


# Natural Language Instructions for Human-Robot Collaborative Manipulation

Journal Title  
XX(X):1-4  
©The Author(s) 2016  
Reprints and permission:  
sagepub.co.uk/journalsPermissions.nav  
DOI: 10.1177/ToBeAssigned  
www.sagepub.com/  


Rosario Scalise\*, Shen Li\*, Henny Admoni, Stephanie Rosenthal, and Siddhartha S. Srinivasa

## Abstract

This paper presents a dataset of natural language instructions for target object reference in manipulation scenarios. It is comprised of 1582 individual written instructions which were collected via online crowdsourcing. This dataset is particularly useful for researchers who work in natural language processing, human-robot interaction, and robotic manipulation. In addition to serving as a rich corpus of domain-specific language, it provides a benchmark of image/instruction pairs to be used in system evaluations as well as uncovers inherent challenges in tabletop object specification. Example code is provided for easy access via Python.

## Keywords

Natural Language Instructions, Human-Robot Collaboration, Manipulation, Ambiguity, Perspective, Spatial Reference

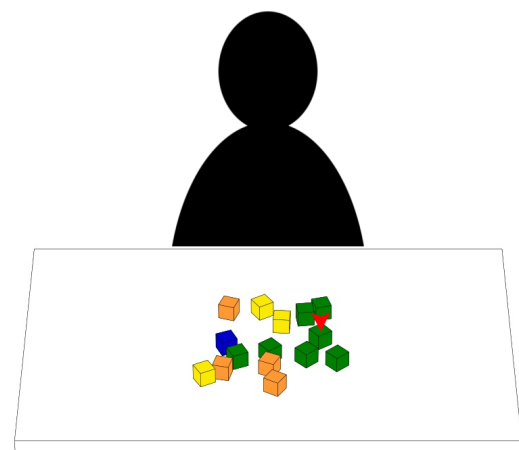
## Introduction

In this paper, we present a corpus of natural language instructions used to specify objects of interest in a collaborative tabletop manipulation setting. Understanding and generating natural language is essential for fluid human-robot collaboration. We are particularly interested in situations where there are many potential objects to reference. To this end, we provide a dataset of scenarios accompanied by human generated instructions specifying tabletop objects. Fig. 1 shows an example of one such scenario.

These scenarios embody the challenges inherent in object specification. Clutter is a pervasive issue in dealing with environments found in our daily lives. [cite] One major challenge is that of object reference ambiguities. These ambiguities stem from phenomena such as object uniqueness, object proximity, and perspective. Humans use natural language strategies to overcome the limitations such phenomena impose. Identifying these strategies and finding methods to resolve ambiguities is important to researchers in the fields of robotics and natural language processing.

We provide researchers with a dataset from which these strategies can be extracted. We include the images used to elicit the instructions and data collected in evaluating instruction performance as well as lexical categorization data. Instructions were generated by human participants via Amazon's Mechanical Turk (AMT) <sup>1</sup>. Data coders labeled each instruction with the number of objects it could refer to and the type of perspective present in each instruction. We also provide a supplementary dataset collected via AMT in which the original set of instructions was measured through correct object resolution accuracy by a separate set of participants.

Image annotation datasets have seen significant growth in popularity in the past decade, particularly within the computer vision community. Two such examples are



**Figure 1.** Example of block configuration stimulus image. Image 6, Configuration Version 1

ImageNet (Deng et al. 2009) and VQA (Antol et al. 2015). These datasets, which combine images and natural language, serve as baselines for informing and evaluating algorithms relating language, semantics, and object recognition. Our dataset aims to provide an analogous set of baseline scenarios catered towards tabletop manipulation.

In a similar vein, robotics researchers have put considerable efforts into establishing corpora for use in navigational instructions which are inherently spatial tasks. Skubic et al. (2012) collect a set of indoor route following instructions

---

Personal Robotics Laboratory, Robotics Institute, Carnegie Mellon University, USA

## Corresponding author:

Rosario Scalise & Shen Li, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 15213, USA

Email: {rscalise,shenl1}@andrew.cmu.edu

by showing participants a target object within a virtual room and asking them to instruct an avatar to go to the target. They examine varying the participants' prompts, the type of agent being instructed, and the perspectives they should use. Similarly, MacMahon and Stankiewicz (2006) collects first-person navigational instructions in a virtual 3D environment, and then evaluates these instructions with a separate set of participants to test instruction effectiveness. Unlike our dataset, they place emphasis on environmental landmarks rather than perspective or ambiguity. We note that while these works focus on navigational trajectories, we focus on defining manipulation goals that are not necessarily constrained to particular trajectories.

