Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 288–297
Marseille, 11–16 May 2020

CRWIZ: A Framework for Crowdsourcing Real-Time Wizard-of-Oz Dialogues

Francisco J. Chiyah Garcia, José Lopes, Xingkun Liu, Helen Hastie
School of Mathematical and Computer Sciences,
Heriot-Watt University,
Edinburgh, United Kingdom,
{fjc3, jd.lopes, x.liu, h.hastie}@hw.ac.uk

Abstract

Large corpora of task-based and open-domain conversational dialogues are hugely valuable in the field of data-driven dialogue systems. Crowdsourcing platforms, such as Amazon Mechanical Turk, have been an effective method for collecting such large amounts of data. However, difficulties arise when task-based dialogues require expert domain knowledge or rapid access to domain-relevant information, such as databases for tourism. This will become even more prevalent as dialogue systems become increasingly ambitious, expanding into tasks with high levels of complexity that require collaboration and forward planning, such as in our domain of emergency response. In this paper, we propose CRWIZ: a framework for collecting real-time Wizard of Oz dialogues through crowdsourcing for collaborative, complex tasks. This framework uses semi-guided dialogue to avoid interactions that breach procedures and processes only known to experts, while enabling the capture of a wide variety of interactions. The framework is available at `https://github.com/JChiyah/crwiz`.

Keywords: Wizard-of-Oz, Data Collection, Crowdsourcing, Dialogue System

1. Introduction

Recent machine learning breakthroughs in dialogue systems and their respective components have been made possible by training on publicly available large scale datasets, such as ConvAI (Logacheva et al., 2018), bAbI (Weston et al., 2016) and MultiWoZ (Budzianowski et al., 2018), many of which are collected on crowdsourcing services, such as Amazon Mechanical Turk and Figure-eight. These data collection methods have the benefits of being cost-effective, time-efficient to collect and scalable, enabling the collection of large numbers of dialogues.

Where this crowdsourcing method has its limitations is when specific domain expert knowledge is required, rather than general conversation. These tasks include, for example, call centre agents (Peskov et al., 2019) or clerks with access to a database, as is required for tourism information and booking (Budzianowski et al., 2018). In the near future, there will be a demand to extend this to workplace-specific tasks and procedures. Therefore, a method of gathering crowdsourced dialogue data is needed that ensures compliance with such procedures, whilst providing coverage of a wide variety of dialogue phenomena that could be observed in deployment of a trained dialogue system.

Wizard-of-Oz data collections in the past have provided such a mechanism. However, these have traditionally not been scalable because of the scarcity of Wizard experts or the expense to train up workers. This was the situation with an initial study reported in (Lopes et al., 2019), which was conducted in a traditional lab setting and where the Wizard (an academic researcher) had to learn, through training and reading manuals, how best to perform operations in our domain of emergency response.

We present the CRWIZ Intelligent Wizard Interface that enables a crowdsourced Wizard to make intelligent, relevant choices without such intensive training by providing a restricted list of valid and relevant dialogue task actions, which changes dynamically based on the context, as the interaction evolves.

Prior crowdsourced wizarded data collections have divided the dialogue up into turns and each worker's job consists of one turn utterance generation given a static dialogue context, as in the MultiWoZ dataset (Budzianowski et al., 2018). However, this can limit naturalness of the dialogues by restricting forward planning, collaboration and use of memory that humans use for complex multi-stage tasks in a shared dynamic environment/context.

Our scenario is such a complex task. Specifically, our scenario relates to using robotics and autonomous systems on an offshore energy platform to resolve an emergency and is part of the EPSRC ORCA Hub project (Hastie et al., 2018). The ORCA Hub vision is to use teams of robots and autonomous intelligent systems to work on offshore energy platforms to enable cheaper, safer and more efficient working practices. An important part of this is ensuring safety of robots in complex, dynamic and cluttered environments, co-operating with remote operators. With this data collection method reported here, we aim to automate a conversational Intelligent Assistant (Fred), who acts as an intermediary between the operator and the multiple robotic systems (Chiyah Garcia et al., 2020; Lopes et al., 2020). Emergency response is clearly a high-stakes situation, which is difficult to emulate in a lab or crowdsourced data collection environment. Therefore, in order to foster engagement and collaboration, the scenario was gamified with a monetary reward given for task success.

In this paper, we provide a brief survey of existing datasets and describe the CRWIZ framework for pairing crowdworkers and having half of them acting as Wizards by limiting their dialogue options only to relevant and plausible ones, at any one point in the interaction. We then perform a data collection and compare our dataset to a similar dataset collected in a more controlled lab setting with a single Wizard (Lopes et al., 2019) and discuss the advantages/disadvantages of both approaches. Finally, we present future work. Our contributions are as follows:

- The release of a platform for the CRWIZ Intelligent Wizard Interface to allow for the collection of dialogue data for longer complex tasks, by providing a dynamic selection of relevant dialogue acts.
- A survey of existing datasets and data collection platforms, with a comparison to the CRWIZ data collection for Wizarded crowdsourced data in task-based interactions.

2. Related Work

Table 1 gives an overview of prior work and datasets. We report various factors to compare to the CRWIZ dataset corresponding to columns in Table 1: whether or not the person was aware they were talking to a bot; whether each dialogue had a single or multiple participants per role; whether the data collection was crowdsourced; and the modality of the interaction and the domain. As we see from the bottom row, none of the datasets reported in the table meet all the criteria we are aiming for, exemplifying the need for a new and novel approach.

Collecting large amounts of dialogue data can be very challenging as two interlocutors are required to create a conversation. If one of the partners in the conversation is a machine as in (Logacheva et al., 2018), the challenge becomes slightly easier since only one partner is lacking. However, in most cases these datasets are aimed at creating resources to train the conversational system itself. Self-authoring the dialogues (Krause et al., 2017) or artificially creating data (Weston et al., 2016) could be a solution to rapidly collect data, but this solution has been shown to produce low quality unnatural data (Jonell et al., 2019).

