Collecting Interactive Multi-modal Datasets for Grounded Language Understanding

Shrestha Mohanty[1]*, **Negar Arabzadeh**[2]*, **Milagro Teruel**[1], **Yuxuan Sun**[3],
Artem Zholus[4], **Alexey Skrynnik**[3,4], **Mikhail Burtsev**[4,5], **Kavya Srinet**[3],
Aleksandr Panov[4,5], **Arthur Szlam**[3], **Marc-Alexandre Côté**[1], **Julia kiseleva**[1]

Microsoft Research[1], University of Waterloo[2], Meta Research[3],
Moscow Institute of Physics and Technology[4], Artificial Intelligence Research Institute[5]

Abstract

Human intelligence can remarkably adapt quickly 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 which can enable similar capabilities in machines, we made the following contributions (1) formalized the collaborative embodied agent using natural language task; (2) developed a tool for extensive and scalable data collection; and (3) collected the first dataset for interactive grounded language understanding.

1 Introduction

Humans learn and acquire new skills by imitating others or by following instructions in natural language [An, 1988, Council, 1999]. Studies in developmental psychology have shown evidence of natural language communication being an effective method for imparting generic knowledge to individuals as young as infants [Csibra and Gergely, 2009]. This form of learning can even accelerate the acquisition of new skills by avoiding trial-and-error when learning occurs only from observations [Thomaz et al., 2019]. Inspired by this, the AI research community has attempted to develop grounded interactive embodied agents Kiseleva et al. [2021, 2022a], Kojima et al. [2021] 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, Abramson et al., 2020]. Notably, 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, 2021, Rao and Daumé III, 2018, Narayan-Chen et al., 2019, Jayannavar et al., 2020]. Despite all these efforts, the task is far from solved. One of the biggest challenges is lack of general approaches for data collection and human-in-the-loop evaluation, where the collaborative agent is paired with a human.

In this paper, we present the following contributions:

C1 formalization of a task that allows studying collaborative embodied agents that use natural language for communication (Section 2);

C2 a large dataset to enable research for the above task;

C3 comprehensive open-sourced tooling for scalable data collection and to allow reproduction of experiments and results;
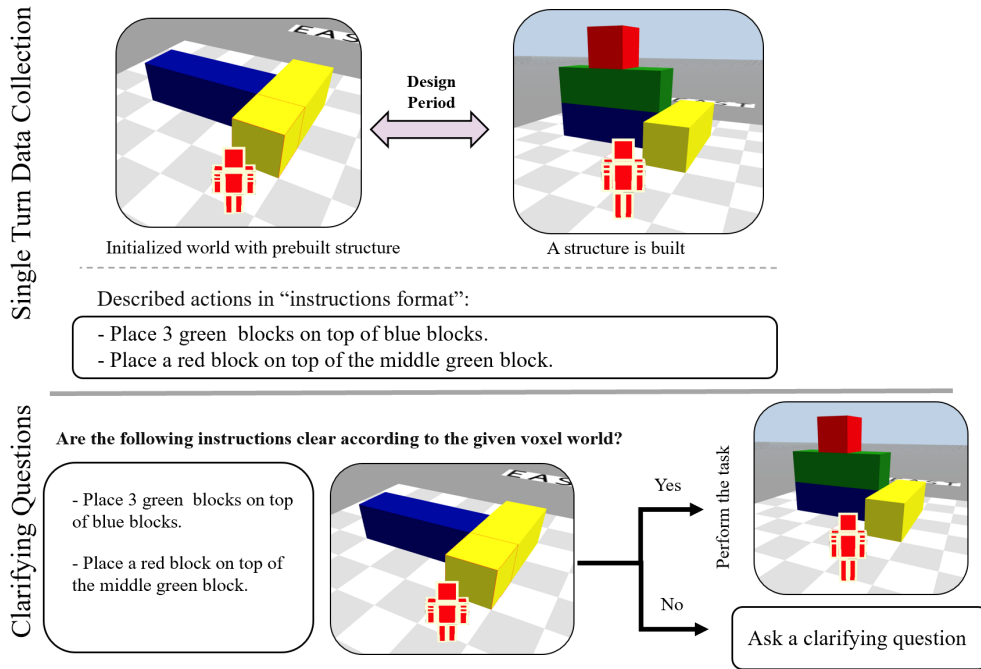
---

*Equal contribution

Figure 1: Overview of data collection for a single turn and clarifying questions

2 Collecting interactive data

The goal of the Collaborative Building Task, similar to Kiseleva et al. [2022a], Jayannavar et al. [2020], Narayan-Chen et al. [2019], is to train interactive embodied 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 can: (1) follow the instructions provided in natural language in relation to the current world correctly, (2) ask for clarification in case of uncertainty, and (3) quickly adapt to newly acquired skills.

