

Interactive Grounded Language Understanding in a Collaborative Environment: IGLU 2021

Julia Kiseleva¹ Ziming Li³ Mohammad Aliannejadi² Shrestha Mohanty¹ Maartje ter Hoeve²
Mikhail Burtsev^{4,5} Alexey Skrynnik⁵ Artem Zholus⁴ Aleksandr Panov⁴ Kavya Srinet⁶ Arthur
Szlam⁶ Yuxuan Sun⁶ Marc-Alexandre Côté¹ Katja Hofmann¹ Ahmed Awadallah¹

Linar Abdrazakov⁷ Igor Churin⁸ Putra Manggala² Kata Naszadi² Michiel van der Meer¹⁰ Taewoon
Kim¹¹

julia.kiseleva@microsoft.com

¹Microsoft Research ²University of Amsterdam ³Alexa AI ⁴MIPT ⁵AIRI ⁶Facebook AI ⁷Moscow Institute
of Physics and Technology ⁸Kalashnikov Izhevsk State Technical University ¹⁰Universiteit Leiden ¹¹Vrije
Universiteit Amsterdam

Editors: Douwe Kiela, Marco Ciccone, Barbara Caputo

Abstract

Human intelligence has the remarkable ability to quickly adapt to new tasks and environments. Starting from a very young age, humans acquire new skills and learn how to solve new tasks either by imitating the behavior of others or by following provided natural language instructions. To facilitate research in this direction, we propose *IGLU: Interactive Grounded Language Understanding in a Collaborative Environment*. The primary goal of the competition is to approach the problem of how to build interactive agents that learn to solve a task while provided with grounded natural language instructions in a collaborative environment. Understanding the complexity of the challenge, we split it into sub-tasks to make it feasible for participants.

Keywords

Natural Language Understanding (NLU), Reinforcement Learning (RL), Grounded Learning, Interactive Learning, Games

1. Introduction

Humans have the remarkable ability to quickly adapt to new tasks and environments. Starting from a very young age, humans acquire new skills and learn how to solve new tasks either by imitating behavior of others or by following natural language instructions that are provided to them (An, 1988; Council, 1999). Natural language communication provides a natural way for humans to acquire new knowledge, enabling us to learn quickly through language instructions and other forms of interaction such as visual cues. This form of learning can even accelerate the acquisition of new skills by avoiding trial-and-error and statistical generalization when learning only from observations (Thomaz et al., 2019). Studies in developmental psychology have shown evidence of human communication being an effective method for transmission of generic knowledge between individuals as young as infants (Csibra and Gergely, 2009). These observations have inspired attempts from the AI

research community to develop grounded interactive *agents* that are capable of engaging in natural back-and-forth dialog with humans to assist them in completing real-world tasks (Winograd, 1971; Narayan-Chen et al., 2017; Levinson, 2019; Chen et al., 2020).

Importantly, the agent needs to understand when to initiate feedback requests if communication fails or instructions are not clear and requires learning new domain-specific vocabulary (Aliannejadi et al., 2020; Rao and Daumé III, 2018; Narayan-Chen et al., 2019; Jayannavar et al., 2020). Despite all these efforts, the task is far from solved. For that reason, we propose the IGLU competition, which stands for *Interactive Grounded Language Understanding in a collaborative environment*.

Specifically, the goal of our competition is to approach the following scientific challenge: *How to build interactive agents that learn to solve a task while provided with grounded natural language instructions in a collaborative environment?*

By ‘*interactive agent*’ we mean that the agent is able to follow the instructions correctly, is able to ask for clarification when needed, and is able to quickly adapt newly acquired skills, just like humans are able to do while collaboratively interacting with each other.

The described research challenge is naturally related, but not limited, to two fields of study that are highly relevant to the NeurIPS community: Natural Language Understanding and Generation (NLU / NLG) and Reinforcement Learning (RL).

2. Background and Related Work

Relevance of NLU/G Natural language interfaces (NLIs) have been the “holy grail” of human-computer interaction and information search for decades (Woods et al., 1972; Codd, 1974; Hendrix et al., 1978). The recent advances in language understanding capability (Devlin et al., 2018; Liu et al., 2019; Clark et al., 2020; Adiwardana et al., 2020; Roller et al., 2020; Brown et al., 2020) powered by large-scale deep learning and increasing demand for new applications has led to a major resurgence of natural language interfaces in the form of virtual assistants, dialog systems, semantic parsing, and question answering systems (Liu and Lane, 2017, 2018; Dinan et al., 2020; Zhang et al., 2019). The horizon of NLIs has also been significantly expanding from databases (Copestake and Jones, 1990) to, knowledge bases (Berant et al., 2013), robots (Tellex et al., 2011), Internet of Things (virtual assistants like Siri and Alexa), Web service APIs (Su et al., 2017), and other forms of interaction (Fast et al., 2018; Desai et al., 2016; Young et al., 2013). Recent efforts have also focused on interactivity and continuous learning to enable agents to interact with users to resolve the knowledge gap between them for better accuracy and transparency. This includes systems that can learn new task from instructions (Li et al., 2020), assess their uncertainty (Yao et al., 2019), ask clarifying questions (Aliannejadi et al., 2020, 2021) and seek and leverage feedback from humans to correct mistakes (Elgohary et al., 2020).