Towards the goal of fluent interfacing for robotic tabletop manipulation, Bisk et al. (2016) contribute a dataset. The authors generated configurations of blocks on a tabletop where each block is uniquely identified by a number or symbol on its faces. They asked participants to give instructions to transform one configuration to another and recorded the instructions given as series of steps called Problem-Solution Sequences.

The dataset we present draws from a closely related line of work, with a greater emphasis on the ambiguities often encountered in real-world settings when issuing manipulation instructions. For example, in manipulating objects amongst clutter, it is common to inadvertently give descriptions which can refer to multiple targets. This raises the issues of resolving spatial references, understanding visual feature landmarks, and utilizing perspective in an effective manner. Li et al. (2016) uses this dataset to present initial results that explore these issues.

Researchers interested in utilizing natural language commands to perform collaborative manipulation tasks (particularly in cluttered environments) should find the presented corpus valuable in establishing a baseline of unconstrained human performance when describing ambiguous objects on a tabletop. NLP researchers can also utilize this as a corpus of instructional language as well as corpus dense with spatial terms. Applications where unambiguous natural language is critical include providing robotic accessibility in the home to people with mobility disabilities, enabling mutual understanding via language in a robot-human collaboration task such as a robotic sous chef in a restaurant kitchen, and interfacing with an industrial PCB pick-and-place robot where discrete components can differ only slightly in appearance.

The dataset can be found at the accompanying website:

[https://personalrobotics.github.io/collaborative\\_manipulation\\_corpus/](https://personalrobotics.github.io/collaborative_manipulation_corpus/).

RS: include copywrite or license?

## Data Collection Methods

### Participants and Demographics

All participants were sourced through AMT. 120 participants were involved in the first study and 356 participants were involved in the second study. These two groups of participants were mutually exclusive.

Demographics were collected for each user. We asked participants to report their Age, Gender, Occupation, Computer Usage, Hand Dominance, English as Primary

Language, and Experience with Robots, RC Cars, First-Person-Shooter (FPS) Video Games, and Real-Time-Strategy (RTS) Video Games.

We took care to ensure study designs avoided any confounding participant specific variables such as color-blindness. For example, red blocks were not used because red-green color blindness is the significantly predominant form of the deficiency. (Judd 1943)

### Study 1: Collecting Natural Language Instruction Corpus

The purpose of the first study was to collect a corpus of instructions. In particular, instructions elucidated from scenarios where often times blocks are not easily uniquely identified and spatial references or other visually apparent traits must be utilized to aid in object identification. We created a set of stimulus images from which a randomly selected subset would be presented to each participant. An example of a stimulus image is shown in .

Each image in the set consists of 15 randomly-spaced, randomly-colored blocks (orange, yellow, green, or blue) resetting on a table with a silhouetted figure behind the table. The silhouetted figure represents a partner the participant is to interact with. We synthesized 14 images and selected 2 blocks from each image which are indicated by an arrow to elicit a range of possible participant responses. In total, there are 28 unique scenarios in the set of stimuli.

We presented each participant with 14 randomly selected stimuli from the full set of 28. For each stimulus, we asked the participant to instruct the silhouetted figure to pick up the block indicated by the arrow. We randomly selected and told each participant the partner they were instructing was either a human or a robot. We held this assignment constant across all stimuli shown to a particular participant.<sup>2</sup> The participant entered their response into a textbox in typed natural language. We also asked the participant to subjectively rate the difficulty of creating the instruction for each stimulus on the Likert scale (1 (easy) to 5 (difficult)). At the end of the sequence of stimuli, we asked each participant 1) if they employed any particular strategy in completing the task, 2) how challenging they found the overall task, and 3) for their general comments.

We collected 1582 instructions in total. For each instruction, we captured meta-data such as the total time duration it took the participant to write the instruction after being shown the stimulus.

### Study 2: Evaluating Corpus

Given the original set of stimuli and the set of instructions collected in the first study, we collected responses from new participants indicating which block the participant believed the accompanying instruction specified within the stimulus scenario. For each stimulus, we removed the indication arrow.