One way to mitigate the necessity of pairing two users simultaneously is to allow several participants to contribute to the dialogue, one turn at the time. This approach has been used both in task-oriented (Wen et al., 2017; Budzianowski et al., 2018; Eric and Manning, 2017) and chitchat (Jonell et al., 2019). This means that the same dialogue can be authored by several participants. However, this raises issues in terms of coherence and forward-planning. These can be addressed by carefully designing the data collection to provide the maximum amount of information to the participants (e.g. providing the task, personality traits of the bot, goals, etc.) but then this adds to cognitive load, time, cost and participant fatigue.

Pairing is a valid option, which has been used in a number of recent data collections in various domains, such as navigating in a city (de Vries et al., 2018), playing a negotiation game (Lewis et al., 2017), talking about a person (He et al., 2017), playing an image game (Manuvinakurike and DeVault, 2015) or having a chat about a particular image that is shown to both participants (Ilinykh et al., 2019; Das et al., 2017). Pairing frameworks exist such as `Slurk` (Schlangen et al., 2018). Besides its pairing management feature, `Slurk` is designed in order to allow researchers to modify it and implement their own data collection rapidly.

The scenarios for the above-mentioned data collections are mostly intuitive tasks that humans do quite regularly, unlike our use-case scenario of emergency response. Role playing is one option. For example, recent work has tried to create datasets for non-collaborative scenarios (Li et al., 2019; Wang et al., 2019), requesting participants to incarnate a particular role during the data collection. This is particularly challenging when the recruitment is done via a crowdsourcing platform. In (Wang et al., 2019), the motivation for the workers to play the role is intrinsic to the scenario. In this data collection, one of the participants tries to persuade their partner to contribute to a charity with a certain amount of money. As a result of their dialogue, the money that the persuadee committed to donate was actually donated to a charity organising. However, for scenarios such as ours, the role playing requires a certain expertise and it is questionable whether the desired behaviour would be achieved simply by letting two non-experts converse with free text.

Therefore, in recent data collections, there have been a number of attempts to control the data quality in order to produce a desired behaviour. For example, in (El Asri et al., 2017), the data collection was done with a limited number of subjects who performed the task several days in a row, behaving both as the Wizard and the customer of a travel agency. The same idea was followed in (Wei et al., 2018), where a number of participants took part in the data collection over a period of 6 months and, in (Peskov et al., 2019; Byrne et al., 2019) where a limited number of subjects were trained to be the Wizard. This quality control, however, naturally comes with the cost of recruiting and paying these subjects accordingly.

The solution we propose in this paper tries to minimise these costs by increasing the pool of Wizards to anyone wanting to collaborate in the data collection, by providing them the necessary guidance to generate the desired dialogue behaviour. This is a valuable solution for collecting dialogues in domains where specific expertise is required and the cost of training capable Wizards is high. We required fine-grained control over the Wizard interface so as to be able to generate more directed dialogues for specialised domains, such as emergency response for offshore facilities. By providing the Wizard with several dialogue options (aside from free text), we guided the conversation and could introduce actions that change an internal system state. This proposes several advantages:

1. A guided dialogue allows for set procedures to be learned and reduces the amount of data needed for a machine learning model for dialogue management to converge.
2. Providing several dialogue options to the Wizard increases the pace of the interaction and allows them to understand and navigate more complex scenarios.

3. System Overview

The CRWIZ Intelligent Wizard Interface resides on `Slurk` (Schlangen et al., 2018), an interaction server built for conducting dialogue experiments and data collections. `Slurk` handles the pairing of participants and provides a basic chat layout amongst other features. Refer to (Schlangen et al., 2018) for more information on the pairing of participants and the original chat layout. Our chat layout remains similar to `Slurk` with an important difference. In our scenario, we

Dataset	WoZ?	Single Participant?	Crowdsourced?	Interaction Modality	Domain
MultiWoZ (Budzianowski et al., 2018)	Partially†	No	Yes	Text	Tourism
RDG-Image Game (Manuvinakurike and DeVault, 2015)	No	Yes	Yes	Speech	Image game
MultiDiaGo (Peskov et al., 2019)	Controlled Wizards	N/A	User only	Text	Fast food, airline, finance, etc.
Stanford Multi-Domain Dialog Data (Eric and Manning, 2017)	Partially†	No	Yes	Text	Car assistant
Cambridge Restaurant (Wen et al., 2017)	Partially†	No	Yes	Text	Restaurants
MetalWoz (Lee et al., 2019)	N/A	Yes	Yes	Text	Multiple domains
AirDialogue (Wei et al., 2018)	N/A	N/A	Yes	Text	Flight booking
TalkTheWalk (de Vries et al., 2018)	No	No	Yes	Text/images	Navigation
Deal or No Deal (Lewis et al., 2017)	No	Yes	Yes	Text	Negotiation
Frames (El Asri et al., 2017)	Partially†	Yes	No	Text	Tourism
ConvAI (Logacheva et al., 2018)	No	Yes	Yes‡	Text	Context-based chat
bAbI Dialogues (Weston et al., 2016)	No	Artificial data	No	Text	Restaurants
EDINA (Krause et al., 2017)	No	Yes	Yes	Text	Chitchat
Fantom (Jonell et al., 2019)	No	No	Yes	Text and speech	Chitchat
MutualFriends (He et al., 2017)	No	Yes	Yes	Text	Context-based chat
Taskmaster-1 (Byrne et al., 2019)	Controlled Wizards	Yes	User only	Text and speech	Multiple domains
Collaborative Planning Corpus (Katsakioris et al., 2019)	Yes	Yes	No	Text and images	Mission planning
Our Data	**Yes**	**Yes**	**Yes**	**Text**	**Emergency response**[1]

Table 1: Comparison of relevant recent works. In order, the columns refer to: the dataset and reference; if the dataset was generated using Wizard-of-Oz techniques; if there was a unique participant per role for the whole dialogue; if the dataset was crowdsourced; the type of interaction modality used; and finally, the type of task or domain that the dataset covers. † The participants were aware that the dialogue was authored by humans. ‡ The participants were volunteers without getting paid.

assign each new participant a role (Operator or Wizard) and, depending on this role, the participant sees different game instructions and chat layout schemes. These are illustrated in Figures 1 and 2, for the Operator and Wizard respectively. The main components are described in turn below: 1) The Intelligent Wizard Interface; 2) dialogue structure; and 3) system-changing actions.