Narayan-Chen et al. [2019] has proposed the following setup: An 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, while the Architect can see the actions of the Builder. Narayan-Chen et al. [2019] required installing Microsoft's Project Malmo Johnson et al. [2016] client, which provides an API for Minecraft agents to chat, build, and the ability to save and load game states. This is used to collect multi-turn interactions between the Architect and the Builder collaboratively working towards the common goal of building a given target structure. However, the data collection setup is limited to lab-based studies, which prevents massive online data collection.

2.1 Single-Turn Data Collection

Scaling up data collection   We have leveraged and extended the previously collected multi-turn interactions dataset in our work. For that, we modified the data collection strategy by removing the need for installing a local Minecraft client. We use an online crowdsourcing platform such as Amazon Mechanical Turk[2] where we integrate our data collection tool, which leverages the CraftAssisft library Srinet et al. [2020]. Our data collection tool can be easily plugged into the crowdsourcing platform enabling us to scale for participants quickly.

---

[2]https://www.mturk.com/

Simplifying the task We further breakdown and simplify the multi-turn dialogues interactions to single-turn interactions. We do this by removing the added complexity of building a target structure and instead having an agent perform actions and provide instructions to another agent. We also leverage the multi-turn interactions data collected earlier to provide a starting point from which agents can build. To elaborate, we design on the following setup, as shown in Figure 1, for collecting data:

- An interactive agent is dropped in the middle of an ongoing game where the structure is built partially. The partially completed game is retrieved from the multi-turn interactions dataset mentioned above.

- The agent is prompted to perform a sequence of actions for a duration of one minute.

- After which, the agent describes their performed set of actions in natural language, which will be displayed to another agent as an instruction.

- The next agent is shown the instruction and is asked to perform the steps mentioned in the instruction. If the instruction is not clear, the agent specifies it as thus and asks clarification questions.

This setup enables us to collect a dataset consisting of natural language instructions, actions performed based on those instructions, and a set of clarifying questions (described more in Section 2.2). Since the current data collection is limited to building-related tasks, the agent and builder are interchangeably used. Currently, we are working on enhancing the data collection tool to allow agents to perform various tasks, such as 'grab,' 'bring,' etc., to convert a builder into an actor. The dataset can be used to train models and agents towards solving tasks such as *instruction generation* and *building structures given natural language instructions*.

2.2 Clarifying Questions

One of the advantages of collecting data in individual single turns is that the independence of every sample in the collected data will allow it to be used for different tasks more easily. In other words, it could be plugged into different settings, and each turn can be interpreted separately as a complete set of information for that turn. For instance, single-turn data provide a great host for collecting a comprehensive dataset for the *Asking Clarifying Question* task. This has been studied in Information Retrieval, and Natural Language Processing domain to some extent Aliannejadi et al. [2021, 2020, 2019a], Braslavski et al. [2017], Zamani et al. [2020]. While *what to ask* has been relatively extensively studied Zamani et al. [2020], Aliannejadi et al. [2021, 2019b], Braslavski et al. [2017], Rao and Daumé III [2018], Stoyanchev et al. [2014], De Boni and Manandhar [2003], Sekulić et al. [2022], Sekulić et al. [2021], *when to ask* clarifying questions in the case of unclear instructions or queries has been understudied and explored to limited extent. To the best of our knowledge Aliannejadi et al. [2021] were the first who explored both research questions i,e., *when* and *what* to ask in one work[3]. The advantage of collecting such data in this way is that *first*, it is close to real-world problems and applications, and *second*, it has the potential to be used to evaluate the whole pipeline all at once. In other words, it can evaluate how often a system decided correctly to issue clarifying questions and further more if they are asking appropriate clarifying questions in that case. While *when to ask* clarifying questions are usually treated as a classification task, i.e., either the system asks a clarifying question or not, the second question can be answered in generation mode, i.e., a model generates a clarifying question, and the data can be used to evaluate how close the generated questions are to the target clarifying questions. As another option, the data can be used as a ranking problem in which the model is supposed to find the most relevant clarifying question from a corpus of all collected clarifying questions.