Relevance of RL Recently developed RL methods celebrated successes for a number of tasks (Belle-mare et al., 2013; Mnih et al., 2015, 2016; Silver et al., 2017; Hessel et al., 2018). One of the aspects that helped to speed up RL methods development is game-based environments, which provide clear goals for an agent to achieve in flexible training settings. However, training RL agents that can follow human instructions has attracted fewer exploration (Chevalier-Boisvert et al., 2019; Cideron et al., 2019; Hu et al., 2019; Chen et al., 2020; Shu et al., 2017), due to complexity of the task and lack of proper experimental environments. Shu et al. (2017) proposed a hierarchical policy modulated by a stochastic temporal grammar for efficient multi-task reinforcement learning where each learned task corresponds to a human language description in Minecraft environment. The

BabyAI platform (Chevalier-Boisvert et al., 2019) aims to support investigations towards learning to perform language instructions with a simulated human in the loop. Chen et al. (2020) demonstrated that using step-by-step human demonstrations in the form of natural language instructions and action trajectories can facilitate the decomposition of complex tasks in a crafting environment.

Minecraft as an Environment for Grounded Language Understanding Szlam et al. (2019) substantiated the advantages of building an open interactive assistant in the sandbox construction game of Minecraft instead of a “real world” assistant, which is inherently complex and inherently costly to develop and maintain. The Minecraft world’s constraints (e.g., coarse 3-d voxel grid and simple physics) and the regularities in the head of the distribution of in-game tasks allow numerous scenarios for grounded NLU research (Yao et al., 2020; Srinet et al., 2020). Minecraft is an appealing competition domain due to its popularity as a video game, of all games ever released, it has the second-most total copies sold. Moreover, since it is a popular game environment, we can expect players to enjoy interacting with the assistants as they are developed, yielding a rich resource for a human-in-the-loop studies. Another important advantage of using Minecraft is the availability of the highly developed set of tools for logging agents interactions and deploying agents for evaluation with human-in-the-loop, including:

- *Malmo* (Johnson et al., 2016): a powerful platform for AI experimentation;
- *Craftassist* (Gray et al., 2019): a framework for dialog-enabled interactive agents¹;
- *TaskWorldMod* (Ogawa et al., 2020): a platform for situated task-oriented dialog data collection using gamification; and
- *MC-Saar-Instruct* (Köhn et al., 2020): a platform for Minecraft Instruction Giving Agents.

Besides, mainly due to the success of previous competitions (Guss et al., 2019; Perez-Liebana et al., 2019), Minecraft is a widely used environment by the RL community for experimentation with (mainly single) agents trained by demonstration. Therefore, using Minecraft would set a low barrier for the RL community to contribute to IGLU. To simplify the competition settings and possibly lower the entry bar for the NLU/NLG community, we will use simulated Blocks World in Minecraft (Jayannavar et al., 2020).

Relevance to Real Live Scenarios and Societal Impact Several important real-life scenarios have the potential to benefit from the results of our competition:

- **Education:** *Minecraft: Education Edition*² is a game-based learning platform that promotes creativity, collaboration, and problem-solving in an immersive digital environment. As of 2021, educators in more than 115 countries are using Minecraft across the curriculum. As stated in Ayoub, adding AI elements to this educational platform will move its potential to a new level. AI applications have the power to become a great equalizer in education. Students can get personalized education and scaffolding while being less dependent on uncontrollable factors such as the quality of their teachers or the amount of help they receive from their caregivers.
- **Robotics:** Bisk et al. (2016) proposed a protocol and framework for collecting data on human-robot interaction through natural language. The work demonstrated the potential for unrestricted contextually grounded communications between human and robots in blocks world. Developing robots to assist humans in different tasks at home has attracted much attention in the Robotics

1. CraftAssist is utilized for data collection

2. <https://education.minecraft.net/>

field (Stuckler et al., 2012). In fact, the Robocup@Home³ and the Room-Across-Room⁴ have run for several years. Given that the main human-robot interaction is through dialog, and the robot is supposed to assist the human in multiple tasks, we envision IGLU to enable more effective task grounded dialog training between human and robots.

There is a long history of competitions focused on NLU/G tasks. Especially in recent years we have seen a large number of challenges dedicated to open-domain dialog systems (Hauff et al., 2021; Dalton et al., 2020; Spina et al., 2019; Chuklin et al., 2018; Arguello et al., 2018), such as ConvAI (Burtsev and Logacheva, 2020), ConvAI2 (Dinan et al., 2020), ConvAI3: Clarifying Questions for Open-Domain Dialogue Systems (ClariQ) (Aliannejadi et al., 2020, 2021), as well as a series of competitions of the Alexa Prize⁵. There are great efforts in the community to advance task-oriented dialogs by suggesting competitions, such as the Dialog System Technology Challenge (DSTC-8) (Kim et al., 2019); benchmarks and experimental platforms, e.g., Convlab, which offers the annotated MultiWOZ dataset (Budzianowski et al., 2018) and associated pre-trained reference models (Lee et al., 2019). There are fewer attempts to study multi-modal dialog systems, e.g., Situated Interactive Multi-Modal Conversational Data Collection And Evaluation Platform (SIMMC) (Crook et al., 2019) or Audio Visual Scene-Aware Dialog Track (Hori et al., 2018).

There are a number of RL competitions such as MineRL (Guss et al., 2019) and MARLO (Perez-Liebana et al., 2019) that leverage the Minecraft environment. RL approaches have also been tried for text games environments, such as TextWold (Yuan et al., 2019)⁶ and Learning in Interactive Games with Humans and Text(Light) (Urbanek et al., 2019)⁷.

In comparison with previous efforts, to our knowledge, we are the first to propose a competition that tackles the task of grounded language understanding and interactive learning that brings together the NLU/G and RL research communities. The other key difference is our attempt to perform a human-in-the-loop evaluation as a final way for evaluating.