Wizard interface: the interface shown to participants with the Wizard role provides possible actions on the right-hand side of the browser window. These actions could be verbal, such as sending a message, or non-verbal, such as switching on/off a button to activate a robot. Figure 2 shows this interface with several actions available to be used in our data collection.

Dialogue structure: we introduced structured dialogues through a Finite State Machine (FSM) that controls the current dialogue state and offers multiple suitable and relevant state transitions (actions) to the Wizard depending on the point in the interaction, the state of the world and the history. A graph of dialogue states, transitions and utterances is loaded when the system is initialised, and each chat room has its own dialogue state, which changes through actions.

System-changing actions: actions trigger transitions between the states in the FSM. We differentiate two types of actions:

1. Verbal actions, such as the dialogue options available at that moment. The Wizard can select one of several predefined messages to send, or type their own message if needed. Free text messages do not change the dialogue state in the FSM, so it is important to minimise their use by providing enough dialogue options

[1] The CRWIZ framework is domain-agnostic, but the data collected with it corresponds to the emergency response domain.

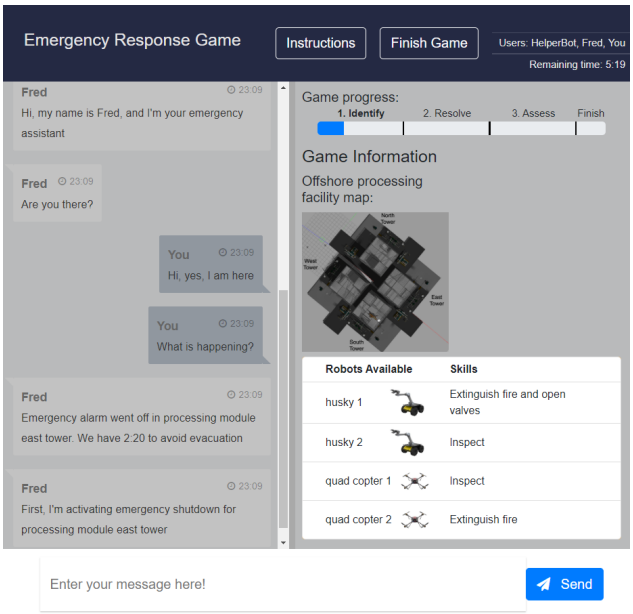

Figure 1: Interface shown to those in the Operator role running on the `Slurk` interaction server. It has a similar layout to other chat applications with the chat window on the left and a field to send messages at the bottom. The right side is used to display additional information.

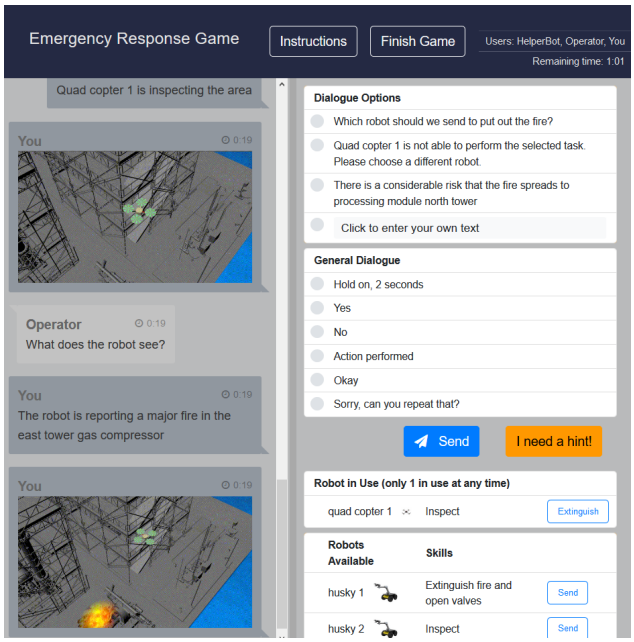

Figure 2: Interface shown to those in the Emergency Assistant Wizard role running on the `Slurk` interaction server. The chat window is on the left, with the dialogue options and buttons to control the robots on the right. The chat here shows GIFs that appear to increase engagement and show game progress visually.

to the Wizard. Predefined messages can also trigger other associated events such as pop-ups or follow-up non-verbal actions.

2. Non-verbal actions, such as commands to trigger events. These can take any form, but we used buttons to control robots in our data collection.

Submitting an action would change the dialogue state in the FSM, altering the set of actions available in the subsequent turn visible to the Wizard. Some dialogue options are only possible at certain states, in a similar way as to how non-verbal actions are enabled or disabled depending on the state. This is reflected in the Wizard interface.

The advantage of the CRWIZ framework is that it can easily be adapted to different domains and procedures by simply modifying the dialogue states loaded at initialisation. These files are in YAML format and have a simple structure that defines their NLG templates (the FSM will pick one template at random if there is more than one) and the states that it can transition to. Note, that some further modifications may be necessary if the scenario is a slot-filling dialogue requiring specific information at various stages.

Once the dialogue between the participants finishes, they receive a code in the chat, which can then be submitted to the crowdsourcing platform for payment. The CRWIZ framework generates a JSON file in its log folder with all the information regarding the dialogue, including messages sent, FSM transitions, world state at each action, etc. Automatic evaluation metrics and annotations are also appended such as number of turns per participant, time taken or if one of the participants disconnected. Paying the crowdworkers can be done by just checking that there is a dialogue file with the token that they entered.