To collect such a dataset, we leveraged the instructions in Section 2.1, and as shown in Figure 1, we asked annotators if the instructions were clear. In case they found the instruction clear enough, we ask them to perform the instructions, i.e., try to continue building the structure given the clear instruction. Otherwise, when the instructions are decided as ambiguous, we ask the annotators to issue clarifying questions which might helps them to understand the instructions better. Figure 2 in the appendix illustrates the designed data collection web form.

Next, we will provide a detailed overview of the collected dataset and tool.

---

[3]https://github.com/aliannejadi/ClariQ

Table 1: Statistics of single turn and clarifying questions collected data

| Section | #Samples | Instructions | | Average Length (words) | |
		#Clear	#Ambigous	clarifying questions	instructions
train	6843	5951	892	12.12	18.51
test	1293	1129	164	11.99	18.08

3 Dataset and collection Tool description

This section provides details of the collected single-turn and clarifying questions data. As shown in Table 1, in total, after processing and cleaning the data, we have 8,136 single-turn data pairs of instructions and actions. Every single sample is randomly initialized with a pre-built structure from previously collected multi-turn interactions data. We filtered the data by some heuristic criteria, including but not limited to the given instruction must be in "English," and the length of the instructions should not very short. We manually evaluated the data during the curation stage and filtered out jobs that were annotated by the annotators who had low-quality instructions e.g., those who kept issuing the same instructions. As shown in Table 1, the filtered data, on average, has 18 words per instruction. This indicates that the instructions are descriptive enough for a one-minute building process.

After collecting and filtering the single-turn data, we run a data annotation process for clarifying questions on the collected single-turn data as described in the lower section of Figure 1. We kept the filtering heuristics used in single turn data collection for clarifying questions, i.e., the clarifying question must be in English and not too short; as well as included additional filtering criteria. For example, if annotators annotate the instructions as ambiguous, they must have issued a clarifying question. Otherwise, they would get a warning, and the task would be filtered out. This was to ensure that every instruction annotated as "not clear" is accompanied by at least one clarifying question. Similar to filtering single-turn data, we manually evaluated collected clarifying questions data frequently and blocked annotators that kept issuing similar clarifying questions. Out of 8,136 instructions , 1,056 (12.98%) were annotated as *Not Clear* and 7,080 (87.02%) as *Clear* instructions. The average length of clarifying questions is around 12 words, which indicates that the questions are specific.

Table 2 exemplifies some of the collected instructions which were annotated as "not clear" as well as their clarifying questions. Consider the instruction issued by a crowdsourced annotator that says, *"Place four blocks to the east of the highest block, horizontally"*. The corresponding clarifying question issued by another crowdsourced annotator was *"Which color blocks?"*. This reflects that the data collection flow enables a natural, conversational way to get feedback about unclear instructions.

Open Source Collection Tool   We have released the data collection tool and the processed dataset on our GitHub repository[4]. The tool is run on the Amazon Mechanical Turk crowdsourcing platform to collect instructions, actions, gridworld states and clarifying questions in a scalable manner.
While the tool is currently limited to collecting single-turn instruction and action pairs, we plan to develop and release a multi-turn version that would enable multiple agents to interact and collaborate towards achieving a common goal through a shared task.

Dataset Applications   The collected dataset can be used towards solving several Natural Language Processing and Reinforcement Learning tasks. Examples include:

- Teaching an agent to build structures using natural language instructions.

- Generation of instruction given a completed structure.

- Building interactive agents that learn when and what feedback to seek when an unclear instruction is provided. As discussed in Section 2.2, this problem can be formulated into a classification task of when to ask and ranking problem of what to ask .