3. Data

The general setup IGLU is partially motivated by the HCRC Map Task Corpus (Thompson et al., 1993), which consists of route-following dialogs between an *Instruction Giver* and a *Follower*. Narayan-Chen et al. (2019) collected an openly available Minecraft dialog Corpus for a Collaborative Building Task. The authors used the following setup: the Architect is provided with a target structure that needs to be built by the Builder. The Architect provides instructions to the Builder on how to create the target structure and the Builder can ask clarifying questions to the Architect if an instruction is unclear Zhang et al. (2021). This dialog happens by means of a chat interface. The Architect is invisible to the Builder, but the Architect can see the actions of the Builder.

- 509 collected human-to-human dialogs along with RGB observations, and inventory information;
- games played over the course of 3 weeks (approx. 62 hours overall) with 40 volunteers. Each game took 8.55 minutes on average; and
- 163 tasks for the Architect-Builder collaborative game.

3. <https://athome.robocup.org/>

4. <https://ai.google.com/research/rxr/habitat>

5. <https://developer.amazon.com/alexaprize>

6. <https://www.microsoft.com/en-us/research/project/textworld/>

7. <https://parl.ai/projects/light/>

This dataset is used at the warm-up stage. To enable data collection outside of lab settings and bring it to the crowdsourcing platform, we adapted and extended the tool described in [Johnson et al. \(2016\)](#)⁸ We record the progression of each task, corresponding to the construction of a target structure by an Architect and Builder pair, as a discrete sequence of game observations. Each observation contains the following information: 1) a time stamp, 2) the chat history up until that point in time, 3) the Builder’s position (a tuple of real-valued x, y, z coordinates as well as pitch and yaw angles, representing the orientation of their camera), 4) the Builder’s block inventory, 5) the locations of the blocks in the build region. The final dataset contains 47 completed games, where the median duration of the game is 59 minutes. The collected dataset has 871 utterances, where the average length of utterance is 19, 32 words. Out of all utterances, 126 was clarifying questions.

During the competition, the submissions were evaluated in CodaLab. The CodaLab platform allows the submission system’s flexible design, leaderboard, metrics, and competition phases. Participants submit the code of their agents optionally with trained model weights and all other necessary files as one zip file. During the evaluation, the automated platform receives a solution, runs several episodes of the IGLU environment or feeds the model with several dataset utterances depending on a track of the competition, and calculates automatic evaluation scores.

4. IGLU Tasks

Given the current state of the field, our main research challenge (i.e., *how to build interactive agents that learn to solve a task while provided with grounded natural language instructions in a collaborative environment*) might be too complex to suggest a reasonable end-to-end solution. Therefore, we split the problem into the following concrete research questions, which correspond to separate tasks that can be used to study each component individually before joining all of them into one system ([Jones, 1988](#)):

RQ1 *How to teach?*

In other words, what is the best strategy for an Architect when instructing a Builder agent, such that the concept is reasonably explained? (The suggested task is presented in Section 4.1).

RQ2 *How to learn?*

That is, what methods should be used to train a Builder that can follow given instructions from an Architect?

This question can be further split into two sub-questions:

RQ2.1 *How is a ‘silent’ Builder able to learn?*

A silent Builder follows instructions without the ability to ask for any clarification from the Architect. (The suggested task is presented in Section 4.2).

RQ2.2 *How is an ‘interactive’ Builder able to learn?*

An interactive Builder can ask clarifying questions to the Architect to gain more information about the task in case of uncertainty. (Due to difficulty we leave this task for exploration in future competitions).

4.1. Task 1: Architect

In this task, our goal is to develop an Architect that can generate appropriate step instructions based on the observations of environment and the Builder’s behavior. At the beginning of each task, we give

8. Here is the visualization of the resulted data <https://youtu.be/Ls6Wv7EUDA0>

all the details of the target structure (e.g., types, colors and coordinated of blocks) to the Architect. The Architect needs to decompose the building process of this compound structure into a sequence of step instructions that the Builder can follow. During the interaction, the Architect has to compare the half-finished structure with the target structure and guide the Builder to complete the building of remaining components via generated instructions. The step instructions can be neither too detailed nor too general. In summary, the Architect is expected to be able to give instructions, correct the Builders’ mistakes and answer their questions by comparing the built structure against the target structure and by understanding the preceding dialog⁹.

4.1.1. TASK SETUP

We aim to generate a suitable Architect utterance, given access to 1) the detailed information of the target structure and 2) the entire game state context leading up to a certain point in a human-human game at which the human Architect spoke next. This task can be seen as a multimodal text generation, where the target structure, the built structure and the dialog history are input and the next Architect’s utterance is the output. The model developed for this task can involve both language understanding and visual understanding depending on the methods for world state representations.

4.1.2. EVALUATION AND BASELINE

Automatic evaluation To evaluate how closely the generated utterances resemble the human utterances, we adopt standard BLEU scores (Papineni et al., 2002). We also make use of the modified *Precision* and *Recall* of domain-specific keywords defined in Narayan-Chen et al. (2019). The defined keywords are instrumental to task success, including colors, spatial relations, and other words that are highly indicative of dialog actions.

We provide two baselines that are presented in Narayan-Chen et al. (2019)¹⁰. We reproduce the results on the test set presented in (Narayan-Chen et al., 2019) in Table 1. The hyper-parameters of architect models have been fine-tuned on the validation set. By augmenting both the global world state and local world state, Seq2Seq with global and local information managed to show noticeable improvements on each of the automatic metrics. The provided baseline definitely leaves room for improvement. All the architect models will be re-evaluated after we collect a larger dialog corpus. In the course of the competition, none of the teams could outperform the suggested baseline.