4. Data Collection

We set up a crowdsourced data collection through Amazon Mechanical Turk, in which two participants chatted with each other in a setting involving an emergency at an offshore facility. As mentioned above, participants had different roles during the interaction: one of them was an Operator of the offshore facility whereas the other one acted as an Intelligent Emergency Assistant. Both of them had the same goal of resolving the emergency and avoiding evacuation at all costs, but they had different functions in the task:

- The **Operator** was responsible for the facility and had to give instructions to the Emergency Assistant to perform certain actions, such as deploying emergency robots. Participants in the role of Operator were able to chat freely with no restrictions and were additionally given a map of the facility and a list of available robots (see Figure 1).

- The **Emergency Assistant** had to help the Operator handle the emergency by providing guidance and executing actions. Participants in the role of Emergency Assistant had predefined messages depending on the task progress. They had to choose between one of the options available, depending on which made sense at the time, but they also had the option to write their own message if necessary. The Emergency Assistant role mimics that of the Wizard in a Wizard-of-Oz experiment (see Figure 2).

The participants had a limited time of 6 minutes to resolve the emergency, which consisted of the following subtasks: 1) identify and locate the emergency; 2) resolve the

emergency; and 3) assess the damage caused. They had four robots available to use with different capabilities: two ground robots with wheels (Husky) and two Quadcopter UAVs (Unmanned Aerial Vehicles). For images of these robots, see Figure 1. Some robots could inspect areas whereas others were capable of activating hoses, sprinklers or opening valves. Both participants, regardless of their role, had a list with the robots available and their capabilities, but only the Emergency Assistant could control them. This control was through high-level actions (e.g. moving a robot to an area, or ordering the robot to inspect it) that the Emergency Assistant had available as buttons in their interface, as shown in Figure 2. For safety reasons that might occur in the real world, only one robot could be active doing an action at any time. The combinations of robots and capabilities meant that there was not a robot that could do all three steps of the task mentioned earlier (inspect, resolve and assess damage), but the robots could be used in any order allowing for a variety of ways to resolve the emergency.

Participants would progress through the task when certain events were triggered by the Emergency Assistant. For instance, inspecting the area affected by an alarm would trigger the detection of the emergency. After locating the emergency, other dialogue options and commands would open up for the Emergency Assistant. In order to give importance to the milestones in the dialogue, these events were also signalled by GIFs (short animated video snippets) in the chat that both participants could see (e.g. a robot finding a fire), as in Figure 3. The GIFs were added for several reasons: to increase participant engagement and situation awareness, to aid in the game and to show progress visually. Note that there was no visual stimuli in the original WoZ study (Lopes et al., 2019) but they were deemed necessary here to help the remote participants contextualise the scenario. These GIFs were produced using a Digital Twin simulation of the offshore facility with the various types of robots. See (Pairet et al., 2019) for details on the Digital Twin.

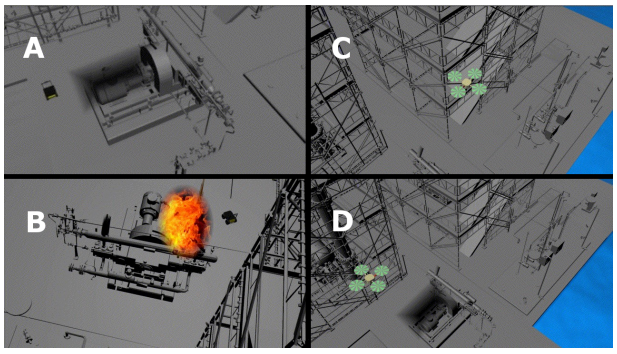

Figure 3: Some of the GIFs shown during the game. A and B are Husky robots assessing damages and inspecting a fire respectively. C and D show Quadcopter UAVs moving and inspecting an area.

4.1. Implementation

The dialogue structure for the Emergency Assistant (the Wizard) followed a dialogue flow previously used for the original lab-based Wizard-of-Oz study (Lopes et al., 2019) but which was slightly modified and simplified for this crowdsourced data collection. In addition to the transitions that the FSM provides, there are other fixed dialogue options always available such as *"Hold on, 2 seconds"*, *"Okay"* or *"Sorry, can you repeat that?"* as a shortcut for commonly used dialogue acts, as well as the option to type a message freely.

The dialogue has several paths to reach the same states with varying levels of Operator control or engagement that enriched the heterogeneity of conversations. The Emergency Assistant dialogue options show various speaking styles, with a more assertive tone (*"I am sending Husky 1 to east tower"*) or others with more collaborative connotations (*"Which robot do you want to send?"* or *"Husky 1 is available to send to east tower"*). Refer to (Lopes et al., 2019) for more details. Furthermore, neither participants were restricted in the number of messages that they could send and we did not require a balanced number of turns between them. However, there were several dialogue transitions that required an answer or authorisation from the Operator, so the FSM would lock the dialogue state until the condition was met. As mentioned earlier, the commands to control the robots are also transitions of the FSM, so they were not always available.

The Emergency Assistant interface contains a button to get a hint if they get stuck at any point of the conversation. This hint mechanism, when activated, highlights one of the possible dialogue options or robot buttons. This highlighted transition was based on the observed probability distribution of transitions from (Lopes et al., 2019) to encourage more collaborative interaction than a single straight answer.

As in the real world, robot actions during the task were simulated to take a certain period of time, depending on the robot executing it and the action. The Emergency Assistant had the option to give status updates and progress reports during this period. Several dialogue options were available for the Emergency Assistant whilst waiting. The time that robots would take to perform actions was based on simulations run on a Digital Twin of the offshore facility implemented in Gazebo (Pairet et al., 2019). Specifically, we pre-simulated typical robot actions, with the robot's progress and position reflected in the Wizard interface with up-to-date dialogue options for the Emergency Assistant. Once the robot signals the end of their action, additional updated dialogue options and actions are available for the Emergency Assistant. This simulation allowed us to collect dialogues with a realistic embedded world state.