We showed the participant the stimulus/instruction pair with an interactive selection interface which displayed a circle around each block as the participant hovered their mouse as seen in 2. When the participant clicked on the block they believed the instruction specified, the circle would change to a checkered pattern and remain overlaid

on the image. The participant was free to change their selection as many times as necessary and their time to select was unconstrained. We gave the participant no feedback to indicate the accuracy of their selection.

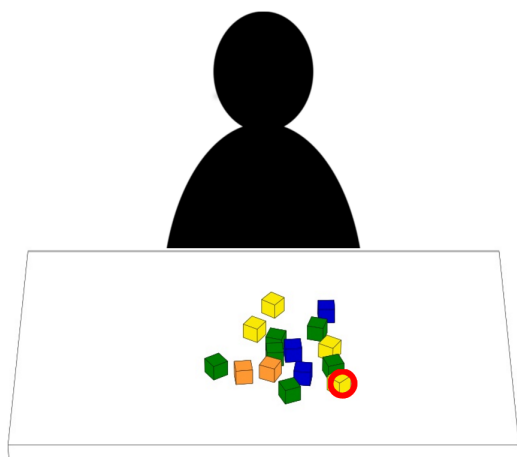
For a subset of 1400 instructions randomly sampled from the full set after removing invalid responses, we collected the number of blocks a participant clicked on while selecting their final answer as well as the final answer itself. We collected approximately 10 final answers per stimuli/instruction pair across all participants and computed the accuracy for a particular instruction.

## Data Coding

We coded the corpus of instructions for two distinct purposes. The first purpose was to determine the ambiguity present within an instruction. To address this, we defined a set of criteria based upon whether an explicit perspective was used, and if so which perspective the instruction was given in according to the set: { partner, participant, neither }. We labeled each instruction with one of these perspective or the ambiguous ‘unknown’ perspective. In addition, we labeled each instruction according to how many blocks could possibly be specified without using any inference. As an example, if there were 3 blocks fitting the description ‘red block near you’, then the block ambiguity for this instruction would be 3.

The second purpose was to develop a categorization of descriptive features. We performed hand-coded word binning for each unique word in the set of all instructions according to a set of categories we determined during analysis.

Perspective and block ambiguity was manually coded by four raters. In order to establish interrater reliability, we required each of the four coders to code the same 10% of the full dataset. We corrected for any discrepancies in coding until a high interrater reliability was achieved. This was confirmed by conducting pairwise Cohen’s  $\kappa$  tests and averaging the results. For coding perspective, the  $\kappa$  value was .85 which indicates a very high interrater reliability. For coding block ambiguity, the  $\kappa$  value was .68 which indicates a high interrater reliability. Once the reliabilities were established, each of the four coders processed one quarter of the data and the results were merged.



**Figure 2.** Example of evaluation stimulus image with on red circle indicating mouse hover block selection. Image 15

## Dataset Details

### Instruction Corpus Table

The file named `NLICorpusData.csv` which contains the initially collected corpus of natural language instructions is formatted such that each line corresponds to an individual instruction (line 0 is a header which contains the field names). Note that there are multiple instructions which correspond to a given image. Here, we briefly describe each field’s relationship to its corresponding instruction.

The *Instruction* field contains the string(s) entered by the participant along with an *Index* for ease of reference. Each instruction corresponds to a particular *Scenario* image which is named by the originating image and its variant (1 or 2). The *AgentType* indicates whether the instruction was given to a human or robot.

Each instruction was also given a *Difficulty* rating by the participant which indicates how challenging they thought a particular prompt was when they generated the instruction. Additionally, the *TimeToComplete* a given instruction was recorded. The duration was started immediately following the current prompt loading and ended when the participant clicked next after entering their instruction and difficulty rating for the prompt.

At the end of the study, an individual participant is asked a series of questions to collect additional information and demographics. Note the participant is only asked once, however, for each instruction the corresponding answers are repeated in the table for ease of reference. These questions include *Strategy* which asks the user to enter their general comments on the strategies they might have used to generate instructions throughout the study, *Challenging* which asks how challenging the participant found the overall study, *GeneralComments* which asks for the participant’s general comments on the study, the participant’s *Age*, *Gender*, and *Occupation*, the *ComputerUsage* habits of the participant indicated by the number of hours per week, whether the participant has *EnglishAsFirst* language, and finally the participant’s experiences with robots, RC cars, first-person shooter video games, and real-time strategy video games indicated by a number between 1-5 according to the Likert scale (where 1 is strongly disagree and 5 is strongly agree). Additionally, we assigned a unique user ID to each participant that is distinct from the Amazon Mechanical Turk ID. A summary of all fields is shown in Table 1.