Metrics	BLEU				Precision/ Recall			
	BLEU-1	BLEU-2	BLEU-3	BLEU-4	all keywords	colors	spatial	dialog
Seq2Seq	15.3	7.8	4.5	2.8	11.8/11.1	8.1/17.0	9.3/8.6	17.9/19.3
+global & local	15.7	8.1	4.8	2.9	13.5/14.4	14.9/28.7	8.7/8.7	18.5/19.9

Table 1: BLEU and term-specific precision and recall scores on the test set, originally reported in (Narayan-Chen et al., 2019), which were able to reproduce.

4.2. Task 2: Silent Builder

The overall setup for training initial baselines for the silent Builder that will be used for comparison is presented in Figure 1.

9. We can see the similarity with newly published work on using human instructions to improve generalization in RL (Chen et al., 2020).

10. The implementation of the baseline can be found using the following repository <https://github.com/prashant-jayan21/minecraft-dialogue-models>.

4.2.1. IGLU ENVIRONMENT

As the main tool for training grounded agents, we built a gym environment¹¹ that allows training embodied agents in the Minecraft-like gridworld. In the environment, the agent is placed at the center of an empty building zone marked with white blocks. The agent can navigate over the world, place, and break blocks. It has an inventory with six different types of blocks: blue, red, green, orange, purple, and yellow. This setup mimics the data collection setting of Narayan-Chen et al. (2019). The goal of the agent is to analyze the past human-human dialog and reproduce a structure that was a result of this dialog.

The observation space consists of a POV image (64, 64, 3), inventory item counts (6,), building zone snapshot (11, 11, 9), full dialog as string, and the agent (X, Y, Z) position with two angles (pitch, yaw) (5,). In each cell of the building zone grid component there is an ID that defines the block type in that position (e.g., 0 for air, 1 for blue block, etc.). The action space can be chosen from three possible options: human-level actions, discrete actions, and continuous actions. For human-level control, we employ the same action space as is used in MineRL environment Guss et al. (2019). That is, the agent can simulate key press to move forward, backward, left, right, to jump, also it can control the player’s camera with a two dimensional continuous vector. It can place a block by simulating right click press and break by simulating a left one. Last, the block type can be chosen using a specific action. For the discrete type of control, we restrict our agent to step over the centers of blocks. The navigation actions are the same but they are discrete. All other actions are the same. The continuous control also changes only navigation actions. With it, the agent can fly freely inside the building zone. The navigation action is a three dimensional continuous vector that describes a shift direction for an agent.

For simplicity of the task, we allow an agent to build a target structure anywhere inside the building zone by comparing all relative translations and rotations of two zones. The environment calculates intersections over two zones and takes the maximum over these intersections. The agent then receives a reward according to the best match if it has been changed (i.e., non-zero reward if and only if the max match was changed since the last action). If the agent gets closer to the target structure in terms of the maximal match, it receives a reward +2. If the structure built by the agent moves further away in terms of a maximal match from the target one (e.g., the agent removes a correctly

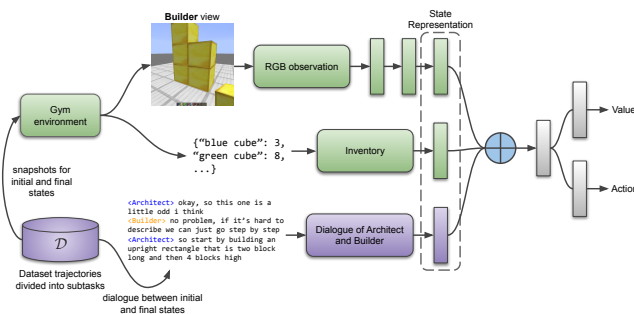


Figure 1: The overall pipeline for suggested baselines. That is, the agent can simulate key press to move forward, backward, left, right, to jump, also it can control the player’s camera with a two dimensional continuous vector. It can place a block by simulating right click press and break by simulating a left one. Last, the block type can be chosen using a specific action. For the discrete type of control, we restrict our agent to step over the centers of blocks. The navigation actions are the same but they are discrete. All other actions are the same. The continuous control also changes only navigation actions. With it, the agent can fly freely inside the building zone. The navigation action is a three dimensional continuous vector that describes a shift direction for an agent.

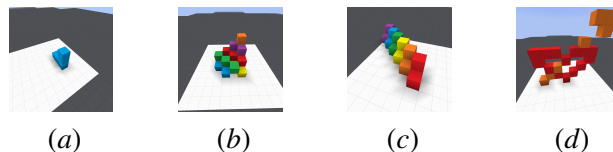


Figure 2: Example structures for the IGLU environment.

receives a reward according to the best match if it has been changed (i.e., non-zero reward if and only if the max match was changed since the last action). If the agent gets closer to the target structure in terms of the maximal match, it receives a reward +2. If the structure built by the agent moves further away in terms of a maximal match from the target one (e.g., the agent removes a correctly

11. <https://github.com/iglu-contest/iglu>

placed block), the agent gets a reward of -2 . Otherwise, if the agent places/removes a block outside of a partially built structure (i.e., without changing maximal intersections of structures), it receives a reward of $-1/+1$ respectively. If the agent moves outside of the building zone, then the environment terminates immediately.