4.2. Deployment

We used Amazon Mechanical Turk (AMT) for the data collection. We framed the task as a game to encourage engagement and interaction. The whole task, (a Human Intelligence Task (HIT) in AMT) consisted of the following:

1. Reading an initial brief set of instructions for the overall task.
2. Waiting for a partner for a few seconds before being able to start the dialogue.
3. When a partner was found, they were shown the instructions for their assigned role. As these were differ-

ent, we ensured that they both took around the same time. The instructions had both a text component and a video explaining how to play, select dialogues, robots, etc[2].

4. Playing the game to resolve the emergency. This part was limited to 6 minutes.

5. Filling a post-task questionnaire about partner collaboration and task ease.

The participants received a game token after finishing the game that would allow them to complete the questionnaire and submit the task. This token helped us link their dialogue to the responses from the questionnaire.
Several initial pilots helped to define the total time required as 10 minutes for all the steps above. We set the HIT in AMT to last 20 minutes to allow additional time should any issues arise. The pilots also helped setting the payment for the workers. Initially, participants were paid a flat amount of $1.4 per dialogue. However, we found that offering a tiered payment tied to the length of the dialogue and bonus for completing the task was the most successful and cost-effective method to foster engagement and conversation:

- $0.5 as base for attempting the HIT, reading the instructions and completing the questionnaire.
- $0.15 per minute during the game, for a maximum of $0.9 for the 6 minutes.
- $0.2 additional bonus if the participants were able to successfully avoid the evacuation of the offshore facility.

The pay per worker was therefore $1.4 for completing a whole dialogue and $1.6 for those who resolved the emergency for a 10-minute HIT. This pay is above the Federal minimum wage in the US ($7.25/hr or $0.12/min) at the time of the experiment.
The post-task questionnaire had four questions rated in 7-point rating scales that are loosely based on the PARADISE (Walker et al., 1997) questions for spoken dialogue systems:

Q1. **Partner collaboration:** *"How helpful was your partner?"* on a scale of 1 (not helpful at all) to 7 (very helpful).

Q2. **Information ease:** *"In this conversation, was it easy to get the information that I needed?"* on a scale of 1 (no, not at all) to 7 (yes, completely).

Q3. **Task ease:** *"How easy was the task?"* on a scale of 1 (very easy) to 7 (very difficult).

Q4. **User expertise:** *"In this conversation, did you know what you could say or do at each point of the dialog?"* on a scale of 1 (no, not at all) to 7 (yes, completely).

At the end, there was also an optional entry to give free text feedback about the task and/or their partner.

[2] Video with instructions for the emergency assistant is available at http://bit.ly/32Rjg8N

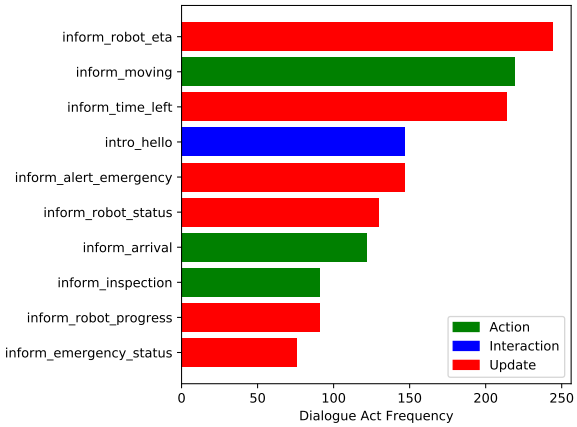

Figure 4: Frequency of the top-10 Emergency Assistant dialogue acts in the data collected. There were 40 unique dialogue acts, each with two or more distinct formulations on average. Most of them also had slots to fill with contextual information, such as the name of the robot. Dialogue acts are colour-coded based on 3 main types.

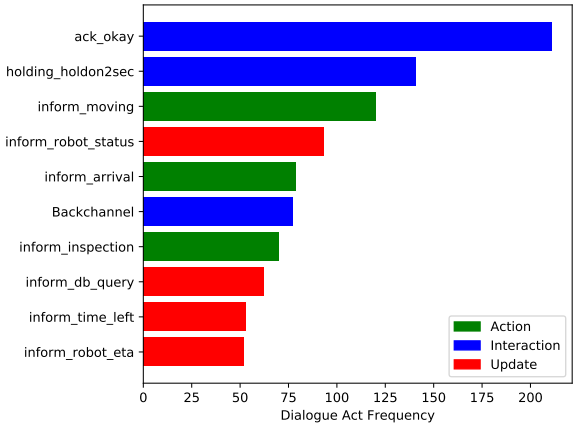

Figure 5: Frequency of the top-10 Emergency Assistant dialogue acts in (Lopes et al., 2019).

5. Data Analysis

For the intitial data collection using the CRWIZ platform, 145 unique dialogues were collected (each dialogue consists of a conversation between two participants). All the dialogues were manually checked by one of the authors and those where the workers were clearly not partaking in the task or collaborating were removed from the dataset. The average time per assignment was 10 minutes 47 seconds, very close to our initial estimate of 10 minutes, and the task was available for 5 days in AMT. Out of the 145 dialogues, 14 (9.66%) obtained the bonus of $0.2 for resolving the emergency. We predicted that only a small portion of the participants would be able to resolve the emergency in less than 6 minutes, thus it was framed as a bonus challenge rather than a requirement to get paid[3]. The fastest time recorded to resolve the emergency was 4 minutes 13 seconds with a mean of 5 minutes 8 seconds. Table 2 shows several interaction statistics for the data collected compared to the single lab-based WoZ study (Lopes et al., 2019).

[3] Dialogues where the emergency was not resolved are still valid.