IGLU Challenge NeurIPS 2022   Above topics can help advance research in the areas of Natural Language Processing and Reinforcement Learning. Towards this, we set up the Interactive Grounded

---

[4]https://github.com/iglu-contest/iglu-dataset/

Table 2: Examples of unclear instructions and their clarifying questions

Instruction	Clarifying Question
Place four blocks to the east of the highest block, horizontally.	Which color blocks?
Destroy 2 purple blocks and then build 3 green blocks diagonally.	Which two purple blocks need to be destroyed?
Destroy the 3 stacked red blocks on the east side. Replace them with 3 stacked blue boxes	Which three of the four stacked red blocks on the east side need to be destroyed?
Make a rectangle that is the width and height of the blue shape and fill it in with purple blocks.	Which side I need to make the rectangle is not clear
Facing South remove the rightmost purple block. Place a row of three orange blocks to the left of the upper leftmost purple block. Place two orange blocks above and below the leftmost orange block.	Which one of the rightmost blocks should be removed?
facing north and purple green blocks will arrange on one by one.	Where would you like to place the purple and green blocks exactly?

Language Understanding(IGLU) Kiseleva et al. [2022b] competition which is part of NeurIPS 2022. The goal of this competition is to *build embodied agents that learn to solve a task while provided with grounded natural language instructions in a collaborative environment*. The data we collected, is used to train models for the Reinforcement Learning and Natural Language Processing sub-tasks of the competition. Next, we elaborate on the baselines we set up for the Natural Language Processing task of the competition.

Baselines   As a practical use case of the collected data for NLP community, we adapt the baselines from Clarifying Questions for Open-Domain Dialogue Systems (ClariQ) challenge on our collected data Aliannejadi et al. [2021]. Given an initialized voxel world with pre-built structure and an instruction, the first goal is to determine whether any clarification question is needed.Thus, for *when to ask Clarifying Questions*, we fine-tuned BERT Devlin et al. [2018] followed by a classification layer to predict if instructions are clear or not. This approach has shown to have promising performance on similar tasks Aliannejadi et al. [2021], Arabzadeh et al. [2022]. Referring to Table 1, we see the majority of instructions are annotated as clear making the classification task of *when to ask* clarifying questions even more challenging by having an unbalanced number of samples for each class. After detecting ambiguous questions, *what Clarifying Questions to ask* would be the main challenge. Inspired by ClariQ dataset Aliannejadi et al. [2020], we curate a pool of clarification questions for each ambiguous instruction and the goal of this ranking-based task is to find the most relevant clarifying question for them. We adapted the well known BM25 to rank the clarifying questions in the question bank Robertson et al. [1995].

For the classification task i.e., When to ask clarifying question, the baseline got the F-1 score of 0.732.In addition, for the ranking task, since we have one clarification question per unclear instruction, we employed Mean Reciprocal Rank for evaluation purposes and BM25 achieved 0.341 in terms of MRR@20. In future, we would like to try question generation baselines on our dataset and report their performance on quality of generated clarifying questions.

4   Conclusions

Studying interactive agents that have the ability of grounded language understanding is crucial to the development of the field. One of the current bottlenecks is a lack of an interactive platform for flexible data collection from multiple humans. In our work, we formalized and simplified an interactive, collaborative task and developed open-sourced tooling for flexible data collection that allows the easy plug-in to any crowdsourcing platform that makes it scalable. Moreover, we have collected an extensive dataset shared with the community to enable the study of grounded language understanding.

References

Josh Abramson, Arun Ahuja, Iain Barr, Arthur Brussee, Federico Carnevale, Mary Cassin, Rachita Chhaparia, Stephen Clark, Bogdan Damoc, Andrew Dudzik, et al. Imitating interactive intelligence.

arXiv preprint arXiv:2012.05672, 2020.

Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft. Asking clarifying questions in open-domain information-seeking conversations. In *International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, SIGIR '19, 2019a.

Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W Bruce Croft. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd international acm sigir conference on research and development in information retrieval*, pages 475–484, 2019b.

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, Jeff Dalton, and Mikhail Burtsev. Building and evaluating open-domain dialogue corpora with clarifying questions. pages 4473–4484, 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.

Negar Arabzadeh, Mahsa Seifikar, and Charles LA Clarke. Unsupervised question clarity prediction through retrieved item coherency. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 3811–3816, 2022.

Pavel Braslavski, Denis Savenkov, Eugene Agichtein, and Alina Dubatovka. What do you mean exactly? analyzing clarification questions in cqa. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*, pages 345–348, 2017.

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.

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`.