Architect simulator for training the silent Builder The correct instruction sequence to achieve the final goal is known since the target structures are associated with dialogs. We annotated each dialog’s sub-goals and stored them in a queue, where each sub-goal corresponds to one specific step instruction from the Architect. We pop up a sub-task (e.g., in about the middle, build a column five tall) and wait until the agent completes it. If the agent completes this sub-task, we pop up the next sub-task. We trained a matching model to decide if the current sub-task is finished.

4.2.2. EVALUATION

For each subtask, the submitted agent is evaluated on the environment, with fixed initial and target states. We used the following three metrics:

- The reward score S_r is based on the average reward received by the agent in evaluation episodes, which is calculated as follows: $S_r = \frac{1}{N} \sum_{i=1}^N g_i$, where N is a number of evaluation episodes, g_i is episode reward, defined by $g_i = \sum_{t=1}^T r_t$.
- The success score S_s indicates the number completely solved subtasks: $S_s = \frac{1}{N} \sum_{i=1}^N c_i$, where

$$c_i = \begin{cases} +1, & \text{if success,} \\ 0, & \text{otherwise.} \end{cases}$$

- Completion rate score S_c : $S_c = \frac{1}{N} \sum_{i=1}^N 1 - \rho_i$, where ρ is a normalized Hamming distance between target and built structures.

4.2.3. BASELINES

For the Silent Builder track, we provided several baselines, each training for a single target structure. First, we built grid-based agents implemented in RLlib (Liang et al., 2017) framework. Agents were implemented with APE-X (Horgan et al., 2018) and Rainbow (Hessel et al., 2017) algorithms. Due to the combinatorial nature of the task, grid-baselines were able to solve only simplest possible tasks such as 5 block L-shaped structure. Second, we implemented a purely visual model-free agent using IMPALA algorithm (Espeholt et al., 2018) implemented with RLlib. The IMPALA agent yields a better performance achieving the normalized average reward of 12 in 20 million steps. This corresponds to 12 correctly placed blocks in the 18 blocks task. Last, we provided a visual model-based baseline agent which uses the Dreamer algorithm (Hafner et al., 2019, 2020). The Dreamer agent shows similar performance to the model-free IMPALA agent but requires only 2 million steps to converge. The baselines are open sourced on Github^{12,13}.

4.2.4. WINNING SOLUTIONS

Team 1: Hybrid Intelligence The Hybrid Intelligence team competed in the Builder track. Their solution improves upon the random agent by inducing a more accurate color distribution. They

12. APE-X, Rainbow, IMPALA: <https://github.com/iglu-contest/iglu-builder-baseline-rlib>

13. Dreamer: <https://github.com/iglu-contest/iglu-builder-baseline-dreamer>

first extract a set of colors from the chat data via multilabel classification. When the random agent samples a hotbar action, which contains a color choice, with probability $1 - \epsilon$ they instead equip a color sampled at random from the extracted set. This allows the agent to explore with probability ϵ via equipping colors that have not been extracted by the multilabel classification.

Team 2: NeuroAI The NeuroAI team trained a three stage compound model, which consists of: the pretrained 3D goal grid predictor that uses the text conversation as input, a convolutional autoencoder pretrained on random episodes, a policy that acts on a combination of the outputs of the previous two. The text-to-target predictor fine-tunes a head on top of the frozen Distil-RoBERTa model. The policy uses cross-attention with image embeddings as keys and values and goal grid as query followed by action and value heads. The agent was trained with PPO on an augmented set of tasks with a custom curriculum.

4.2.5. RESULTS

Silent Builder Track Results By the end of the main competition, the Hybrid Intelligence team was placed first with a completion rate score of 0.365. The NeuroAI team took second place with the leading score of 0.34. In total, there were 96 submissions and 37 registered participants.

Human Evaluation After the main challenge, we ran human evaluations on the best agents submitted by both teams. We evaluated the agent’s ability to respond to specific instructions correctly. To do this, each game episode was initialized with a prefix of dialog utterances and a partially built structure that corresponded to this dialog prefix. The last utterance defined an open goal with a result not presented in partial structure. With all that given, we measured the ability of the agent to respond to the instruction with actions correctly. Three judges performed the evaluation. We revealed a series of instructions to each of them. The instructions were combined with a pair of videos with action responses of each of the agents. Evaluators had to vote for the agent, which they thought performed better. Evaluators did not know which agent each animation corresponds to. The human evaluation confirmed the results of the automatic one, with the first team obtaining an average vote rate of 0.88 and Krippendorff’s disagreement score of 0.56.

5. Conclusions and Future Work

IGLU has proposed unique and novel directions towards building collaborative embodied agents. In the course of the challenge, we have collected a new dataset suitable for NLP- and RL-related experiments. We developed a gym-like environment for training agents that takes instructions in the natural language as part of the observation, which should enable research in this area. The results of the completion clearly shows that the proposed direction is challenging and requires more exploration in the near future.

As for future direction, we will focus on investing more into the designing methodology for the builder automatic evaluation as well as human-in-the-loop one and specifying task for the interactive builder.

Acknowledgement

We thank our sponsors Microsoft, AIRI, and Google. We would like to thank our advisory board: Julia Hockenmaier, Bill Dolan, Ryen W. White, Maarten de Rijke, and Sharada Mohanty.