Feature	Dialogues Collected Mean (SD)	Lopes et al. (2019) Mean (SD)
Number of Turns	25.22 (9.69)	53.26 (9.13)
Number of Operator Turns	7.99 (3.96)	9.78 (7.67)
Number of Emergency Assistant Turns	17.23 (7.97)	43.64 (4.45)
Operator Turn Length (words)	3.88 (1.69)	3.02 (1.59)
Emergency Assistant % typed Utterances	2.29% (5.16%)	1.72% (3.34 %)

Table 2: Interaction features of the dialogues collected. We compare it with the results of the Wizard-of-Oz experiment in a controlled setting from (Lopes et al., 2019).

Type of DA	Dialogues Collected	(Lopes et al., 2019)
% Request	7.14	6.85
% Interaction	20.31	29.20
% Action	20.19	21.40
% Update	52.36	42.54

Table 3: Distribution of the types of dialogue acts in the data collected with CRWIZ, compared with (Lopes et al., 2019).

Subjective Data Table 4 gives the results from the post-task survey. We observe, that subjective and objective task success are similar in that the dialogues that resolved the emergency were rated consistently higher than the rest.
Mann-Whitney-U one-tailed tests show that the scores of the Emergency Resolved Dialogues for Q1 and Q2 were significantly higher than the scores of the Emergency Not Resolved Dialogues at the 95% confidence level (Q1: $U = 1654.5$, $p < 0.0001$; Q2: $U = 2195$, $p = 0.009$, both $p < 0.05$). This indicates that effective collaboration and information ease are key to task completion in this setting.
Regarding the qualitative data, one of the objectives of the Wizard-of-Oz technique was to make the participant believe that they are interacting with an automated agent and the qualitative feedback seemed to reflect this: *"The AI in the game was not helpful at all [...]"* or *"I was talking to Fred a bot assistant, I had no other partner in the game"*.

Single vs Multiple Wizards In Table 2, we compare various metrics from the dialogues collected with crowdsourcing with the dialogues previously collected in a lab environment for a similar task. Most figures are comparable, except the number of emergency assistant turns (and consequently the total number of turns). To further understand these differences, we have first grouped the dialogue acts in four different broader types: Updates, Actions, Interactions and Requests, and computed the relative frequency of each of these types in both data collections. In addition, Figures 4 and 5 show the distribution of the most frequent dialogue acts in the different settings. It is visible that in the lab setting where the interaction was face-to-face with a robot, the Wizard used more Interaction dialogue acts (Table 3). These were often used in context where the Wizard needed to hold the turn while looking for the appropriate prompt or waiting for the robot to arrive at the specified goal in the environment. On the other hand, in the crowdsourced data collection utterances, the situation updates were a more common choice while the assistant was waiting for the robot to travel to the specified goal in the environment.
Perhaps not surprisingly, the data shows a medium strong positive correlation between task success and the number of Action type dialogue acts the Wizard performs, triggering events in the world leading to success ($R = 0.475$). There is also a positive correlation between task success and the number of Request dialogue acts requesting confirmation before actions ($R = 0.421$), e.g., *"Which robot do you want to send?"*. As Table 3 shows, these are relatively rare but perhaps reflect a level of collaboration needed to further the task to completion. Table 5 shows one of the dialogues collected where the Emergency Assistant continuously engaged with the Operator through these types of dialogue acts.
The task success rate was also very different between the two set-ups. In experiments reported in (Lopes et al., 2019), 96% of the dialogues led to the extinction of the fire whereas in the crowdsourcing setting only 9.66% achieved the same goal. In the crowdsourced setting, the robots were slower moving at realistic speeds unlike the lab setting[4]. A higher bonus and more time for the task might lead to a higher task success rate.

Limitations It is important to consider the number of available participants ready and willing to perform the task at any one time. This type of crowdsourcing requires two participants to connect within a few minutes of each other to be partnered together. As mentioned above, there were some issues with participants not collaborating and these dialogues had to be discarded as they were not of use[5].

5.1. Future Work

In future work, we want to expand and improve the platform. Dialogue system development can greatly benefit from better ways of obtaining data for rich task-oriented domains such as ours. Part of fully exploiting the potential of crowdsourcing services lies in having readily available tools that help in the generation and gathering of data. One such tool would be a method to take a set of rules, procedures or business processes and automatically convert to a FSM, in a similar way to (Lemon et al., 2008), ready to be uploaded to the Wizard interface.
Regarding quality and coherence, dialogues are particularly challenging to automatically rate. In our data collection, there was not a correct or wrong dialogue option for the messages that the Emergency Assistant sent during the conversation, but some were better than others depending on the

[4] There was no live connection with the simulated physical environment implemented.

[5] Participants who collaborated still received the full payment regardless of their partner's behaviour.

	Dialogues Collected (145) Mean/Median/Mode (SD)	**Emergency Not Resolved Dialogues (131)** Mean/Median/Mode (SD)	**Emergency Resolved Dialogues (14)** Mean/Median/Mode (SD)
Q1. Partner collaboration	3.76/4/1 (2.0)	3.59/4/1 (1.99)	**5.19/5/5** (1.52)*
Q2. Information ease	3.65/4/1 (2.0)	3.55/3/1 (2.01)	**4.48/5/5** (1.72)*
Q3. Task ease	3.08/3/2 (1.73)	3.03/3/2 (1.74)	**3.56/3/3** (1.67)
Q4. User expertise	4.09/4/4 (1.81)	4.03/4/4 (1.82)	**4.59/5/6** (1.67)

Table 4: Subjective ratings for the post-task survey reporting Mean, Median, Mode and Standard Deviation (SD). Scales were on a 7-point rating scale. "Dialogues Collected" refers to all the dialogues collected after filtering, whereas the other columns are for the dialogues that did not resolved the emergency ("Emergency Not Resolved Dialogues") and those that did ("Emergency Resolved Dialogues"). Higher is better (Q3 reversed for this table). Highest numbers are **bold**. * indicates significant differences ($p < 0.05$, Mann-Whitney-U) between Emergency Resolved and Emergency Not Resolved dialogues.

context with the Operator. This context is not easily measurable for complex tasks that depend on a dynamic world state. Therefore, we leave to future work automatically measuring dialogue quality through the use of context.