### Corpus Evaluation Table

The file named `evaluationData.csv` summarizes the results from the evaluation study. Many of the fields are the same as those included in the instruction corpus table. We will briefly describe the non-overlapping fields found in the corpus evaluation table.

The *NumOfWords* field specifies the number of words in the corresponding instruction. *TargetBlockId* is the annotated block number that was specified when the instruction prompt was given (i.e. the correct block). *ClickedBlockId* is the annotated block number that the participant clicked as their final decision before pressing the next button. *Correctness* is a boolean indicating whether the clicked block was the intended target block. In other words, *Correctness* is true if and only if the participant made a final selection on the block

that matches the one shown in the prompt during the first study when the instruction was generated.

Aside from these fields, there are a few which contain comments that correspond with a particular participant who took the study rather than an individual instruction. These include *DifficultyComm* which asks for the participant's comments on the overall study difficulty, *ObsHardComm* which asks for the participant's observations on what made instructions hard to understand, *ObsEasyComm* which asks for the participant's observations on what made instructions easy to understand, and *AddiComm* which asks for any additional comments the participant may have.

An additional file we include named `evaluationDataAvg.csv` contains the averaged responses from 10 participants for each instruction. *ClickedBlockIdList* is a list of 10 annotated block numbers that correspond to each final block selection made across 10 participants that were assigned the particular corresponding instruction. It is composed of 10 of the previously mentioned *ClickedBlockId* fields in the `evaluationData.csv` file. *InternalIDList* is a list of the 10 participants respectively. In a similar fashion, *AccuracyAvg* indicates the average correctness for the corresponding instruction and *TimeToCompleteAvg* indicates the average time to complete the evaluation prompt for the corresponding instruction. A summary of all fields is shown in Table 2

## Accessing the Dataset

### Natural Language Instruction Corpus

We provide simple Python code to initialize data structures which allow for versatile access of the dataset. The file `access_NLICorpusData_CSV.py` loads *NLICorpusData.csv* and creates a dictionary of lists as well as a list of dictionaries. Note only one data structure is necessary for access, however each provides different indexing benefits. The argument for the dictionary is a key corresponding to the field in the table header while the argument for the list is the index corresponding to a particular instruction. Basic examples are included to demonstrate intended usage.

### Supplementary Evaluation Dataset

Similarly, we provide Python code for accessing the evaluation dataset. This code functions in exactly the same manner as the access code for the NLI corpus. The file `access_evaluationData_CSV.py` loads both *evaluationData.csv* and *evaluationDataAvg.csv* and creates appropriate datastructures. Examples for accessing these files are also included. It is important to note that there are 'holes' in the lists for these files since they were downsampled to 1400 individual instructions from the original corpus of 1582. Certain list entries will simply return nothing.

For the evaluation dataset, we include the original data collected in the JSON format as well as the easier to view CSV format. We provide access code named `access_EvaluationData_JSON.py` for the files *evaluationData.json* and *evaluationDataAvg.json*. Note this data is exactly the same as the above CSV version - it is simply included for completeness.

## Acknowledgements

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. [Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution. Carnegie Mellon is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University. DM-0003432. This work was (partially) funded by the DARPA SIMPLEX program through ARO contract number 67904LSDRP, National Institute of Health R01 (#R01EB019335), National Science Foundation CPS (#1544797), the Office of Naval Research, and the Richard K. Mellon Foundation.

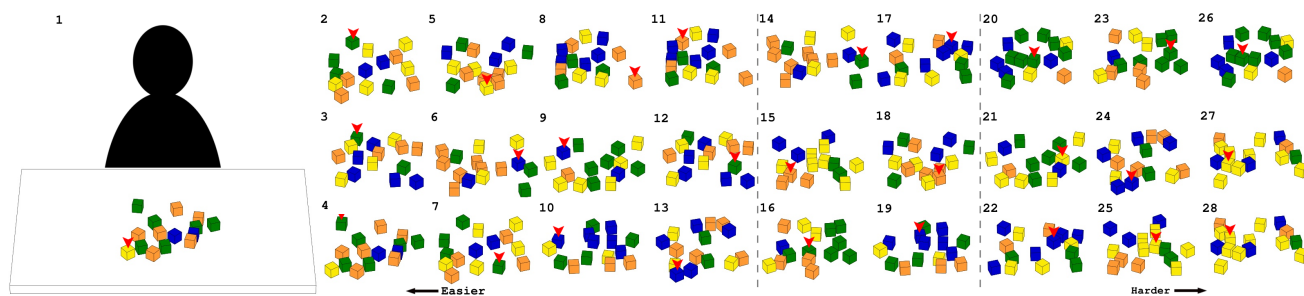
## Notes

1. <https://www.mturk.com>
2. The silhouetted figure was not changed.