References

- Daniel Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, et al. Towards a human-like open-domain chatbot. *arXiv preprint arXiv:2001.09977*, 2020.
- Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. Convai3: Generating clarifying questions for open-domain dialogue systems (clariq). 2020.
- Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeffrey Dalton, and Mikhail Burtsev. Building and evaluating open-domain dialogue corpora with clarifying questions. *arXiv preprint arXiv:2109.05794*, 2021.
- Meltzoff An. Imitation, Objects, Tools, and the Rudiments of Language in Human Ontogeny, February 1988. URL <https://pubmed.ncbi.nlm.nih.gov/23997403/>. ISSN: 0393-9375 Issue: 1-2 Publisher: Hum Evol Volume: 3.
- Jaime Arguello, Filip Radlinski, Hideo Joho, Damiano Spina, and Julia Kiseleva. Second international workshop on conversational approaches to information retrieval (cair'18) workshop at sigir 2018. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1435–1437, 2018.
- Dan Ayoub. Unleashing the power of AI for education. URL <https://www.technologyreview.com/2020/03/04/905535/unleashing-the-power-of-ai-for-education/>.
- Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47: 253–279, 2013.
- Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544, 2013.
- Yonatan Bisk, Deniz Yuret, and Daniel Marcu. Natural language communication with robots. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 751–761, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1089. URL <https://www.aclweb.org/anthology/N16-1089>.
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. 2020.

- Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 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, 2018.
- Mikhail Burtsev and Varvara Logacheva. Conversational intelligence challenge: Accelerating research with crowd science and open source. *AI Magazine*, 41(3):18–27, 2020.
- Valerie Chen, Abhinav Gupta, and Kenneth Marino. Ask your humans: Using human instructions to improve generalization in reinforcement learning. *arXiv preprint arXiv:2011.00517*, 2020.
- Maxime Chevalier-Boisvert, Dzmitry Bahdanau, Salem Lahlou, Lucas Willems, Chitwan Saharia, Thien Huu Nguyen, and Yoshua Bengio. Babyai: First steps towards grounded language learning with a human in the loop. In *International Conference on Learning Representations*, 2019.
- Aleksandr Chuklin, Jeff Dalton, Julia Kiseleva, Alexey Borisov, and Mikhail Burtsev. Proceedings of the 2018 emnlp workshop scai: The 2nd international workshop on search-oriented conversational ai. In *Proceedings of the 2018 EMNLP Workshop SCAI: The 2nd International Workshop on Search-Oriented Conversational AI*, 2018.
- Geoffrey Cideron, Mathieu Seurin, Florian Strub, and Olivier Pietquin. Self-educated language agent with hindsight experience replay for instruction following. *arXiv preprint arXiv:1910.09451*, 2019.
- Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*, 2020.
- Edgar F Codd. *Seven steps to rendezvous with the casual user*. IBM Corporation, 1974.
- Ann Copestake and Karen Sparck Jones. Natural language interfaces to databases. 1990.
- National Research Council. *How People Learn: Brain, Mind, Experience, and School: Expanded Edition*. August 1999. ISBN 978-0-309-07036-2. doi: 10.17226/9853. URL <https://www.nap.edu/catalog/9853/how-people-learn-brain-mind-experience-and-school-expanded-edition>.
- Paul A Crook, Shivani Poddar, Ankita De, Semir Shafi, David Whitney, Alborz Geramifard, and Rajen Subba. Simmc: Situated interactive multi-modal conversational data collection and evaluation platform. *arXiv preprint arXiv:1911.02690*, 2019.
- Gergely Csibra and György Gergely. Natural pedagogy. *Trends in cognitive sciences*, 13(4):148–153, 2009.
- Jeff Dalton, Aleksandr Chuklin, Julia Kiseleva, and Mikhail Burtsev, editors. *Proceedings of the 5th International Workshop on Search-Oriented Conversational AI (SCAI)*, Online, November 2020. Association for Computational Linguistics. URL <https://aclanthology.org/2020.scai-1.0>.
- Aditya Desai, Sumit Gulwani, Vineet Hingorani, Nidhi Jain, Amey Karkare, Mark Marron, Subhjit Roy, et al. Program synthesis using natural language. In *Proceedings of the 38th International Conference on Software Engineering*, pages 345–356. ACM, 2016.

- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, 2018.
- Emily Dinan, Varvara Logacheva, Valentin Malykh, Alexander Miller, Kurt Shuster, Jack Urbanek, Douwe Kiela, Arthur Szlam, Iulian Serban, Ryan Lowe, et al. The second conversational intelligence challenge (convai2). In *The NeurIPS'18 Competition*, pages 187–208. Springer, Cham, 2020.
- Ahmed Elgohary, Saghar Hosseini, and Ahmed Hassan Awadallah. Speak to your parser: Interactive text-to-SQL with natural language feedback. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2065–2077, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.187. URL <https://www.aclweb.org/anthology/2020.acl-main.187>.
- Lasse Espeholt, Hubert Soyer, Rémi Munos, Karen Simonyan, Volodymyr Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane Legg, and Koray Kavukcuoglu. IMPALA: scalable distributed deep-rl with importance weighted actor-learner architectures. *CoRR*, abs/1802.01561, 2018. URL <http://arxiv.org/abs/1802.01561>.
- Ethan Fast, Binbin Chen, Julia Mendelsohn, Jonathan Bassen, and Michael S Bernstein. Iris: A conversational agent for complex tasks. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 473. ACM, 2018.
- Jonathan Gray, Kavya Srinet, Yacine Jernite, Haonan Yu, Zhuoyuan Chen, Demi Guo, Siddharth Goyal, C. Lawrence Zitnick, and Arthur Szlam. CraftAssist: A Framework for Dialogue-enabled Interactive Agents. *arXiv:1907.08584 [cs]*, July 2019. URL <http://arxiv.org/abs/1907.08584>. arXiv: 1907.08584.
- William H. Guss, Cayden Codell, Katja Hofmann, Brandon Houghton, Noboru Kuno, Stephanie Milani, Sharada Mohanty, Diego Perez Liebana, Ruslan Salakhutdinov, Nicholay Topin, et al. The MineRL competition on sample efficient reinforcement learning using human priors. *NeurIPS Competition Track*, 2019.
- Danijar Hafner, Timothy P. Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *CoRR*, abs/1912.01603, 2019. URL <http://arxiv.org/abs/1912.01603>.
- Danijar Hafner, Timothy P. Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. *CoRR*, abs/2010.02193, 2020. URL <https://arxiv.org/abs/2010.02193>.
- Claudia Hauff, Julia Kiseleva, Mark Sanderson, Hamed Zamani, and Yongfeng Zhang. Conversational search and recommendation: Introduction to the special issue. *ACM Trans. Inf. Syst.*, 39(4), August 2021. ISSN 1046-8188. doi: 10.1145/3465272. URL <https://doi.org/10.1145/3465272>.
- Gary G Hendrix, Earl D Sacerdoti, Daniel Sagalowicz, and Jonathan Slocum. Developing a natural language interface to complex data. *ACM Transactions on Database Systems (TODS)*, 3(2): 105–147, 1978.