The introduction of Instructional Manipulation Checks (Oppenheimer et al., 2009) before the game to filter out inattentive participants could improve the quality of the data (Crowdworkers are known for performing multiple tasks at once). Goodman et al. (2013) also recommend including screening questions that check both attention and language comprehension for AMT participants. Here, there is a balance that needs to be investigated between experience and quality of crowdworkers and the need for large numbers of participants in order to be quickly paired.

We are currently exploring using the data collected to train dialogue models for the emergency response domain using Hybrid Code Networks (Williams et al., 2017).

6. Conclusion

In conclusion, this paper described a new, freely available tool to collect crowdsourced dialogues in rich task-oriented settings. By exploiting the advantages of both the Wizard-of-Oz technique and crowdsourcing services, we can effortlessly obtain dialogues for complex scenarios. The predefined dialogue options available to the Wizard intuitively guide the conversation and allow the domain to be deeply explored without the need for expert training. These predefined options also reinforce the feeling of a true Wizard-of-Oz experiment, where the participant who is not the Wizard thinks that they are interacting with a non-human agent.

As the applications for task-based dialogue systems keep growing, we will see the need for systematic ways of generating dialogue corpora in varied, richer scenarios. This platform aims to be the first step towards the simplification of crowdsourcing data collections for task-oriented collaborative dialogues where the participants are working towards a shared common goal. The code for the platform and the data are also released with this publication.

7. Acknowledgements

This work was supported by the EPSRC funded ORCA Hub (EP/R026173/1, 2017-2021). Chiyah Garcia's PhD is funded under the EPSRC iCase EP/T517471/1 with Siemens.

8. Bibliographical References

Budzianowski, P., Wen, T.-H., Tseng, B.-H., Casanueva, I., Ultes, S., Ramadan, O., and Gašić, M. (2018). MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 5016–5026, Brussels, Belgium, October-November. Association for Computational Linguistics.

Byrne, B., Krishnamoorthi, K., Sankar, C., Neelakantan, A., Goodrich, B., Duckworth, D., Yavuz, S., Dubey, A., Kim, K.-Y., and Cedilnik, A. (2019). Taskmaster-1: Toward a Realistic and Diverse Dialog Dataset. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 4515–4524, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chiyah Garcia, F. J., Lopes, J., and Hastie, H. (2020). Natural language interaction to facilitate mental models of remote robots. In Proceedings of the Workshop on Mental Models of Robots, HRI'20, HRI'20, Cambridge, UK, 3. ACM.

Das, A., Kottur, S., Gupta, K., Singh, A., Yadav, D., Moura, J. M., Parikh, D., and Batra, D. (2017). Visual dialog. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 326–335.

de Vries, H., Shuster, K., Batra, D., Parikh, D., Weston, J., and Kiela, D. (2018). Talk the walk: Navigating new york city through grounded dialogue. *arXiv preprint arXiv:1807.03367*.

El Asri, L., Schulz, H., Sharma, S., Zumer, J., Harris, J., Fine, E., Mehrotra, R., and Suleman, K. (2017). Frames: a corpus for adding memory to goal-oriented dialogue systems. In Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue, pages 207–219, Saarbrücken, Germany, August. Association for Computational Linguistics.

Eric, M. and Manning, C. D. (2017). Key-value retrieval networks for task-oriented dialogue. In Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue, pages 37–49, Saarbrücken, Germany, August. Association for Computational Linguistics.

Goodman, J. K., Cryder, C. E., and Cheema, A. (2013). Data Collection in a Flat World: The Strengths and Weak-

Participant	Utterance	Dialogue Acts
Assistant:	*"Hi, my name is Fred, and I'm your emergency assistant"*	intro_hello
	"Are you there?"	request_attention
Operator:	*"Hi Fred, I am _"*	
Assistant:	*"Emergency alarm went off in processing module east tower. We have 4:56 to avoid evacuation"*	inform_alert_emergency
Operator:	*"Ok what do you suggest we do first"*	
Assistant:	*"Do you want to make a PA announcement to evacuate processing module east tower?"*	request_pa_announcement
Operator:	*"Yes that sounds good"*	
Assistant:	*"Action performed"*	action_performed
	"First, I'm activating emergency shutdown for processing module east tower"	inform_activate_emergency_shutdown
Operator:	*"Ok"*	
Assistant:	*"What type of robot do you want to use to perform the inspection?"*	request_robot_type
Operator:	*"I would like to use the quad copter 1"*	
Assistant:	*"Moving quad copter 1 to processing module east tower"*	inform_moving
	"The estimated time of arrival is 9 seconds" inform_robot_eta	
	"There is still 3:08 before evacuation"	inform_time_left
	"Quad copter 1 has arrived to processing module east tower"	inform_arrival
Operator:	*"Is Quad copter indicating what the problem is?"*	
Assistant:	*"Quad copter 1 is going to inspect processing module east tower"*	inform_inspection
	"The robot is reporting a major fire in the east tower gas compressor"	inform_emergency_status
	"Which robot should we send to put out the fire?"	request_robot_emergency
Operator:	*"Should we extinguish the fire now using quad copter 2"*	
Assistant:	*"Sending quad copter 2 to processing module east tower"*	inform_moving
	...	

Table 5: Interaction between participants from one of the dialogues collected.

nesses of Mechanical Turk Samples. *Journal of Behavioral Decision Making*, 26(3):213–224, jul.

Hastie, H., Lohan, K., Chantler, M., Robb, D. A., Ramamoorthy, S., Petrick, R., Vijayakumar, S., and Lane, D. (2018). The ORCA Hub: Explainable offshore robotics through intelligent interfaces. In Proc. of Explainable Robotic Systems Workshop, ACM HRI Conference, pages 1–2, 3.