## References

- S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. In *International Conference on Computer Vision (ICCV)*, 2015.
- Y. Bisk, D. Yuret, and D. Marcu. Natural language communication with robots. In *Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics*, San Diego, CA, June 2016.
- J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255, June 2009. doi: 10.1109/CVPR.2009.5206848.
- D. B. Judd. Facts of color-blindness\*. *J. Opt. Soc. Am.*, 33(6): 294–307, Jun 1943. doi: 10.1364/JOSA.33.000294.
- S. Li, R. Scalise, H. Admoni, S. Rosenthal, and S. Srinivasa. Spatial references and perspective in natural language instructions for collaborative manipulation. In *IEEE International Symposium on Robot and Human Interactive Communication*, 2016.
- M. MacMahon and B. Stankiewicz. Human and automated indoor route instruction following. *Def*, 2(6):4, 2006.
- M. Skubic, T. Alexenko, Z. Huo, L. Carlson, and J. Miller. Investigating spatial language for robot fetch commands. 2012.





**Figure 3.** All 28 table-top block configuration stimuli arranged by participant subjective difficulty rating

**Table 1.** Overview of fields in Natural Language Instruction Corpus.

Parameter	Note	Data Format
Instruction	Given by participant as response to prompt	String
Index	Corresponds to individual instruction	Integer 1–1582
Scenario	Configuration by image and variation	String with Configuration 1–14 and variation 1–2
AgentType	Agent participant is instructing	String from { <i>robot, human</i> }
Difficulty	Rating for a scenario given by participant	Integer 1–5 (5 being most difficult)
TimeToComplete	Duration participant took to create instruction	Time in hours:minutes:seconds
Strategy	Comments on strategy for used	String
Challenging	Comments on overall study difficulty	String
GeneralComments	Comments on general on overall study	String
Age	Age of participant	Integer in years
Gender	Gender of participant	String from { <i>Male, Female, Prefer not to answer</i> }
Occupation	Occupation of participant	String
ComputerUsage	Duration participant uses a computer per week	String from {0–5, 5–10, 10–15, 15–20, > 20} in hours
DominantHand	Dominant hand of participant	String from { <i>Left, Right, Ambidextrous</i> }
EnglishAsFirst	Participant's first language	1 if yes, 0 if no
ExpWithRobots	Participant experience with robots	Likert scale 1–5
ExpWithRCCars	Participant experience with RC cars	Likert scale 1–5
ExpWithFPS	Participant experience with first-person shooter video games	Likert scale 1–5
ExpWithRTS	Participant experience with real-time strategy video games	Likert scale 1–5
ExpWithRobotComments	Comments on experience with robots	String
InternalUserID	Participant ID unaffiliated with Amazon ID	Integer 1–600

**Table 2.** Overview of fields in Corpus Evaluation.

Parameter	Note	Data Format
Instruction	Instruction being evaluated	String
Index	Corresponds to individual instruction	Integer 1–1582
Scenario	Configuration by image and variation	String with Configuration 1–14 and variation 1–2
NumOfWords	Number of words within an instruction	Positive Integer
TargetBlockId	Corresponds to the target block of an instruction	Integer 1–15
ClickedBlockId	Corresponds to the block participant finally selected	Integer 1–15
Correctness	Whether participant chose the correct target block	1 if right, 0 if wrong
TimeToComplete	Duration participant took to evaluate an instruction	Float number in seconds
DifficultyComm	Comment on the general difficulty of overall task.	String
ObsHardComm	Observation on instructions difficult to interpret	String
ObsEasyComm	Observation on instructions easy to interpret	String
AddiComm	Comment in general on overall study	String
ClickedBlockIdList	Corresponds to the blocks multiple participants respectively selected	List of Integer 1–15
AccuracyAvg	Average correctness among multiple participants	Float 0.0–1.0
TimeToCompleteAvg	Average duration among multiple participants	Float number in seconds
InternalUserIDList	Multiple participant IDs unaffiliated with Amazon ID	Integer 1–600