- Matteo Hessel, Joseph Modayil, Hado van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Daniel Horgan, Bilal Piot, Mohammad Gheshlaghi Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. *CoRR*, abs/1710.02298, 2017. URL <http://arxiv.org/abs/1710.02298>.
- Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.
- Dan Horgan, John Quan, David Budden, Gabriel Barth-Maron, Matteo Hessel, Hado van Hasselt, and David Silver. Distributed prioritized experience replay. *CoRR*, abs/1803.00933, 2018. URL <http://arxiv.org/abs/1803.00933>.
- Chiori Hori, Anoop Cherian, Tim K Marks, and Florian Metze. Audio visual scene-aware dialog track in dstc8. *DSTC Track Proposal*, 2018.
- Hengyuan Hu, Denis Yarats, Qucheng Gong, Yuandong Tian, and Mike Lewis. Hierarchical decision making by generating and following natural language instructions. *arXiv preprint arXiv:1906.00744*, 2019.
- Prashant Jayannavar, Anjali Narayan-Chen, and Julia Hockenmaier. Learning to execute instructions in a Minecraft dialogue. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2589–2602, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.232. URL <https://www.aclweb.org/anthology/2020.acl-main.232>.
- Matthew Johnson, Katja Hofmann, Tim Hutton, and David Bignell. The malmo platform for artificial intelligence experimentation. In *IJCAI*, pages 4246–4247. Citeseer, 2016.
- K Sparck Jones. A look back and a look forward. In *Proceedings of the 11th annual international ACM SIGIR conference on Research and Development in Information Retrieval*, pages 13–29, 1988.
- Seokhwan Kim, Michel Galley, Chulaka Gunasekara, Sungjin Lee, Adam Atkinson, Baolin Peng, Hannes Schulz, Jianfeng Gao, Jinchao Li, Mahmoud Adada, et al. The eighth dialog system technology challenge. *arXiv preprint arXiv:1911.06394*, 2019.
- Arne Köhn, Julia Wichlacz, Christine Schäfer, Alvaro Torralba, Jörg Hoffmann, and Alexander Koller. Mc-saar-instruct: a platform for minecraft instruction giving agents. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 53–56, 2020.
- Sungjin Lee, Qi Zhu, Ryuichi Takanobu, Xiang Li, Yaoqin Zhang, Zheng Zhang, Jinchao Li, Baolin Peng, Xiujun Li, Minlie Huang, et al. Convlab: Multi-domain end-to-end dialog system platform. *arXiv preprint arXiv:1904.08637*, 2019.
- Stephen C Levinson. Tom m. mitchell, simon garrod, john e. laird, stephen c. levinson, and kenneth r. koedinger. *Interactive Task Learning: Humans, Robots, and Agents Acquiring New Tasks through Natural Interactions*, 26:9, 2019.

- Toby Jia-Jun Li, Tom Mitchell, and Brad Myers. Interactive task learning from GUI-grounded natural language instructions and demonstrations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, July 2020.
- Eric Liang, Richard Liaw, Robert Nishihara, Philipp Moritz, Roy Fox, Joseph Gonzalez, Ken Goldberg, and Ion Stoica. Ray rllib: A composable and scalable reinforcement learning library. *CoRR*, abs/1712.09381, 2017. URL <http://arxiv.org/abs/1712.09381>.
- Bing Liu and Ian Lane. Iterative policy learning in end-to-end trainable task-oriented neural dialog models. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 482–489. IEEE, 2017.
- Bing Liu and Ian Lane. Adversarial learning of task-oriented neural dialog models. In *Proceedings of the SIGDIAL 2018 Conference*, pages 350–359, 2018.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. URL <http://arxiv.org/abs/1907.11692>.
- Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.
- Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937. PMLR, 2016.
- Anjali Narayan-Chen, Colin Graber, Mayukh Das, Md Rakibul Islam, Soham Dan, Sriraam Natarajan, Janardhan Rao Doppa, Julia Hockenmaier, Martha Palmer, and Dan Roth. Towards problem solving agents that communicate and learn. In *Proceedings of the First Workshop on Language Grounding for Robotics*, pages 95–103, 2017.
- Anjali Narayan-Chen, Prashant Jayannavar, and Julia Hockenmaier. Collaborative dialogue in minecraft. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5405–5415, 2019.
- Haruna Ogawa, Hitoshi Nishikawa, Takenobu Tokunaga, and Hikaru Yokono. Gamification platform for collecting task-oriented dialogue data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 7084–7093, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL <https://www.aclweb.org/anthology/2020.lrec-1.876>.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.
- Diego Perez-Liebana, Katja Hofmann, Sharada Prasanna Mohanty, Noburu Kuno, Andre Kramer, Sam Devlin, Raluca D Gaina, and Daniel Ionita. The multi-agent reinforcement learning in malm\” o (marl\” o) competition. *arXiv preprint arXiv:1901.08129*, 2019.