Gergely Csibra and György Gergely. Natural pedagogy. *Trends in cognitive sciences*, 13(4):148–153, 2009.

Marco De Boni and Suresh Manandhar. An analysis of clarification dialogue for question answering. In *Proceedings of the 2003 human language technology conference of the north american chapter of the association for computational linguistics*, pages 48–55, 2003.

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.

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.

Julia Kiseleva, Ziming Li, Mohammad Aliannejadi, Shrestha Mohanty, Maartje ter Hoeve, Mikhail Burtsev, Alexey Skrynnik, Artem Zholus, Aleksandr Panov, Kavya Srinet, Arthur Szlam, Yuxuan Sun, Katja Hofmann, Michel Galley, and Ahmed Awadallah. Neurips 2021 competition iglu: Interactive grounded language understanding in a collaborative environment. *arXiv preprint arXiv:2110.06536*, 2021.

Julia Kiseleva, Ziming Li, Mohammad Aliannejadi, Shrestha Mohanty, Maartje ter Hoeve, Mikhail Burtsev, Alexey Skrynnik, Artem Zholus, Aleksandr Panov, Kavya Srinet, et al. Interactive grounded language understanding in a collaborative environment: Iglu 2021. In *NeurIPS 2021 Competitions and Demonstrations Track*, pages 146–161. PMLR, 2022a.

Julia Kiseleva, Alexey Skrynnik, Artem Zholus, Shrestha Mohanty, Negar Arabzadeh, Marc-Alexandre Côté, Mohammad Aliannejadi, Milagro Teruel, Ziming Li, Mikhail Burtsev, Maartje ter Hoeve, Zoya Volovikova, Aleksandr Panov, Yuxuan Sun, Kavya Srinet, Arthur Szlam, and Ahmed Awadallah. Interactive grounded language understanding in a collaborative environment at neurips 2022: Iglu 2022. 2022b.

Noriyuki Kojima, Alane Suhr, and Yoav Artzi. Continual learning for grounded instruction generation by observing human following behavior. *Transactions of the Association for Computational Linguistics*, 9:1303–1319, 2021.

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.

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.

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 E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. Okapi at trec-3. *Nist Special Publication Sp*, 109:109, 1995.

Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Towards facet-driven generation of clarifying questions for conversational search. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 167–175, 2021.

Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Exploiting document-based features for clarification in conversational search. In Matthias Hagen, Suzan Verberne, Craig Macdonald, Christin Seifert, Krisztian Balog, Kjetil Nørvåg, and Vinay Setty, editors, *Advances in Information Retrieval*, pages 413–427, Cham, 2022. Springer International Publishing. ISBN 978-3-030-99736-6.

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`.

Svetlana Stoyanchev, Alex Liu, and Julia Hirschberg. Towards natural clarification questions in dialogue systems. In *AISB symposium on questions, discourse and dialogue*, volume 20, 2014.

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.

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.

Hamed Zamani, Susan Dumais, Nick Craswell, Paul Bennett, and Gord Lueck. Generating clarifying questions for information retrieval. In *Proceedings of The Web Conference 2020*, pages 418–428, 2020.

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`.
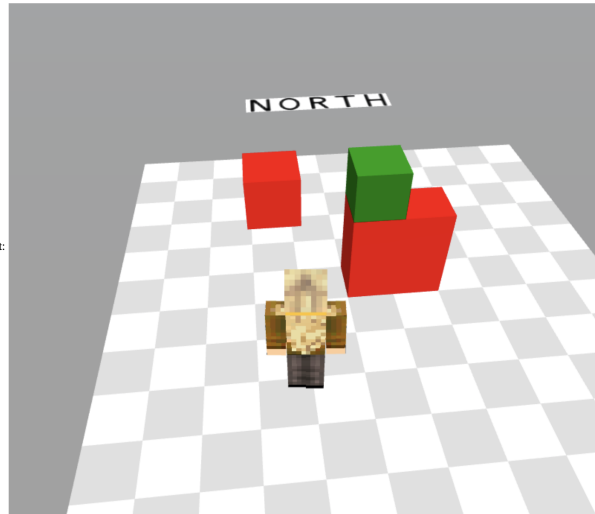
A Appendix

Figure 2: A web view of clarifying question data collection on Amazon Mechanical Turk