur, Jeffrey V Nickerson, Michael Bernstein, Elizabeth Gerber, Aaron Shaw, John Zimmerman, Matt Lease, and John Horton. 2013. The future of crowd work. In *Proceedings of the 2013 conference on Computer supported cooperative work*. ACM, 1301–1318.

[19] Pavel Kucherbaev, Azad Abad, Stefano Tranquillini, Florian Daniel, Maurizio Marchese, and Fabio Casati. 2016. CrowdCafe-Mobile Crowdsourcing Platform. *arXiv preprint arXiv:1607.01752* (2016).

[20] P. Kucherbaev, A. Bozzon, and G. J. Houben. 2018. Human Aided Bots. *IEEE Internet Computing* (2018), 11. https://doi.org/10.1109/MIC.2018.252095348

[21] Abhishek Kumar, Kuldeep Yadav, Suhas Dev, Shailesh Vaya, and G. Michael Youngblood. 2014. Wallah: Design and Evaluation of a Task-centric Mobile-based Crowdsourcing Platform. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MOBIQUITOUS '14)*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium, 188–197.

[22] Xulei Liang, Rong Ding, Mengxiang Lin, Lei Li, Xingchi Li, and Song Lu. 2017. CI-Bot: A Hybrid Chatbot Enhanced by Crowdsourcing. In *Web and Big Data*, Shaoxu Song, Matthias Renz, and Yang-Sae Moon (Eds.). Springer International Publishing, Cham, 195–203.

[23] Rich Ling and Chih-Hui Lai. 2016. Microcoordination 2.0: Social coordination in the age of smartphones and messaging apps. *Journal of Communication* 66, 5 (2016), 834–856.

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

[25] Panagiotis Mavridis, David Gross-Amblard, and Zoltán Miklós. 2016. Using Hierarchical Skills for Optimized Task Assignment in Knowledge-Intensive Crowdsourcing. In *Proceedings of the 25th International Conference on World Wide Web (WWW '16)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 843–853.

[26] Prayag Narula, Philipp Gutheim, David Rolnitzky, Anand Kulkarni, and Bjoern Hartmann. 2011. MobileWorks: A Mobile Crowdsourcing Platform for Workers at the Bottom of the Pyramid. *Human Computation* 11, 11 (2011), 45.

[27] Navkar Samdaria, Ajith Sowndararajan, Ramadevi Vennelakanti, and Sriganesh Madhvanath. 2015. Mobile Interfaces for Crowdsourced Multimedia Microtasks. In *Proceedings of the 7th International Conference on HCI, IndiaHCI 2015 (IndiaHCI'15)*. ACM, New York, NY, USA, 62–67.

[28] Jessica Smith. 2018. THE MESSAGING APPS REPORT: How brands, businesses, and publishers can capitalize on the rising tide of messaging platforms. https://www.businessinsider.com/messaging-apps-report-2018-4. (2018).

[29] Kristen Vaccaro, Tanvi Agarwalla, Sunaya Shivakumar, and Ranjitha Kumar. 2018. Designing the Future of Personal Fashion. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 627, 11 pages.

[30] Alexandra Vtyurina, Denis Savenkov, Eugene Agichtein, and Charles L. A. Clarke. 2017. Exploring Conversational Search With Humans, Assistants, and Wizards. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '17)*. ACM, New York, NY, USA, 2187–2193.

[31] Tingxin Yan, Matt Marzilli, Ryan Holmes, Deepak Ganesan, and Mark Corner. 2009. mCrowd: A Platform for Mobile Crowdsourcing. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys '09)*. ACM, New York, NY, USA, 347–348.

[32] Victor W Zue and James R Glass. 2000. Conversational interfaces: Advances and challenges. *Proc. IEEE* 88, 8 (2000), 1166–1180.

[33] Darius Zumstein and Sophie Hundertmark. 2017. Chatbots–An Interactive Technology for personalized communication, transactions and services. *IADIS International Journal on WWW/Internet* 15, 1 (2017).