- Sudha Rao and Hal Daumé III. Learning to ask good questions: Ranking clarification questions using neural expected value of perfect information. *arXiv preprint arXiv:1805.04655*, 2018.
- Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M Smith, et al. Recipes for building an open-domain chatbot. *arXiv preprint arXiv:2004.13637*, 2020.
- Tianmin Shu, Caiming Xiong, and Richard Socher. Hierarchical and interpretable skill acquisition in multi-task reinforcement learning. *arXiv preprint arXiv:1712.07294*, 2017.
- David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.
- Damiano Spina, Jaime Arguello, Hideo Joho, Julia Kiseleva, and Filip Radlinski. Cair’18: second international workshop on conversational approaches to information retrieval at sigir 2018. In *ACM SIGIR Forum*, volume 52, pages 111–116. ACM New York, NY, USA, 2019.
- Kavya Srinet, Yacine Jernite, Jonathan Gray, and Arthur Szlam. CraftAssist instruction parsing: Semantic parsing for a voxel-world assistant. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4693–4714, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.427. URL <https://www.aclweb.org/anthology/2020.acl-main.427>.
- Jorg Stuckler, Dirk Holz, and Sven Behnke. Robocup@ home: Demonstrating everyday manipulation skills in robocup@ home. *IEEE Robotics & Automation Magazine*, 19(2):34–42, 2012.
- Yu Su, Ahmed Hassan Awadallah, Madian Khabsa, Patrick Pantel, Michael Gamon, and Mark Encarnacion. Building natural language interfaces to web apis. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 177–186. ACM, 2017.
- Arthur Szlam, Jonathan Gray, Kavya Srinet, Yacine Jernite, Armand Joulin, Gabriel Synnaeve, Douwe Kiela, Haonan Yu, Zhuoyuan Chen, Siddharth Goyal, Demi Guo, Danielle Rothermel, C. Lawrence Zitnick, and Jason Weston. Why Build an Assistant in Minecraft? *arXiv:1907.09273 [cs]*, July 2019. URL <http://arxiv.org/abs/1907.09273>. arXiv: 1907.09273.
- Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R Walter, Ashis Gopal Banerjee, Seth Teller, and Nicholas Roy. Understanding natural language commands for robotic navigation and mobile manipulation. In *Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.
- Andrea L Thomaz, Elena Lieven, Maya Cakmak, Joyce Y Chai, Simon Garrod, Wayne D Gray, Stephen C Levinson, Ana Paiva, and Nele Russwinkel. Interaction for task instruction and learning. In *Interactive task learning: Humans, robots, and agents acquiring new tasks through natural interactions*, pages 91–110. MIT Press, 2019.
- Henry S. Thompson, Anne Anderson, Ellen Gurman Bard, Gwyneth Doherty-Sneddon, Alison Newlands, and Cathy Sotillo. The HCRC Map Task corpus: natural dialogue for speech recognition. In *Proceedings of the workshop on Human Language Technology - HLT '93*, page 25,

- Princeton, New Jersey, 1993. Association for Computational Linguistics. ISBN 978-1-55860-324-0. doi: 10.3115/1075671.1075677. URL <http://portal.acm.org/citation.cfm?doid=1075671.1075677>.
- Jack Urbanek, Angela Fan, Siddharth Karamcheti, Saachi Jain, Samuel Humeau, Emily Dinan, Tim Rocktäschel, Douwe Kiela, Arthur Szlam, and Jason Weston. Learning to speak and act in a fantasy text adventure game. *arXiv preprint arXiv:1903.03094*, 2019.
- Terry Winograd. Procedures as a representation for data in a computer program for understanding natural language. Technical report, MASSACHUSETTS INST OF TECH CAMBRIDGE PROJECT MAC, 1971.
- W. A. Woods, Ronald M Kaplan, and Bonnie L. Webber. The lunar sciences natural language information system: Final report. *BBN Report 2378*, 1972.
- Ziyu Yao, Yu Su, Huan Sun, and Wen-tau Yih. Model-based interactive semantic parsing: A unified framework and a text-to-SQL case study. 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 5447–5458, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1547. URL <https://www.aclweb.org/anthology/D19-1547>.
- Ziyu Yao, Yiqi Tang, Wen-tau Yih, Huan Sun, and Yu Su. An imitation game for learning semantic parsers from user interaction. 2020.
- Steve Young, Milica Gašić, Blaise Thomson, and Jason D Williams. Pomdp-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101(5):1160–1179, 2013.
- Xingdi Yuan, Marc-Alexandre Côté, Jie Fu, Zhouhan Lin, Christopher Pal, Yoshua Bengio, and Adam Trischler. Interactive language learning by question answering. *arXiv preprint arXiv:1908.10909*, 2019.
- Yi Zhang, Sujay Kumar Jauhar, Julia Kiseleva, Ryen White, and Dan Roth. Learning to decompose and organize complex tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2726–2735, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.217. URL <https://aclanthology.org/2021.naacl-main.217>.
- Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536*, 2019.