He, H., Balakrishnan, A., Eric, M., and Liang, P. (2017). Learning symmetric collaborative dialogue agents with dynamic knowledge graph embeddings. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1766–1776, Vancouver, Canada, July. Association for Computational Linguistics.

Ilinykh, N., Zarrieß, S., and Schlangen, D. (2019). MeetUp! A Corpus of Joint Activity Dialogues in a Visual Environment. In Proceedings of the 23rd Workshop on the Semantics and Pragmatics of Dialogue (SemDial 2019 / LondonLogue), London, UK, September.

Jonell, P., Fallgren, P., Doğan, F. I., Lopes, J., Wennberg, U., and Skantze, G. (2019). Crowdsourcing a self-evolving dialog graph. In Proceedings of the 1st International Conference on Conversational User Interfaces, page 14. ACM.

Katsakioris, M. M., Hastie, H., Konstas, I., and Laskov, A. (2019). Corpus of multimodal interaction for collaborative planning. In Proceedings of the Combined Workshop on Spatial Language Understanding (SpLU) and Grounded Communication for Robotics (RoboNLP), pages 1–6, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Krause, B., Damonte, M., Dobre, M., Duma, D., Fainberg, J., Fancellu, F., Kahembwe, E., Cheng, J., and Webber, B. L. (2017). Edina: Building an open domain socialbot with self-dialogues. *CoRR*, abs/1709.09816.

Lee, S., Schulz, H., Atkinson, A., Gao, J., Suleman, K., El Asri, L., Adada, M., Huang, M., Sharma, S., Tay, W., and Li, X. (2019). Multi-domain task-completion dialog challenge. In Dialog System Technology Challenges 8, March.

Lemon, O., Liu, X., and Hastie, H. (2008). Build your own spoken dialogue systems: automatically generating isu dialogue systems from business user resources. In Proceedings of the 22nd International Conference on on Computational Linguistics (COLING): Demonstration Papers, pages 161–164. Association for Computational Linguistics.

Lewis, M., Yarats, D., Dauphin, Y., Parikh, D., and Batra, D. (2017). Deal or no deal? end-to-end learning of negotiation dialogues. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing,

pages 2443–2453, Copenhagen, Denmark, September. Association for Computational Linguistics.

Li, Y., Qian, K., Shi, W., and Yu, Z. (2019). End-to-end trainable non-collaborative dialog system.

Logacheva, V., Burtsev, M., Malykh, V., Polulyakh, V., and Seliverstov, A. (2018). ConvAI Dataset of Topic-Oriented Human-to-Chatbot Dialogues. In The NIPS '17 Competition: Building Intelligent Systems, pages 47–57. Springer, Cham.

Lopes, J., Robb, D. A., Ahmad, M., Liu, X., Lohan, K., and Hastie, H. (2019). Towards a Conversational Agent for Remote Robot-Human Teaming. In ACM/IEEE International Conference on Human-Robot Interaction, volume 2019-March, pages 548–549. IEEE, March.

Lopes, J. D., Robb, D., Liu, X., and Hastie, H. (2020). Demonstration of a social robot for control of remote autonomous systems. In 2020 15th ACM/IEEE International Conference on Human-Robot Interaction (HRI), United States, 12. IEEE.

Manuvinakurike, R. and DeVault, D. (2015). Pair Me Up: A Web Framework for Crowd-Sourced Spoken Dialogue Collection. In Proceedings of IWSDS 2015, pages 1 –12, Busan, South Korea.

Oppenheimer, D. M., Meyvis, T., and Davidenko, N. (2009). Instructional manipulation checks: Detecting satisficing to increase statistical power. *Journal of Experimental Social Psychology*, 45(4):867–872, jul.

Pairet, Èric., Ardón, P., Liu, X., Hastie, H., and Lohan, K. S. (2019). A digital twin for human-robot interaction. In Proceedings of ACM/IEEE Intl. Conf. on Human-Robot Interaction, New York, NY, USA.

Peskov, D., Clarke, N., Krone, J., Fodor, B., Zhang, Y., Youssef, A., and Diab, M. (2019). Multi-domain goal-oriented dialogues (MultiDoGO): Strategies toward curating and annotating large scale dialogue data. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 4518–4528.

Schlangen, D., Diekmann, T., Ilinykh, N., and Zarrieß, S. (2018). slurk – A lightweight interaction server for dialogue experiments and data collection. In Proceedings of the 22nd Workshop on the Semantics and Pragmatics of Dialogue (AixDial/semdial 2018).

Walker, M. A., Litman, D. J., Kamm, C. A., and Abella, A. (1997). PARADISE. In Proceedings of the 35th annual meeting on Association for Computational Linguistics -, pages 271–280, Morristown, NJ, USA. Association for Computational Linguistics.

Wang, X., Shi, W., Kim, R., Oh, Y., Yang, S., Zhang, J., and Yu, Z. (2019). Persuasion for good: Towards a personalized persuasive dialogue system for social good. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 5635–5649, Florence, Italy, July. Association for Computational Linguistics.

Wei, W., Le, Q., Dai, A., and Li, J. (2018). AirDialogue: An environment for goal-oriented dialogue research. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 3844–3854, Brussels, Belgium, October-November. Association for Computational Linguistics.

Wen, T.-H., Vandyke, D., Mrkšić, N., Gašić, M., Rojas-Barahona, L. M., Su, P.-H., Ultes, S., and Young, S. (2017). A network-based end-to-end trainable task-oriented dialogue system. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pages 438–449, Valencia, Spain, April. Association for Computational Linguistics.

Weston, J., Bordes, A., Chopra, S., and Mikolov, T. (2016). Towards AI-complete question answering: A set of prerequisite toy tasks. In Proceedings of 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings.

Williams, J. D., Asadi, K., and Zweig, G. (2017). Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 665–677, Vancouver, Canada, July. Association for Computational Linguistics.